NKAP BOLTING SYSTEM

TECHNICAL

SPECIFICATION GROUP 01













Team member	Student Number	Email	Contact Number
Jerome Amenigy (Team Leader)	u19131918	u19131918@tuks.co.za	0617993245
Divya Bagratee (Project Manager)	u19078065	u19078065@tuks.co.za	0833680673
S'nethemba Xulu	u19239158	u19239158@tuks.co.za	0659374090
Chardé Devine Vries	u19072912	u19072912@tuks.co.za	0767890267
Sonali Badrinath	u19146851	u19146851@tuks.co.za	0713623778

DELIVERABLE 3 & 4

THIS DOCUMENT CONTAINS THE TECHNICAL SPECIFICATION OF THE SYSTEM WE INTEND TO DEVELOP FOR OUR CLIENT. IN THIS DELIVERABLE WE WILL BE DETAILING THE SYSTEM ARCHITECTURE, SYSTEM DATABASE DESIGN THROUGH THE ERD AND SIZE ESTIMATION OF DATABSASE; AS WELL AS SYSTEM INTERFACE DESIGN USING INPUT AND OUTPUT DESIGN AND ENTITY RELATIONSHIP MODELLING.

Table of Contents

INTRODUCTION	4
SYSTEM ARCHITECTURE DESIGN	5
1. Use Cases INTRODUCTION 1.1. 5 1.2. 17 CONCLUSION	5 5
2. Process/Object-Oriented Design INTRODUCTION 2.1. 398 2.2. 399 2.3. 404 2.4. 445 CONCLUSION	6
SYSTEM DATABASE DESIGN	7
3. 556 INTRODUCTION 3.1. 557 CONCLUSION	7
4. 559 INTRODUCTION 4.1. 559 4.2. 561 CONCLUSION	8
SYSTEM INTERFACE DESIGN	9
5. 562 INTRODUCTION CONCLUSION	9
6. 563 INTRODUCTION CONCLUSION	10 10
7. 596 INTRODUCTION 7.1. 596 7.2. 598 CONCLUSION	11 11
PACKAGE DESIGN SPECIFICATIONS	12
8. 599 INTRODUCTION	12

8.2. 600 8.3. 600 CONCLUSION	12
9. 611 INTRODUCTION CONCLUSION	13 13
10. 613 INTRODUCTION CONCLUSION	14 14
11. 615 INTRODUCTION CONCLUSION	15 15
PROJECT RELATED DOCUMENTS	16
12. Sign-off by Client INTRODUCTION CONCLUSION	16 16 16
13. Complexity INTRODUCTION CONCLUSION	17 17 17
14. General INTRODUCTION 14.1. TEAM SIGN-OFF CONCLUSION	18 18 18 18
CONCLUSION	19

INTRODUCTION

This document contains our technical specification for the NKAP Bolting System. We will look at the system architecture design which includes a Full set of Use Case Diagrams and Design Use Case Documentation of each Use Case. We will also look at Object-Oriented Design and System Database Design which will include the Entity Relationship model and the size estimation of the proposed database. We will then look at the System Interface Design which includes inputs, outputs as well as other interfaces used for our system and lastly Package Design Specifications which includes test specifications, hardware and software requirements, network/web layout specifications, and validations.

SYSTEM ARCHITECTURE DESIGN

1. Use Cases

INTRODUCTION

This section contains the use cases in the form of a full set of use case diagrams as well as Design Use Case Documentation of each Use Case.

1.1. USE CASE DIAGRAMS

	2. Use Case Diagram K	ev
Element	Description	Example
System	Outer Rectangle Element with the system name, 0. NKAP Bolting system and black outline.	
Sub-System	Inner rectangle element with black outline and Sub-System name.	
Use Case	Use Case element with black outline and green fill.	3.1 Register New Customer

Actor	Use Case actor element with green fill and name of actor and actor type below.	< <pba>> Customer</pba>
Association	Use Case relationship line with 'initiates' tag.	
Uses Association	Use case dependency relationship with 'uses' tag.	

Figure 1 Use Case Diagram Key

1.1.1 Login Subsystem

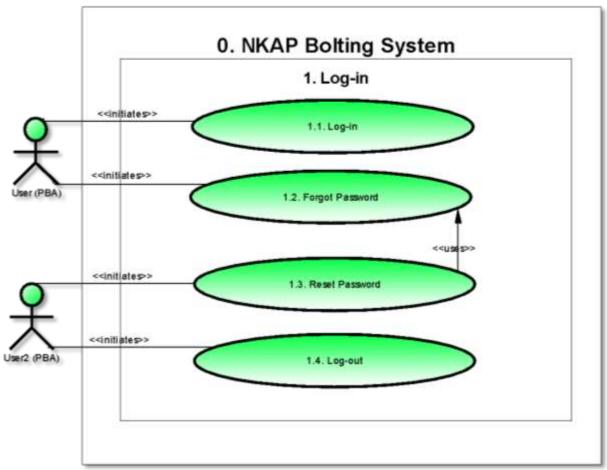
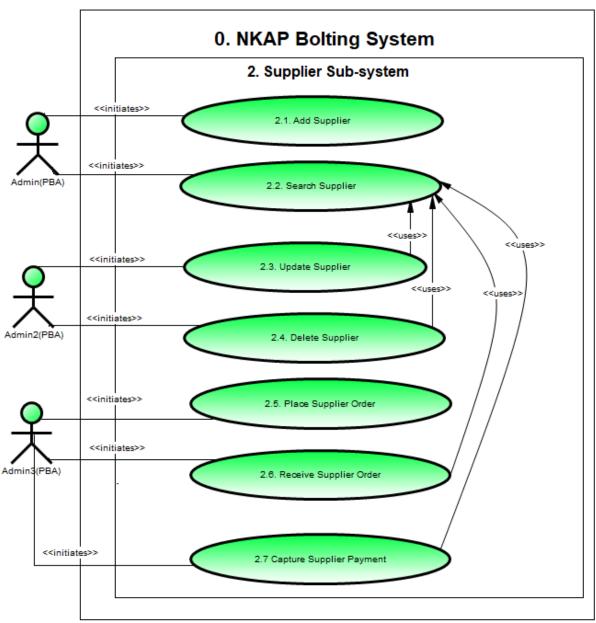


Figure 1 Login Use Case Diagram



1.1.2 Supplier Subsystem

Figure 2 Supplier Use Case Diagram

1.1.3 User Subsystem

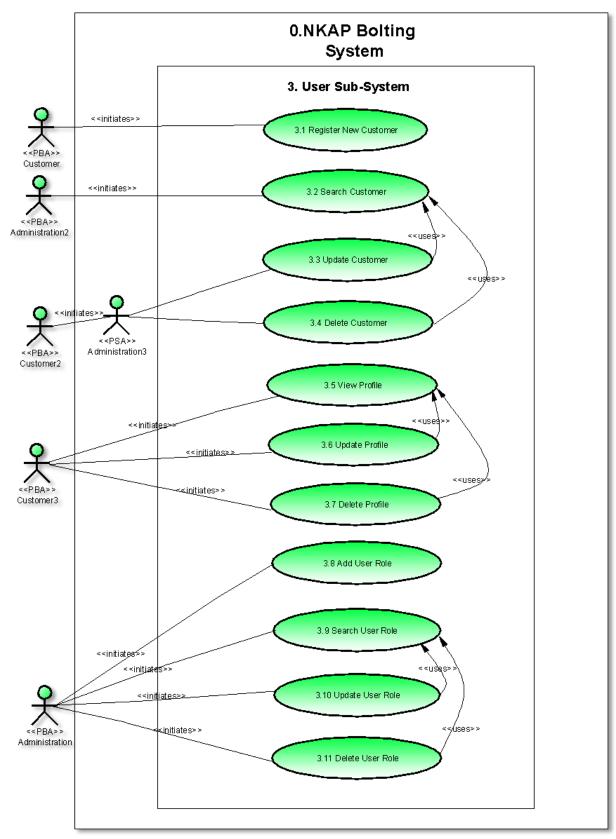


Figure 3 User Use Case Diagram

1.1.4 Employee Delivery Shifts Subsystem

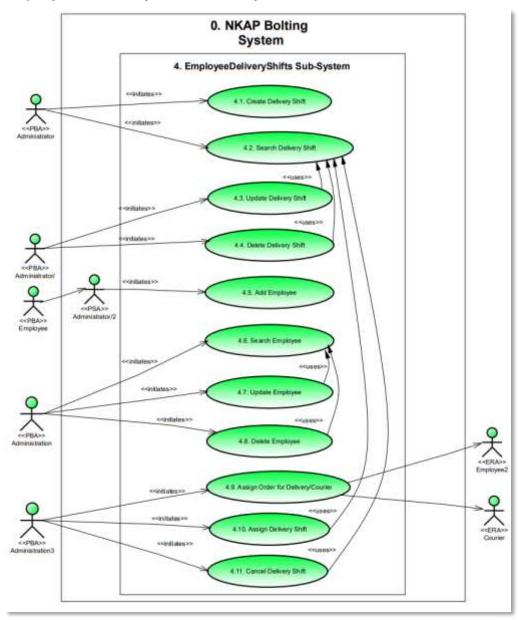


Figure 4 Employee Delivery Shifts Use Case Diagram

1.1.5 Administration Back Office Subsystem

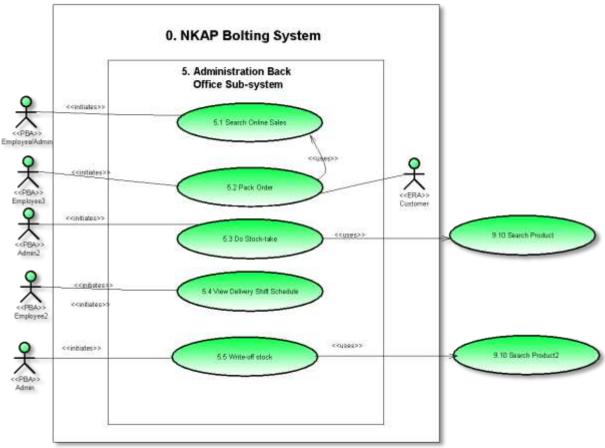


Figure 5 Administration Back Office Use Case Diagram

1.1.6 Customer Subsystem

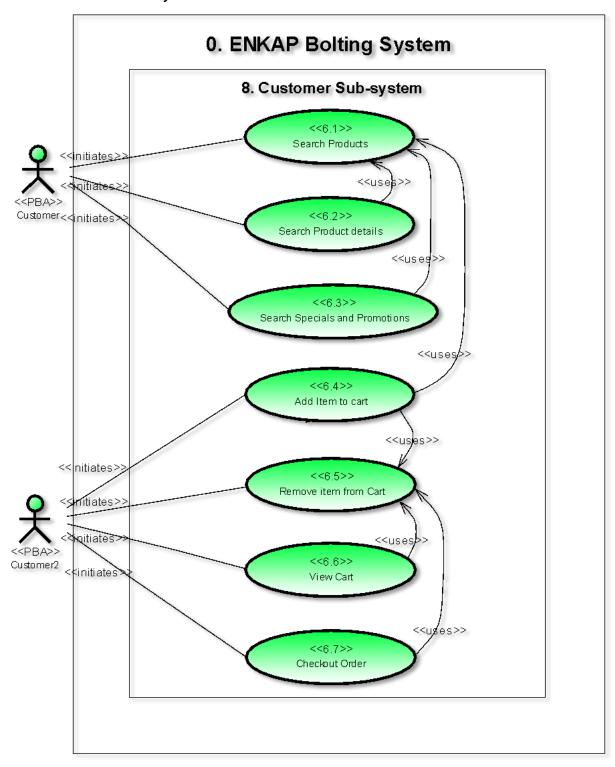


Figure 6 Customer Use Case Diagram

1.1.7 Reports Subsystem

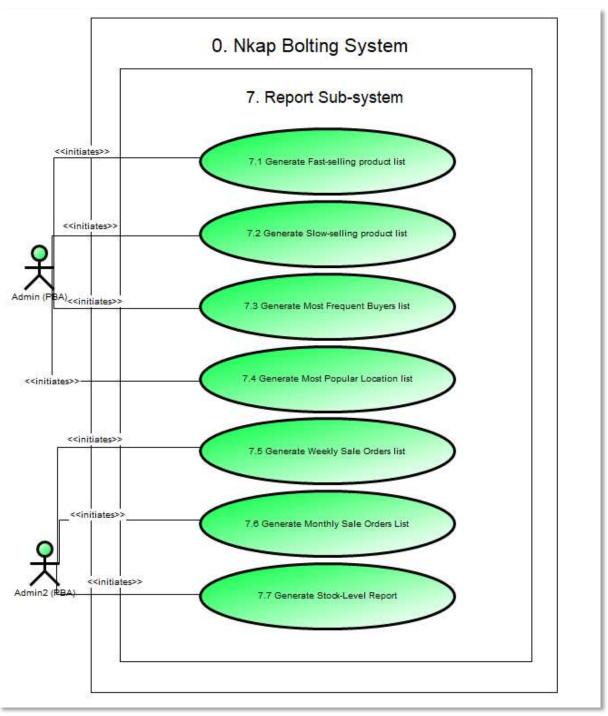


Figure 7 Reports Use Case Diagram

1.1.8 Delivery Subsystem

UPDATED DELIVERY SUBSYTEM

(Paste Use case diagram above the caption)

Figure 8 Delivery Use Case Diagram

1.1.9 Products Subsystem

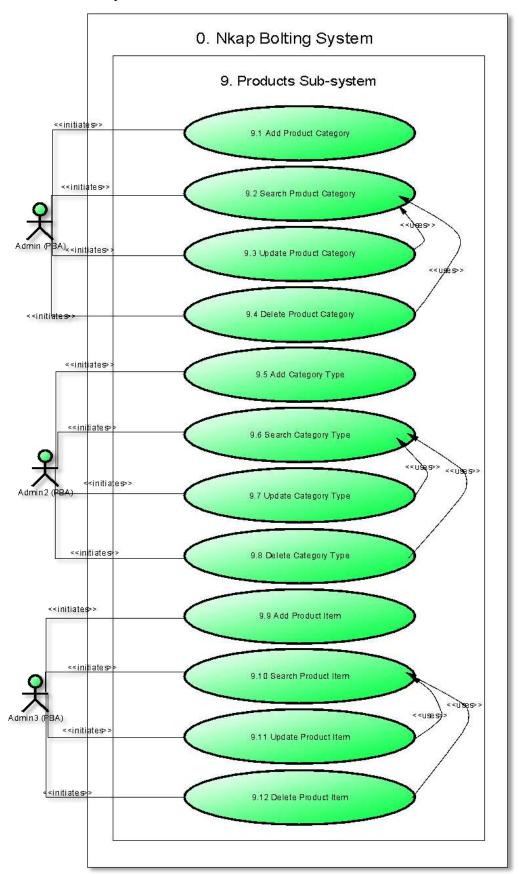


Figure 9 Products Use Case Diagram

1.1.10 Specials Subsystem

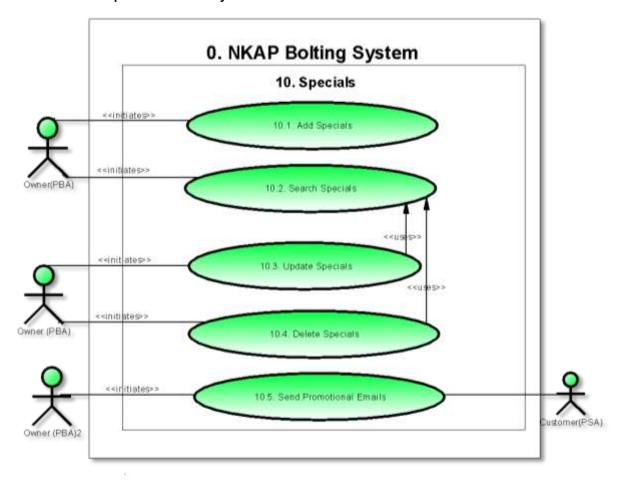


Figure 10 Specials Use Case Diagram

2.1. DESIGN USE CASE DOCUMENTATION

2.1.1. LOGIN SUBSYSTEM

	NKAP BOLTING	SYSTE	VI				
DATE:	03-06-2021						
VERSION:	1.0						
AUTHOR(S)	S'nethemba Xulu						
USE CASE NAME:	Log-in	og-in USE CASE TYPE					
USE CASE ID:	1.1		Business Requirements:□				
PRIORITY:	High		System Analysis: □				
SOURCE:	NKAP Bolting System Requirements		System Design: ☑				
PRIMARY BUSINESS ACTOR:	User (Customer, Admin)						
PRIMARY THE SYSTEM ACTOR:	None						
OTHER PARTICIPATING ACTORS:	None	None					
OTHER INTERESTED STAKEHOLDERS:	None	None					
DESCRIPTION:	The use case describes the event where the user wants to log in to the system. The use case begins when the user enters their Log-in details (username and password). The system will verify the details using the User table. The use case ends when the user is logged in to the system. The system will allow access to functionalities based on their access level defined by their UserRole_ID.						
PRE-CONDITION:	The user should already	be register	red on to the system.				
TRIGGER:	The user wishes to log-in to	the system	1.				
TYPICAL COURSE			SYSTEM RESPONSE:				
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Action				
	Step 1: The user wants to log in to the system.		Step 2: The system will display the log-in screen with the following HTML elements:				
			Input Control Notes name Type				
			Login Heading This serves as the				

Т	Т			
				header of the page.
		Username	Label	None
		Username	Textbox	None
		Password	Label	None
		Password	Textbox	None
		Forgot Password	Link	Invokes Use Case 1.2 Forgot Password.
		Log-in	Button	None
		Register	Button	Invokes Use Case 3.1 Register New Customer
		The system v		
Step 3: The user enters their log-in details.				
Step 4: The user clicks on the log-in button		Step 5: The system captures and validates the user log-in details by reading the Users, Username and Password from the User table. [Alt]		
		Step 6: A SC search for the the User table	e login deta	ery is used to ails entered in
		• Username	е	
		 Password 		
		Step 7: The salgorithm on and compare to the passwo [Alt]	the entered s the enter	d password ed password
		in the AuditT Add. The sys	rail entity tem will ad	add a record using the SQL d the:
		• Usern	ame	
		Date Time		
		• Time		

ALTERNATE	[Alt Step 5]: A validation error occurs because the user entered information that is incorrect. The system will notify the user that their login details incorrect. The system will display a modal with the following controls, to inform the errors.					
	of the error:	•				
	Input Name	Control Type	Notes			
	Validation Error	Label	This serves as a header for the modal.			
	The entered login credentials are incorrect. Please try again.	Label	Tells the user that the details they entered are incorrect.			
	ОК	Button	None			
	Return to step 3.					
	The system will display a modal with of the error:					
	Input Name	Control Type	Notes			
	Validation Error		Notes			
	Validation Error	Label	This serves as a header for the modal.			
	The entered password is incorrect. Please try again.	Label	This serves as a header for the			
	The entered password is incorrect. Please		This serves as a header for the modal. Tells the user that the password they			
	The entered password is incorrect. Please try again.	Label	This serves as a header for the modal. Tells the user that the password they entered is incorrect.			
CONCLUSION:	The entered password is incorrect. Please try again. OK Return to step 3	Label Button ser in to the system an	This serves as a header for the modal. Tells the user that the password they entered is incorrect.			
CONCLUSION: POST-CONDITION:	The entered password is incorrect. Please try again. OK Return to step 3 The system logs the us made available to then • The user will be ab	Label Button ser in to the system ann.	This serves as a header for the modal. Tells the user that the password they entered is incorrect. None			
	The entered password is incorrect. Please try again. OK Return to step 3 The system logs the us made available to then The user will be about the user table is separated.	Button Ser in to the system ann. sole to access the system successfully updated.	This serves as a header for the modal. Tells the user that the password they entered is incorrect. None			

ASSUMPTIONS:	The users have registered before.
OPEN ISSUES:	None

		NKAP	P BOLTING SYSTEM			
DATE:	03-06-2021					
VERSION:	1.0					
AUTHOR(S)	S'nethemba	ı Xulu				
USE CASE NAME:	Forgot Pass	word	USE	E CASE TYPE		
USE CASE ID:	1.2		Business Requiremen	ıts:□		
PRIORITY:	High		System Analysis:			
SOURCE:	NKAP Bolti System Requireme	J	System Design:	☑		
PRIMARY BUSINESS ACTOR:	User (Custo	omer)				
PRIMARY THE SYSTEM ACTOR:	None					
OTHER PARTICIPATING ACTORS:	None					
OTHER INTERESTED STAKEHOLDERS:	None	None				
DESCRIPTION:	they wish to forgot passv	This use case describes the event where the user has forgotten their login details and they wish to create new details. This use case begins when the user clicks on the forgot password link on the login screen. The use case ends when the system opens the reset password screen for the user.				
PRE-CONDITION:	The use	rs login de	etails are incorrect.			
TRIGGER:	The user cli	cking on th	ne forgot password button.			
TYPICAL COURSE	ACTOR		SYSTEM R	RESPONSE:		
OF EVENTS:	ACTION:	Manual Action	Automated Action			

Step 1: The user clicks on	Step 2: The the following		splays the "Fo	orgot Pa	assword" p	page with
the "Forgot password"	Input Name	Control Type	Notes			
link.	Forgot Password	Label	This serves as a header for the modal.			
	Email Address	Label	Prompts the user to enter their email address.			
	Email Address	Textbox	None			
	Submit	Button	Allows the user to submit the Reset password request.			
	Cancel	Button	None			
Step 3: The user enters their email address.						
Step 4: The user clicks on the	Step 5: The been capture					
"Submit" button.	Input Name	Attribute	e in table		Validatio Requirer	
[Alt]	Email Address	Custome	r_EmailAdd		Required 100 character must con "@" and r domain extensior ".com")	rs, tain a

	Step 6: The system the Customer tab email address. [Al	le for a customer h	d Query to search aving the entered
	 Step 7: The system password link and Customer_EmailA	sends it to the cus	
	Step 8: The system sent to their email following controls:		that a link has been -up with the
	Input Name	Control Type	Notes
	Outcome	Label	This is a header for the pop-up
	A password reset link has been sent to your email.	Label	None
	reset link has been sent to	Label	None

ALTERNATE COURSES:

[Alt Step 4]: The user clicks on the "Cancel" button. Terminate the use case.

[Alt Step 5]: The email address captured is invalid. The system notifies the user with the following controls:

Input Name	Control Type	Notes
Invalid Email	Label	This is a header for the pop-up
Invalid email address. Please try again.	Label	None
Ok	Button	None

[Alt Step 6]: The captured email address does not exist in the Customer table. The system will notify the user using a dialog box with the following controls:

	Input Name	Control Type	Notes	
	Email Error	Label	This is a header for the pop-up	
	The entered email address does not exist. Please enter another email address.	Label	None	
	Ok	Button	None	
CONCLUSION:	The use case ende	s when the user ha	as been told that a	link has been sent to
POST-CONDITION:	The email has	been sent to the u	ser.	
BUSINESS RULES:	1. None			
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None			
ASSUMPTIONS:	None			
OPEN ISSUES:	None			

NKAP BOLTING SYSTEM					
DATE:	03-06-2021	03-06-2021			
VERSION:	1.0				
AUTHOR(S)	S'nethemba Xulu				
USE CASE NAME:	Reset Password	USE CASE TYPE			
USE CASE ID:	1.3	Business Requirements:□			
PRIORITY:	High	System Analysis: □			
SOURCE:	NKAP Bolting System Requirements	System Design: ☑			
PRIMARY BUSINESS ACTOR:	User (Customer, Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				

DESCRIPTION: PRE-CONDITION:	This use case describes the event where the user changes their password. The user will be navigated to a "New Password" page. They will enter their new password and click on the submit button. The system will then update the password of the user in the User table. • The user should have forgotten their password.						
	• The user s	should hav	e forgotten th	eir passwo	ord.		
TRIGGER:	The user requ	ests to re	set their passv				
TYPICAL COURSE	ACTOR			TEM RES	PONSE:		
OF EVENTS:	ACTION:	Manual Automated Action Action					
	Step 1: The user navigates to		Step 2: The Password" p	•			
	the "Reset Password" page by clicking on the link they received in their email.		Input Name	Control Type	Notes		
		clicking on the link they received in	clicking on the link they received in	clicking on the link they received in	Reset Password	Label	Prompts the user to enter their new password
			New Password	Textbox	None		
			Confirm Password	Label	Prompts the user to confirm their new password.		
			Confirm Password	Textbox	None		
			Submit	Button	Submits the new password.		
			Cancel	Button	Cancels the whole use case.		
	Step 3: The user enters the password information.						

	Step 4: The user clicks on the "Submit"		Step 5 : The system validates that the information that has been captured matches the validation requirements of the User table [Alt] :			
	button. [Alt]		Input Name		Attribute in Table	Validation Requirements
			Pass	word	User_Password	Max 12 characters
			Confi Pass		User_Password	Must match the above password.
			Step 6 in the I			updated password
			• Use	er_Pas	ssword	
ALTERNATE	[Alt Step 4]: case.	The user	clicks on	the "C	ancel" button. Ter	minate the use
COURSES:	[Alt Step 5]: There is a validation error on the password that the user entered. The system will show a pop-up with the following details:					
	entered. The	e system	will show a		•	
	Input Name	Control Type	will show a		•	
	Input	Control			•	
	Input Name Password does not match with confirm	Control Type	Notes		•	
	Input Name Password does not match with confirm password	Control Type Label	Notes None		•	
CONCLUSION:	Input Name Password does not match with confirm password Ok	Control Type Label	None None	a pop-	•	
CONCLUSION: POST-CONDITION:	Input Name Password does not match with confirm password Ok The user's p	Control Type Label Button	None None will be upon	a pop-	up with the followi	
	Input Name Password does not match with confirm password Ok The user's p	Control Type Label Button	None None will be upon	a pop-	up with the followi	
POST-CONDITION:	Input Name Password does not match with confirm password Ok The user's p	Control Type Label Button	None None will be upon	a pop-	up with the followi	
POST-CONDITION: BUSINESS RULES: IMPLEMENTATION CONSTRAINTS AND	Input Name Password does not match with confirm password Ok The user's part of the user is part of the	Control Type Label Button password will be re	None None will be uponedirected to	dated o the I	up with the followi	

NKAP BOLTING SYSTEM					
DATE:	03-06-2021	03-06-2021			
VERSION:	1.0				
AUTHOR(S)	S'nethemba Xulu				
USE CASE NAME:	Logout	USE CASE T	ГҮРЕ		
USE CASE ID:	1.4	Business Requirement	ts:□		
PRIORITY:	High	System Analysis:			
SOURCE:	NKAP Bolting System Requirements	System Design:	V		
PRIMARY BUSINESS ACTOR:	User (Customer, Admin)				
PRIMARY THE SYSTEM ACTOR:	None				

OTHER PARTICIPATING ACTORS:	None					
OTHER INTERESTED STAKEHOLDERS:	None					
DESCRIPTION:	The use case describes the event where the user wants to log out of the system. The use case begins when the user clicks on the logout option. The system will confirm with the user if they can confirm that they want to logout of the system. The use case ends when the user is successfully logged out.					stem will
PRE-CONDITION:	The user should be log	gged in to	the system.			
TRIGGER:	The user clicks on the log	out item in	the navbar.			
TYPICAL COURSE			SYSTE	EM RESP	ONSE:	
OF EVENTS:	ACTOR ACTION:	Manual	Automated	Action		
	Step 1: The user wants to log out of the system.	Action				
	Step 2: The user clicks on the logout item in the navbar.		Step 3: The confirm thei with the follow	r logout re	quest, using	
			Input Name	Control Type	Notes	
			Outcome	Label	This serves as a header for the pop-up	
			Are you sure you want to logout?	Label	Confirms if the user really wants to logout of the system.	
			Yes	Button	User will be logged out.	
			Cancel	Button	User will remain on the same page.	

	Step 4: The user clicks on the "Yes" button, which confirms the logging out of the user.		Step 5: The system ends the AuditTrail and an SQL delete query is used to remove the AuditTrail from the User table.			
	[Alt]		Details to be removed	Attribute in table	Details	
			AuditTrail_ID	AuditTrail_ID	Token removed at the end of the AuditTrail.	
			Step 6: The sys	tem will log the	user out.	
ALTERNATE COURSES:	[ALT Step 4]: The user cl Terminate use case.	icks on the	e cancel button, v	vhich cancels th	e logout.	
CONCLUSION:	The user is logged out of t	the systen	n.			
POST-CONDITION:	The user is logged out	of the sys	stem.			
	The User table is succ	cessfully u	pdated.			
BUSINESS RULES:	1. AuditTrail_ID is remov	ed from th	e local storage.			
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None					
ASSUMPTIONS:	The user is already logged	d in to the	system.			
OPEN ISSUES:	None					

2.1.2. SUPPLIER SUBSYTEM

NKAP BOLTING SYSTEM						
DATE:	04-06-2021					
VERSION:	1.0					
AUTHOR(S)	S'nethemba Xulu					
Use Case Name:	Add Supplier	Use case type				
Use Case ID:	2.1	Business Requirements: □				
Priority:	High	System Analysis: □				
Source:	NKAP Bolting System Requirements	System Design: ☑				
Primary Business Actor:	Administration					
Primary The system Actor:	None					
Other Participati ng Actors:	None					
Other Interested Stakeholde rs:	None					
Descriptio n:	This use case describes the event where the admin wants to add a new supplier on to the system. The use case begins when the admin requests to add the supplier. The admin will enter all the supplier details and the information					

	will be captured in the Supplier table. The use case ends when the supplier has been successfully added.					
Pre-	The supplier does not exist on the system.					
condition:	The admin is	on the	Supplier scr	een.		
Trigger:	The admin requesting to add a new supplier.					
		system Response:				
Typical Course	Actor Action:	Man ual Actio n	Automated	Action		
of Events:	Step 1: The admin requests to add a new supplier.					
	Step 2: The admin will click on the "Add Supplier" button in the		Step 3: The system responds by loading the "Add Supplier" modal with the following controls:			
	Supplier Screen.		Control Name	Control Type	Notes	
			Add Supplier	Headin g	None	
			Supplier Name:	Label	This is the label for the name of the supplier.	
			Name	Textbox	None	
			Supplier Contact Number:	Label	This is the label for the contact number of	

		the supplier
Contact Number	Textbox	This control should only allow for numeric inputs.
Email Address :	Label	This is the label for the email address of the supplier
Email	Textbox	None
Supplier Address Line 1:	Label	This is the label for the address line 1 of the supplier
Address Line	Textbox	None
Supplier Address Line 2:	Label	This is the label for the address line 2 of the supplier
Address Line	Textbox	None

Supplier Address Line 3:	Label	This is the label for the address line 3 of the supplier
Address Line	Textbox	None
Supplier City/To wn:	Label	This is the label for the city/town of the supplier
City	Textbox	None
Supplier Postal code:	Label	This is the label for the postal code of the supplier
Postal Code	Textbox	None
Supplier Type:	Label	This is the label for the supplier type of the supplier
Туре	Dropdo wn	This will be populated with the information retrieved from the

				SupplierT ype table
	Add	Button	None	
		Cancel	Button	This button is used when the admin decides to terminate the process of adding a new supplier.
	Step 4: The admin enters the details in the respective places:			
	Supplier Name			
	Contact Number			
	Email Address			
	AddressL ine1			
	AddressL ine2			
	• City/Tow n			

	 Postal Code Supplier Type Step 5: The admin clicks on the "Add" button [Alt] 	validates	The system captures and sthe information entered dmin against the Supplier lt]:	
		Input Name	Attribute in table	Validatio n requirem ents
		Name :	Supplier_N ame	Required, maximum of 100 characters
		Conta ct Numb er	Supplier_ ContactNu mber	Required, maximum of 10 characters
	Email	Supplier_ EmailAddre ss	Required, maximum of 100 characters	
		Addre ss Line 1	Supplier_ AddressLin e1	Required, maximum of 100 characters
		Addre ss Line 2	Supplier_ AddressLin e2	Required, maximum of 100 characters

Addre ss Line 3	Supplier_ AddressLin e3	Required, maximum of 100 characters
City	Supplier_ City_Town	Required, maximum of 100 characters
Posta I Code	Supplier_ PostalCode	Required, maximum of 4 numeric characters
insert qu	The system usery to save the on in the Supertakent Attribute in table	e captured
be saved:		
Supplie _ID	er Supplier_I	Generat ed by the system by reading the last Supplier _ID from the Supplie

					and adding it by one.
			Name	Supplier_N ame	None
			Contact Number	Supplier_ ContactNu mber	None
			Email	Supplier_ EmailAddr ess	None
			Address Line 1	Supplier_ AddressLin e1	None
			Address Line 2	Supplier_ AddressLin e2	None
			Address Line 3	Supplier_ AddressLin e3	None
			City	Supplier_ City_Town	None
			Postal Code	Supplier_ PostalCod e	None
Alternate Courses:	[Alt Step 5]: The admin clicks on the "Cancel" button which terminates this use case. Return to step 4				

[Alt Step 6]: The details entered were not in the right format, so the validation failed. The system will display to the admin where the validation errors occurred by showing a red outline where the validation failed.

[Alt Step 7]: The new supplier could not be added to the system. The system will display a modal informing the admin that the saving of the supplier information was unsuccessful.

Input Name	Control Type	Notes
Error	Label	This serves as a header for the modal.
There was an error while saving the new supplier. Please try again later.	Label	This is to show that the saving of the supplier information was unsuccessful.
ОК	Button	None

Conclusio n:	This use case concludes when the supplier has been successfully added on the system.
Post- condition:	The supplier has been added to the Supplier table.
Business Rules:	Only an admin can add a new supplier to the system.
Implement ation Constraint s and	None

Specificati ons:	
Assumptio ns:	None
Open Issues:	None

NKAP BOLTING SYSTEM				
DATE:	04-06-2021			
VERSION:	1.0			
AUTHOR(S)	S'nethemba Xulu			
Use Case Name:	Search Supplier	Use case type		
Use Case ID:	2.2	Business Requirements: □		
Priority:	High	System Analysis: □		
Source:	NKAP Bolting System Requirements	System Design: ☑		
Primary Business Actor:	Admin			
Primary The system Actor:	None			
Other Participating Actors:	None			
Other Interested Stakeholders:	None			

Description:	This use case describes the event where the admin wishes to search for a supplier. The admin will enter the search parameters like the suppler name. The system will search for the supplier based on the parameters the admin entered. This use case concludes when the supplier is shown the results of the search query.			
Pre-condition:	• The su	upplier sh	ould be already exist in the system.	
Trigger:	THE aum		to search for a supplier.	
	Actor Action: Manua Automated Action Action Action			
Typical Course of Events:	Step 1: The admin would like to search for a supplier			
	Step 2: The admin inputs the search criteria and clicks on the		Step 3: The system captures and validates the information against the Supplier table [Alt]:	

	search icon.		
		Step 4: A SQL read query is used to read the following attributes from the Supplier table based on the search criteria:	
		Supplier table:	
		。 Supplier_ID(PK)	
		。 Supplier_Name	
		Supplier_ContactNumb er	
		。 Supplier_EmailAddress	
		。 Supplier_Location	
		。 Supplier_AddressLine1	
		。 Supplier_AddressLine2	
		。 Supplier_City_Town	
		。 Supplier_PostalCode	
		∘ Supplier_Balance	
		SupplierType table:	
		SupplierType_ID(PK)	
		 Description 	
		Step 5: The system will show the search results received from step 6 in a table.	
Alternate		he validation failed. The admin is	
courses:	prompted to enter a new search query.		

Conclusion:	The use case concludes when the admin receives the results of the searched criteria.
Post- condition:	The admin receives the results of the suppliers with the same searched criteria
Business Rules:	1. None
Implementatio n Constraints and Specifications:	None
Assumptions:	None
Open Issues:	None

NKAP BOLTING SYSTEM				
DATE:	04-06-2021			
VERSION:	1.0			
AUTHOR(S)	S'nethemba Xulu			
Use Case Name:	Update Supplier	Use case type		
Use Case ID:	2.3	Business Requirements: □		
Priority:	High	System Analysis: □		
Source:	NKAP Bolting System Requirements	System Design: ☑		
Primary Business Actor:	Admin			
Primary The system Actor:	None			
Other Participati ng Actors:	None			
Other Interested Stakeholde rs:	None			
Descriptio n:	This use case describes the event in which an admin wants to update the details of a supplier. The system will display the previous information that was added before and then the admin will add the new information. This use case ends when the information has been updated successfully.			

Pre-condition:	The admin should be logged on to the system.				
Trigger:	The admin	wishes	s update the d	etails of the	supplier.
		Syste	m Response:		
	Actor Action:	Man ual Actio n	Automated A	ction	
Typical Course	Step 1: The		Step 2: The system invokes use case 2.2 Search Supplier.		
of Events:	admin requests to update the details of a supplier.				
	Step 3: Admin clicks on the "Update"	Step 4: The system responds by loadin the "Update Supplier" modal with the following controls:			•
	button.		Control Name	Control Type	Notes
			Update Supplier	Heading	None
			Supplier Name:	Label	This is the label for the name of the supplier.
			Name	Textbox	None
			Supplier Contact Number:	Label	This is the label for the contact

		number of the supplier
Contact Number	Textbox	This control should only allow for numeric inputs.
Email Address:	Label	This is the label for the email address of the supplier
Email	Textbox	None
Supplier Address Line 1:	Label	This is the label for the address line 1 of the supplier
Address Line	Textbox	None
Supplier Address Line 2:	Label	This is the label for the address line 2 of the supplier
Address Line	Textbox	None
Supplier Address Line 3:	Label	This is the label for the address line 3 of the supplier
Address Line	Textbox	None

Supplier City/Town :	Label	This is the label for the city/town of the supplier
City	Textbox	None
Supplier Postal code:	Label	This is the label for the postal code of the supplier
Postal Code	Textbox	None
Supplier Type:	Label	This is the label for the supplier type of the supplier
Туре	Dropdow n	This will be populated with the information retrieved from the SupplierTyp e table
Update	Button	None
Cancel	Button	This button is used when the admin decides to terminate the process of adding a new supplier.

Step 5: The admin enters the new informati on and clicks on	validates	The system captures the information entegainst the Supplier to	ered by the able [Alt]: Validatio n requirem
the "Update" button. [Alt]	Name :	Supplier_Name	Required, maximum of 100 character s.
	Conta ct Numb er	Supplier_Contact Number	Required, maximum of 10 character s
	Email	Supplier_EmailAd dress	Required, maximum of 100 character s
	Addre ss Line 1	Supplier_Address Line1	Required, maximum of 100 character s
	Addre ss Line 2	Supplier_Address Line2	Required, maximum of 100 character s

Addre ss Line 3	Supplier_Address Line3	Required, maximum of 100 character s
City	Supplier_City_To wn	Required, maximum of 100 character s
Posta I Code	Supplier_PostalC ode	Required, maximum of 4 numeric character s
query to	The system uses a S save the captured in upplier table [Alt]:	
Information to be saved:	ati Attribute in tab e	le Detai Is
Supplie _ID	er Supplier_ID	None
Name	Supplier_Name	None
Contac Numbe	· · · —	ctN None
Email	Supplier_EmailA ress	idd None

	Address Line 1	Supplier_AddressLi ne1	None
	Address Line 2	Supplier_AddressLi ne2	None
	Address Line 3	Supplier_AddressLi ne3	None
	City	Supplier_City_Tow n	None
	Postal Code	Supplier_PostalCo de	None

[Alt Step 5]: The admin clicks on the "Cancel" button which terminates the this use case.

Return to step 1.

[Alt Step 6]: The details entered were not in the right format, so the validation failed. The system will display to the admin where the validation errors occurred by showing a red outline where the validation failed.

Alternate Courses:

[Alt Step 7]: The updated supplier information could not be added to the system. The system will display a modal informing the admin that the saving of the supplier information was unsuccessful.

Input Name	Control Type	Notes
Error	Label	This serves as a header for the modal.
There was an error while saving the new supplier information.	Label	This is to show that the saving of the supplier

	Please try again later.		information was unsuccessful.		
	ОК	Button	None		
Conclusio n:	This use case concludes when the supplier information has been successfully updated on the system.				
Post- condition:	The supplier has been updated in the Supplier table.				
Business Rules:	1. Only an admin can update the information of the supplier in the system.				
Implement ation Constraint s and Specificati ons:	None				
Assumptio ns:	None				
Open Issues:	None				

	NKAP	BOLTING SYSTEM		
DATE:	04-06-2021			
VERSION:	1.0			
AUTHOR(S)	S'nethemba Xu	lu		
Use Case Name:	Delete Supplier	Use case type		
Use Case ID:	2.4	Business Requirements: □		
Priority:	High	System Analysis: □		
Source:	NKAP Bolting System Requirements	System Design: ☑		
Primary Business Actor:	Admin			
Primary The system Actor:	None			
Other Participatin g Actors:	None			
Other Interested Stakeholde rs:	None			
Descriptio n:	wants to delete case begins wh supplier. This us	lescribes the process where the admin the supplier from the system. This use en the Administrator requests to delete a se case concludes when the admin gets supplier has been successfully removed n.		

Pre-	• The a	dmin sł	nould be logg	ed in to th	e system.	
condition:	The supplier should already exist on the systems database.					
Trigger:	The adm	in requ	esting to dele	ete a supp	lier.	
	system Response:					
	Actor Action :	Man ual Actio n	Automated A	Action		
Typical Course of Events:	Step 1: The admin request to delete a supplie r from the system		Step 2: The Search Sup	-	vokes Use c a	ase 2.2
	Step 3: The admin clicks		with the follo	owing cont		al,
	on the "Delete		Control Name	Control Type	Notes	
	" button on the Supplie r screen.	Confirm deletion	Heading	This serves as a header for the modal.		
			Are you sure you want to delete	Label	None	

		this supplier? Yes	Button	This is to confirm that the admin wants to delete a supplier.	
		No	Button	This is selected when the admin does not want to delete a supplier.	
Step 5: The admin clicks on the		Step 6: The query to ren from the Su	nove the s	upplier info	
"Yes" button to confirm	es" tton	Informat ion to be deleted:	Attribute	in table	Details
the deletio n of the supplie r. [Alt]		Supplier _ID	Supplier_I	D	Generat ed by the system by reading the last Supplier _ID from

				the Supplie r table and adding it by one.
		Name	Supplier_Name	None
		Contact Number	Supplier_Contact Number	None
		Email	Supplier_EmailAd dress	None
		Address Line 1	Supplier_Address Line1	None
		Address Line 2	Supplier_Address Line2	None
		Address Line 3	Supplier_Address Line3	None
		City	Supplier_City_To wn	None
		Postal Code	Supplier_PostalCo de	None
		Sten 7: Th	e system removes th	e deleted
		Step 7: The system removes the deleted supplier from the table in the Supplier screen.		
Alternate	[Alt Step 5]: Th	ne admin clic	ks on the "No" buttor	٦.
Alternate	Terminate this use case.			
Courses:	[Alt Step 6]: The supplier could not be deleted from the system. The system will display a modal informing the admin that the deletion of the supplier was unsuccessful.			

	Input Name	Control Type	Notes	
	Error	Label	This serves as a header for the modal.	
	There was an error while deleting the supplier. Please try again later.	Label	This is to show that the saving of the supplier information was unsuccessful.	
	OK	Button	None	
Conclusion :	This use case ends when the supplier has been successfully deleted from the system.			
Post- condition:	The details of the supplier have been removed from the Supplier table.			
Business Rules:	1. Only the admin can remove a supplier.			
Implement ation Constraint s and Specificati ons:	None			
Assumptio ns:	None			
Open Issues:	None			

DATE:	07-06-2021				
VERSION:	1.0				
AUTHOR(S)	S'nethemba Xulu, Jerome Amenigy				
Use Case Name:	Place Supplier Order	Use case type			
Use Case ID:	2.5	Business Requirements: □			
Priority:	High	System Analysis: □			
Source:	NKAP Bolting System Requirements	System Design: ☑			
Primary Business Actor:	Admin				
Primary The system Actor:	None				
Other Participating Actors:	None				
Other Interested Stakeholders:	None				
Description:	This use case describes the event where the admin navigates to the "Place Supplier Order" screen in order to generate a list of what the business wants to order from the supplier. The admin will select all the items they wish to purchase and the quantity. This list will be saved on to the system in the SupplierOrder and SupplierOrderLine tables.				
Pre-condition:	The admin is log	ged on to the system.			
Trigger:	The admin requesting to place an order to the supplier.				
Typical Course	system	Response:			

of Eventor		Manua	Autopostad	A ation	
of Events:	Actor	Manua	Automated	Action	
	Action:	Action			
	Step 1: The admin requests to place a supplier order. Step 2: The admin		"Place Sup	e system dis plier Order"	screen
admin clicks the	clicks on		with the foil	owing contro	UIS.
	"Place		Control	Control	Notes
	Supplier		Name	Туре	
	Order" option under the "Supplier " tab.		Place Supplier Order	Heading	This serves as a header for the screen.
			Supplier	Label	None
			Supplier	Dropdow n	This is for the admin to choose the supplier that the order is going to be placed to.

	Categor y	Label	None
	Categor y	Dropdow n	This is for the admin to choose the category.
	Submit	Button	When this button is clicked a table gets displaye d of all the product items in that category.
Step 4: The admin selects the Supplier and Category .			
Step 5: The admin clicks on the	showing the	e system res e list of the p at category a	roduct

submit button.	The table v				
	Control Name	Control Type	Notes		
	Select	Label	Under the select heading there will be check boxes on each row.		
	Product	Label	Under the product item heading there will be the names of each product item on each row.		
	Price	Label	Under the price heading there will be the prices of each product		

			item on each row.
	Quantity	Label	The admin will select how many units of that product item they want to order.
	At the and	of the se	
	be the follo	owing but	
		owing but	tons:
	be the follo	owing but Control	tons:
	Control Name Place	Control Type	notes

	want and adds the quantity. Step 8: The admin clicks on the "Place Order" button. [Alt]		Step 9: The system will capture the product items chosen and the quantity amounts of each item from the ProductItems table: • ProductItem_Name • ProductItem_CostPrice	
			Step 10: The system will save the information in the SupplierOrderLine and SupplierOrder tables.	
Alternate	[Alt Step 8 Terminate	_	dmin clicks on the "Cancel" button. case.	
Courses:				
Conclusion:	The suppli	er order l	nas been placed on the system.	
Post- condition:	The sup system.	-	er has been saved on to the	
B. Maria	1. Only	an admi	n can place a supplier order.	
Business Rules:	The Supplier must confirm that they have available stock quantities before the supplier order is captured on the system			
Implementatio n Constraints and Specifications:	None			
Assumptions:	None			
Open Issues:	None			

NKAP BOLTING SYSTEM				
DATE:	04-06-2021			
VERSION:	1.0			
AUTHOR(S)	S'nethemba Xulu,	Jerome Amenigy		
Use Case Name:	Receive Supplier Order	Use case type		
Use Case ID:	2.6	Business Requirements: □		
Priority:	High	System Analysis: □		
Source:	NKAP Bolting System Requirements	System Design: ☑		
Primary Business Actor:	Admin			
Primary The system Actor:	None			
Other Participatin g Actors:	None			
Other Interested Stakeholde rs:	None			
Description :	receives the order if the order is corr Then they will cap	scribes the event where the admin r from the supplier. The admin will check ect by comparing it with the invoice. Sture the order on to the system. The use he product item quantity on hand is		

Pre-condition:	The order should have been sent already.					
Trigger:	The admin	The admin receiving the order from the Supplier.				
Typical Course of Events:	Actor Action:	Manu al Actio n	n Response: Automated Action			
	Step 1: The admin checks that the Supplier invoice correspo nds with the items that the Supplier has delivered to the admin.					
	Step 2: The admin accepts the inventory order. Step 3: The admin requests to receive		Step 4: The system invokes Use Case 2.2 Search Supplier.			

the supplier order.		search cri	esponds by teria results w	vith
	Control Name	Control Type	Notes	
	Supplier	Heading	None	
	Search:	Label	This is the label to prompt the user to enter a search query.	
	Search	Textbox	None	
	Search icon	Icon button	None	
	Supplier Table	Table	This is to show the Suppliers that are saved in the database in a table.	
	Supplier Name	Column	This is the column for the supplier names.	

Supplier Contact Number:	Column	This is the column for the contact number of the suppliers	
Supplier Email Address	Column	This is the column for the email address of the suppliers	
Supplier Type	Column	This is the column for the type of supplier.	
Supplier Balance	Column	This is the column for the balance still owed to the supplier.	
Options	Icon	This is to show the update and delete button for each row.	
Update	Button	None	
Delete	Button	None	
Capture Payment	Button	None	

	Receive Supplier Order	Button	None	
Step 6: The admin clicks on the	the followin	upplier O g controls:	rder" screen	with
"Receive Supplier	Control Name	Control Type	Notes	
Order" option.	Receive Supplier Order	Heading	None	
	Supplier Order Table	Table	None	
	Supplier Order ID	Column	None	
	Order Date Placed	Column	None	
	Order Date Received	Column	None	
	Order Date Selector	Date Selector	None	
	Invoice Number	Column	None	
	Invoice Number Textbox	Textbox	None	

	Invoice Date	Column	None	
	Invoice Date	Date Selector	None	
	Selector			
	Invoice Total	Column	None	
	Invoice Total Textbox	Textbox	None	
	Items	Column	None	
	Add Button	Button	None	
	Save Button	Button	None	
Step 8: The admin selects the Supplier Order he wants				
Step 9: The admin enters the relevant order and date				

inform on	nati				
Step The		Step 11: The system displays a modal with the following controls:			
admir clicks the "A	on	Control Name	Control Type	Notes	
the "Items	button in the "Items"	Add Invoice Items	Label	None	
colum	in	Product Item Name	Label	None	
		Name	Textbox	None	
		Product Item Quantity	Label	None	
		Quantity	Textbox	None	
		Product Item Cost	Label	None	
		Cost	Textbox	None	
		Add Button	Button	None	
		Cancel	Button	None	
Step The admin enters the na ,quan and co of the	n s ame tity ost				

items on the invoice			
Step 13: The admin clicks on the add button	Step 14: The system saves and updates the information entered in the: SupplierInvoiceLine Table:		
	Information to be updated:	Attribute in table	Detai Is
	Supplier InvoiceLine	Supplier InvoiceLine	None
	Quantity_ Received	Quantity_ Received	None
	SupplierIte m_ Name	SupplierItem_ Name	None
	LineItem_C ost	LineItem_Cost	None
	SupplierInv oice _ID	SupplierInvoic e_ID	None
Step 15: The admin clicks the "Save Button"	Step 16: The system saves and updates the information entered in the Supplier Table:		
	Information to be updated:		Detai Is

Supplier_Ba ance	I Supplier_Bal ance	None
SupplierInvoi	ce Table:	
Information to be updated:	Attribute in table	Detail s
Supplier Invoice_ID	Supplier Invoice_ID	None
Supplier Invoice_Dat e	Supplier Invoice_Dat e	None
Supplier Invoice_Tot al	Supplier Invoice_Tot al	None
Step 17: The system updates the product item quantity on hand in the ProductItem Table:		
	Attribute in able	Detail s
_	Quantity_on_ha nd	None

Alternate Courses:	[Alt Step 2]: The invoice doesn't correspond with the items delivered. Terminate Use Case. [Alt Step 12] The admin clicks on the cancel button and is taken back to the Receive Supplier order screen.	
Conclusion :	This use case concludes when the information has been updated in the relevant tables.	
Post- condition:	The new items from the Supplier have been captured on to the system successfully.	
Business Rules:	1. None.	
Implementa tion Constraints and Specifications:	None	
Assumptio ns:	None	
Open Issues:	None	

NKAP BOLTING SYSTEM				
DATE:	04-06-2021			
VERSION:	1.0			
AUTHOR(S)	S'nethemba Xulu, Jerome Amenigy			
Use Case Name:	Capture Supplier Payment	Use case type		
Use Case ID:	2.7	Business Requirements: □		

Priority:	High		System Analysis: □					
Source:	NKAP Bol System Requireme	J	System Design: ☑					
Primary Business Actor:	Admin	Admin						
Primary The system Actor:	None							
Other Participating Actors:	None							
Other Interested Stakeholders:	None							
Description:	This use c	ase desc	cribes the event where					
Pre-condition:	 This use case describes the event when the admin wants to capture a payment made to the Supplier. The admin will capture the payment and the information will be added to the Supplier table and the Supplier Payment table. 							
Trigger:	The admir Supplier.	admin requests to capture a payment made to the plier.						
		system	Response:					
Typical	Actor Action:	Manu Automated Action al Action						
Course	Step 1:		Step 2: The system invokes Use					
of Events:	The admin requests to capture a payment		Case 2.2 Search Supplier .					

made to the supplier.	=	search cri	esponds by teria results trols:
	Name	Туре	
	Supplier	Heading	None
	Search:	Label	This is the label to prompt the user to enter a search query.
	Search	Textbox	None
	Search icon	Icon button	None
	Supplier Table	Table	This is to show the Suppliers that are saved in the database in a table.
	Supplier Name	Column	This is the column for the supplier names.

Supplier Contact Number:	Column	This is the column for the contact number of the suppliers
Supplier Email Address	Column	This is the column for the email address of the suppliers
Supplier Type	Column	This is the column for the type of supplier.
Supplier Balance	Column	This is the column for the balance still owed to the supplier.
Options	Icon	This is to show the update and delete button for each row.
Update	Button	None
Delete	Button	None
Receive Supplier Order	Button	None

Step 4:		Capture Payment	Button	None		
The admin will click on the	61	Step 5: The system will show the "Capture Payment" screen with the following controls:				
"Capture Payment		Control Name	Control Type	Notes		
" button.		Capture Payment	Heading	y None		
		Supplier Name:	Label	This is the label for the name of the supplier.		
	Name	Textbox	This texbox is already filled with the informatio n of the current supplier. (inactive textbox)			
		Supplier Contact Number:	Label	This is the label for the contact number of the supplier		

Contact Number	Textbox	This texbox is already filled with the informatio n of the current supplier. (inactive textbox)
Email Address:	Label	This is the label for the email address of the supplier
Email	Textbox	This texbox is already filled with the informatio n of the current supplier. (inactive textbox)
Supplier Address Line 1:	Label	This is the label for the address line 1 of the supplier
Address Line	Textbox	This texbox is already

		filled with the informatio n of the current supplier. (inactive textbox)
Supplier Address Line 2:	Label	This is the label for the address line 2 of the supplier
Address Line	Textbox	This texbox is already filled with the informatio n of the current supplier. (inactive textbox)
Supplier Address Line 3:	Label	This is the label for the address line 3 of the supplier
Address Line	Textbox	This texbox is already filled with the

			informatio n of the current supplier. (inactive textbox)
	Supplier City/Tow n:	Label	This is the label for the city/town of the supplier
	City	Textbox	This texbox is already filled with the informatio n of the current supplier. (inactive textbox)
	Supplier Postal code:	Label	This is the label for the postal code of the supplier
	Postal Code	Textbox	This texbox is already filled with the informatio n of the current supplier.

		(inactive textbox)
Supplier Type:	Label	This is the label for the supplier type of the supplier
Type	Dropdow n	This dropdown is already filled with the informatio n of the current supplier. (inactive drop down)
Capture Payment	Label	None
Capture payment	Textbox	None
Add	Button	None
Cancel	Button	This button is used when the admin decides to terminate the process of adding a

			new supplier.	
Step 6: The admin will add the amount they have paid to the Supplier.	Step 7: The s validate the in	-	-	
	Step 8: The s amount of the from the Supp subtract it with	Supplier_ olier table	Balance and	
	Step 9: The system adds the information to the Supplier Table:			
	Information to be added:	Attribute in table	Details	
	Supplier_ Balance	Supplier Balance	_ None	
	SupplierPayr	nent table	:	
	Informatio n to be added:		Detail s	

		Supplier Payment_I D Supplier_	Supplier Payment_I D Supplier_	None		
		Date	Date			
		Supplier	Supplier	None		
		Payment_	Payment_			
		Amount	Amount	Niero		
		Supplier_I D	Supplier_I D	None		
Alternate COURSES:	[Alt Step 7]: Validation failed. Outline the area where the validation occurred.					
Conclusion:	The use case concludes when the system has notified the admin that the information has been captured successfully.					
Post- condition:	The new payment to the supplier has been added in the relevant tables.					
Business Rules:	1. None					
Implementation Constraints and Specifications:	None					
Assumptions:	None					
Open Issues:	None					

2.1.3. USER SUBSYSTEM

		NKAP B	OLTING SYSTEM		
DATE:	03-06-21				
VERSION:	1.0				
AUTHOR(S):	Divya Bagı	ratee			
USE CASE NAME:	Register N Customer	ew	USE CASE	TYPE	
USE CASE ID:	3.1		Business Requirements:		
PRIORITY:	High		System Analysis:		
SOURCE:	NKAP Bolti System Requirement		System Design:	☑	
PRIMARY BUSINESS ACTOR:	Customer				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS :	None				
DESCRIPTION:	This use case describes the event where a Customer wants to registers on the system. The system will prompt the new Customer to provide their details by displaying the Register New Customer Screen. The system will validate the registration information. Once the system has successfully validated the Information, it will be captured and stored in the User Entity and the Customer Entity respectively. The use case ends when the Customer has successfully registered on the system.				
PRE-CONDITION:	The customer attempting to register must not already exist on the system.				
TRIGGER:	A new customer attempts to register on the system.				
TYPICAL	SYSTEM RESPONSE:				
COURSE OF EVENTS:	ACTOR ACTION:	Manua	Automated Action		
		I Action			
	Step 1: A new customer wants to register				

	n the ystem.					
TI cu cl U or na		Step 3: The system displays the login dropdown menu item				
Step 4: The customer	he ustomer	Step 5: The system displays the register new Customer Modal with the following html controls and enables the input controls:				
"lo	licks the ogin" ropdown	Input Name	Control Type	Notes		
menu item and thereafter clicks the "Register" button on the login screen.	Register New Customer	Modal Heading	None			
	Go Back	Button	Enables the Customer to return to the previous screen			
		Title	Label	Hidden. Displayed when the textbox is selected.		
		Title	Dropdown	Enables customer to select their title		
		Name	Label	Hidden. Displayed when the dropdown is selected.		
		Name	Textbox	Enables customer to enter their name		
		Surname	Label	Hidden. Displayed when the textbox is selected.		
	Surname	Textbox	Enables customer to enter their Surname			
	Cellphone Number	Label	Hidden. Displayed when the textbox is selected.			
	Cellphone Number	Textbox	Enables customer to enter their Cellphone Number			

	Email Address	Label	Hidden. Displayed when the textbox is selected.		
	Email Address	Textbox	Enables customer to enter their Email Address		
	VAT Registration Number	Label	Label for VAT Registration Number		
	VAT Registration Number	Textbox	Enables customer to enter their VAT Registration Number		
	Business Name	Label	Hidden. Displayed when the textbox is selected.		
	Business Name	Textbox	Enables customer to enter their Business Name		
	Next	Button	Enables Customer to proceed with registration.		
Step 6: The customer	Step 7 : The system validates the captured customer information according to the validation requirements of the Customer Entity :				
inputs the requested details and clicks	Input Name	Attribute in Entity	Validation Requirement s		
on the	Title	Title_ID	Required		
"Next" button	Name	Customer_ Name	 Required Max 30 character s Alphabet only 		
	Surname	Customer_ Surname	 Required Max 30 character s Alphabet only 		
	Cellphone Number	Customer_ CellPhoneNu er	Required Max 10 character s		

			•	Numeric character s only
	Email Address	Customer_ EmailAddress	•	Required Max 100 character s Must contain "@" and domain extension
	VAT Registratio n Number	Customer_ VATReg	•	Max 10 character s Numeric character s only
	Business Name	Customer_ BusinessName	•	Max 50 character s

Step 8: The system displays the register new Customer Modal with the following html controls and enables the input controls:

Input Name	Control Type	Notes
User Details	Modal Heading	None
Go Back	Button	Enables customer to return to previous screen.
Username	Label	Hidden. Displayed when the textbox is selected.
Username	Textbox	Enables customer to enter their Username
Password	Label	Hidden. Displayed when the textbox is selected.
Password	Textbox	Enables customer to enter their Password
Confirm Password	Label	Hidden. Displayed when the textbox is selected.

		Confirm Password	Textbox	Enables customer to enter their Confirmed Password
		Register	Button	Button when clicked will proceed with the registration process
Step 9: The customer inputs the requested details and clicks on the "Register" button.	i	Step 10: The information acrequirements Input Name Usernam e	ccording to th	Validation Requirements am Required Max 50 characters Alphanumeri c characters only
		Step 11: The modal:	system displ	only lays a Confirm Details
		Input Name	Control Type	Notes
		Confirm Details	Modal Heading	None
		Is your captured information correct?	Label	Label to confirm the captured information is correct
		Cancel	Button	Button clicked when the customer does not confirm their information.
		Confirm	Button	Button clicked when the customer does confirm their information.
				,

Step 12: The customer confirms their details

Step 13: The system uses a SQL insert query to save the new customer information in the **Customer Table**:

Details to be Saved	Attribute in Table	Notes
Customer ID	Customer_ID	The system generates a new Customer_I D by retrieving the last Customer_I D in the Customer table and incrementing it by one.
Title	Title_ID	None
Name	Customer_Name	None
Surname	Customer_ Surname	None
Cellphone Number	Customer_ CellphoneNumbe r	None
Email Address	Customer_ EmailAddress	None
VAT Registratio n Number	Customer_ VATReg	None
Business Name	Customer_ BusinessName	None

And the **User Table**:

Details to be saved	Attribute in table	Notes
User ID	User_ID	The system generates a new User_ID by retrieving the last User_ID in the User table and incrementing it by one.

Username	User_ Username	Label to confirm the captured information is correct
Password	User_ Password	Button clicked when the customer does not confirm their information.
UserRole ID	UserRole_ID	The system sets the UserRole_ID to "customer"
Step 14: The	system saves t	he new customer

Step 14: The system saves the new customer details in the relevant entities and displays a success modal:

Input Name	Control Type	Notes
Success	Modal Heading	None
You have successfully registered	Label	Label to indicate the customer has registered successfully
Ok	Button	None

ALTERNATE

ALT STEP 7 and 10: The system fails to validate the inputted information and displays a Validation Error Modal.

Input Name	Control Type	Notes
Validation Error	Modal Heading	None
Entered details are in the incorrect format	Label	Label to indicate that the inputted details do not fulfil the validation requirements
Ok	Button	None

COURSES:

ALT STEP 12: The customer does not confirm their details and clicks the "cancel" button. **Terminate Use Case.**

CONCLUSION:	the	ALT STEP 14: The system fails to create the new customer and save the details in the User entity and Customer entity and displays an error modal. Terminate Use Case.					
		Input Name	Control Type	Notes			
		Error	Modal Heading	None			
		An unexpected error occurred when saving the customer.	Label	Label to indicate that the customer has failed to register.			
		Ok	Button	None			
POST- CONDITION:	•	The Customer is able System	to log in to t	heir account on the NKAP Bolting			
BUSINESS RULES:	1.	The Customer cannot register twice).	t already exi	st on the system. (They cannot			
IMPLEMENTATIO N CONSTRAINTS AND SPECIFICATIONS :	No	ne					
ASSUMPTIONS:	No	None					
OPEN ISSUES:	No	one					

NKAP BOLTING SYSTEM					
DATE:	03-06-21				
VERSION:	1.0				
AUTHOR(S):	Divya Bagratee				
USE CASE NAME:	3.2	USE CASE TYPE			
USE CASE ID:	Search Customer	Business Requirements:			
PRIORITY:	High	System Analysis:			
SOURCE:	NKAP Bolting System Requirements	System Design:			

PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS :	None				
DESCRIPTION:	The Use Case describes the event where an Admin wants to search for a Customer on the system. The Admin will select the "User" menu option on the navigation bar and click the "Search Customer" dropdown menu item. The system will display the Search Customer Screen and request the search criteria from the admin. The admin will enter the search criteria and the system will search for a matching record in the Customer Entity. The system will display the customer which matched the search criterion entered by the admin in a datagrid view. The admin will select the customer they want to view and the use case ends when the system successfully displays the searched customer's information.				
PRE- CONDITION:	The customer must exist on the system				
TRIGGER:	An admin wants	to search for a	customer on	the system	
TYPICAL	ACTOR		SYSTEM R	ESPONSE	
COURSE OF EVENTS:	ACTION:	Manual Action	Automated A	Action	
	O1 4 - A -				
	Step 1: An admin wants to search for a customer on the system.				
	admin wants to search for a customer on the system. Step 2: The admin selects the User menuitem in the		Step 3: The Search Cus following htr the input co	tomer Moda nl compone	
	admin wants to search for a customer on the system. Step 2: The admin selects the User menu		Search Cus following htr	tomer Moda nl compone	al with the
	admin wants to search for a customer on the system. Step 2: The admin selects the User menu item in the navigation bar and thereafter selects the "Search Customer"		Search Cus following htr the input co	tomer Moda ml compone ntrols: Control	al with the ents and enables
	admin wants to search for a customer on the system. Step 2: The admin selects the User menuitem in the navigation bar and thereafter selects the "Search		Search Cus following htr the input collingut Name Search	tomer Moda ml compone ntrols: Control Type Modal	Notes

Step 4: The admin enters the search criteria and clicks the "search" button.		Step 5: The system searches for a matching record in the <u>Customer</u> Entity using a SQL read query. Step 6: The system displays the record(s) matching the search criteri provided in a datagrid view with the following components:			
		Input Name	Control Type	Notes	
		Customer	Modal Heading	None	
		Customer Name	Table Column1	Column in the datagrid view.	
		Business Name	Table Column 2	Column in the datagrid view.	
		Action	Table Column 3	Column in the datagrid view.	
		Vertical Elipsis & Pencil	Buttons	Buttons when clicked display the options to update and delete the customer.	
Step 7: The admin selects the customer they want to view by clicking the Customer Name.					
		Step 8: The system displays the "View Customer" Modal with the following html components:			
		Input Nam	e Contro Type	l Notes	
		View Customer	Modal Headin g	None	

			-		·	1
			Title		Label	Label for title
			Nam	ne	Label	Label for Name
			Suri	name	Label	Label for Surname
			Cell Num	phone nber	Label	Label for Cellphone Number
			Ema Add	iil ress	Label	Label for Email Address
				istratio ımber	Label	Label for VAT Registratio n Number
			Bus Nam	iness ne	Label	Label for Business Name
			Go I	3ack	Button	Allows Admin to return to previous screen when clicked
ALTERNATE COURSES:	ALT STEP 6: No found and the sys					
	Input Name	Control Typ	е	Notes		
	Error	Modal Head	ing	None		
	No customers matching the search criteria was found.	ners Label the teria		matchin	pel displayed when no tching records were found he database.	
	Ok	Button		None		
CONCLUSION:	The use case con searched Custom			tem succ	essfully dis	splayed the
POST- CONDITION:	The Admin ca	n view the sea	rched	Custome	r's Informa	ation
BUSINESS RULES:	1. Only an Admin	can search for	r a Cu	stomer		

IMPLEMENTATION CONSTRAINTS AND SPECIFICATION S:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

DATE:	03-06-21			
VERSION:	1.0			
AUTHOR(S)	Divya Bagra	atee		
USE CASE NAME:	3.3		USE CASE 1	YPE
USE CASE ID:	Update Cus	stomer	Business Requirements:	
PRIORITY:	High		System Analysis:	
SOURCE:	NKAP Boltin Requiremen	•	System Design:	Ø
PRIMARY BUSINESS ACTOR:	Admin			
PRIMARY THE SYSTEM ACTOR:	None			
OTHER PARTICIPATIN G ACTORS:	None			
OTHER INTERESTED STAKEHOLDER S:	None			
DESCRIPTION:	their accoun Option on the menu item. information and confirm	at information be navigation The system vertrieved fron the changes	the event where a Customer to be updated. The admin wi bar and click the "Search Cuvill enable the admin to edit the the Customer Entity. The same the Customer's information.	Il select the User stomer" dropdown ne customer system will validate
PRE- CONDITION:	The cust	tomer must e	xist on the system	
TRIGGER:	A customer	calls in to rec	quest their account informatio	n to be updated.
TYPICAL	ACTOR		SYSTEM RESPONSE	<u>:</u>
COURSE OF EVENTS:	ACTION:	Manual Action	Automated Action	
	Step 1: A customer calls in to update their informatio n	Step 2: The admin requests the Customer' s Name.		
	Step 3: The customer provides	Step 4: The admin searches for the	Step 5: The system invokes Search Customer.	s Use Case 3.2

	their name.	Customer Profile.				
		Step 6: The admin	Step 7: The system displays the Update Customer Modal with the following html components and enables the input controls:			
	clicks "Upda Butto		Input Name	Control Type	Notes	
			Update Customer	Modal Headin g	None	
			Title	Label	Label for title	
			Title	Textbox	Title_ID is retrieved from the Client Entity	
			Name	Label	Label for Name	
			Name	Textbox	Customer_Name is retrieved from the Client Entity	
			Surname	Label	Label for Surname	
				Surname	Textbox	Customer_Surnam e is retrieved from the Client Entity
				Cellphone Number	Label	Label for Cellphone Number
				Cellphone Number	Textbox	Customer_ CellphoneNumber is retrieved from the <u>Client Entity</u>
			Email Address	Label	Label for Email Address	
			Email Address	Textbox	Customer_ EmailAddress is retrieved from the Client Entity	
			VAT Registratio n Number	Label	Label for VAT Registration Number	
			VAT Registratio n Number	Textbox	Customer_VATRe g is retrieved from the Client Entity	
			Business Name	Label	Label for Business Name	
			Business Name	Textbox	Customer_ BusinessName	

						ieved from ient Entity			
			Update	Button	enable custor the the	mer to update eir nation when			
			Cancel	Button	None				
		Step 8: The admin requests the Customer Informatio n from the customer.							
	The Customer Provides their informatio	Step 10: The admin updates the informatio	The admin updates the informatio	The stomer admin updates ir the	Step 11: The information acrequirements	ccording to t	he valid	dation	
					the informatio	the informatio	Input Name	Attribute i Entity	n
		clicks the "Update"	Title	Title_ID		• Required			
		button.				Name	Customer_ Name	-	 Required Max 30 characte rs Alphabet only
				Surname	Customer_ Surname	-	 Required Max 30 characte rs Alphabet only 		
			Cellphone Number	Customer_ CellPhone ber		 Required Max 10 characte rs Numeric characte rs only 			
			Email Address	Customer_	-	Required			

			EmailAddres		 Max 100 characte rs Must contain "@" and domain extensio n
		VAT Registrati on Number	Customer_ VATReg		 Max 10 characte rs Numeric characte rs only
		Business Name	Customer_ BusinessNa		 Max 50 characte rs
		Step 12: The confirm the chapter confirmation r	anges and di		
		Input Name	Control Type	Note	es
		Confirm Update	Modal Heading	None	9
		Are you sur you want to update the customer information		the u	el to confirm apdate of the omer's mation
		Cancel	Button	wher custo does	on clicked on the omer/admin onot confirm apdate.
		Confirm	Button	wher	omer/admin confirm the
	Step 13: The admin requests confirmati on for the update from the customer.				

	Step 14: The Customer	Step 15: The Admin	Step 16: The sy query to save the Customer Entire	ne updated i	nformation								
	confirms the update.	clicks the "Confirm" Button.	Details to be Updated	Attribute in Table	n	Notes							
	•		Customer ID	Customer_	ID	Cannot be Updated (stays the same)							
		Title	Title_ID		Can be updated								
			Name	Customer_	Name	Can be updated							
			Surname	Customer_ Surname		Can be updated							
			Cellphone Number	Customer_ Cellphonel									
										Email Address	Customer_ EmailAddre		Can be updated
					VAT Registration Number	Customer_ VATReg		Can be updated					
			Business Name	Customer_ BusinessN		Can be updated							
			Step 17: The sy customer's informodal:										
			Input Name	Control Type	Notes								
					Success	Modal Heading	None						
			Customer information was successfully updated.	Label	the cus	•							
			Ok	Button	None								

	Step The admir inform the custo that the inform n was succe ly update	mer heir natio s essful	
ALTERNATE COURSE:	Use Case.		not confirm the update. Terminate
	displays a Validation		validate the updated information and
	Input Name	Control Type	Notes
	Validation Error	Modal Heading	None
	The information entered is in the incorrect format. Please try again.	Label	None.
	Ok	Button	None
	ALT STEP 17: The displays an error m		update the customer information and
	Input Name	Control Type	Notes
	Error	Modal Heading	None
	There was an error when trying to update the customer information.	Label	Label to indicate the system failed to update the customer's information.
	Ok	Button	None
CONCLUSION:	The use case cond Customer's Informa		system successfully updates
POST- CONDITION:	The Customer's	s Information has	s been updated
BUSINESS RULES:			to update their own details. update a customer's details.

IMPLEMENTATI ON CONSTRAINTS AND SPECIFICATION S:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

NKAP BOLTING SYSTEM						
DATE:	03-06-21					
VERSION:	1.0					
AUTHOR(S)	Divya Bagratee					
USE CASE NAME:	3.4	USE CASE TYP	E			
USE CASE ID:	Delete Customer	Business Requirements:				
PRIORITY:	High	System Analysis:				
SOURCE:	NKAP Bolting System Requirements	System Design:				
PRIMARY BUSINESS ACTOR:	Admin					
PRIMARY THE SYSTEM ACTOR:	None					
OTHER PARTICIPATING ACTORS:	None	None				
OTHER INTERESTED STAKEHOLDERS:	None					
DESCRIPTION:	request their account to be d menu option on the navigation dropdown menu item. The sy	The Use Case describes the event where a customer calls in to request their account to be deleted. The Admin will select the User menu option on the navigation bar and click the "Search Customer" dropdown menu item. The system will retrieve the Customer Information Customer Entity . The system will confirm the deletion.				

	The use case ends when the system has successfully deleted the Customer's Account						
PRE-CONDITION:		mer must exist must be logge	•				
TRIGGER:	A customer calls in to request account deletion						
TYPICAL COURSE	ACTOR		SYSTEM RE	SPONSE:			
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Action				
	Step 1: A customer requests the admin to delete their account.	Step 2: The admin requests the Customer's Name					
	Step 3: The customer provides their name	Step 4: The admin searches for the Customer Profile.	Step 5: The system invokes Use Cas 3.2 Search Customer.				
		Step 6: The admin clicks the "Delete"	dmin clicks confirmation for the deletion and displays a confirmation modal:				
		Button.	Input Name	Control Type	Notes		
			Confirm Deletion	Modal Heading	None		
			Are you sure you want to delete this customer?	Label	Label to request confirmation of deletion.		
			Confirm	Button	Button when clicked will proceed with the deletion.		
			Cancel	Button	Button when clicked will cancel the deletion.		
		Step 8: The admin requests confirmation for the Deletion					

	from the Customer.				
Step 9: The Customer confirms the deletion.	Step 10: The Admin clicks the "Confirm"	Step 11: The sy SQL Delete que customer inform Customer Entit	ery to delete nation from	em makes use of a to delete the ion from the	
	Button.	Attribute in E	ntity		
		Title_ID			
		Customer_Na	me		
		Customer_Su	rname		
		Customer_Ce	IIPhoneNu	ımber	
		Customer_En	nailAddres	ss	
		Customer_VA	TReg		
		Customer_Bu	sinessNaı	me	
		And the User E			
		Attribute in ta	ble		
		User_ID			
		User_Usernar			
		User_Passwo			
		Step 12: The sy modal to inform customer was s	the admin	that the	
		Input Name	Control Type	Notes	
		Success	Modal Heading	None	
		The customer was successfully deleted.	Label	Label to indicate the system successfully deleted the customer.	
		Ok	Button	None	
	Step 13: The admin informs the customer that their account was successfully deleted				

ALTERNATE	ALT STEP 9: The customer does not confirm the deletion. Terminate Use Case.					
COURSES:	ALT STEP 12: The system fails to delete the customer and displays an error modal.					
	Input Name	Control Type	Notes			
	Error	Modal Heading	None			
	There was an error when trying to delete the customer.	Label to indicate the system failed to delete the customer.				
	Ok Button None					
CONCLUSION:	The use case concludes when the system successfully deletes the Customer's Account.					
POST-CONDITION:	The Customer's Information has been deleted.					
BUSINESS RULES:	1. The admin must be logged in to delete a customer from the system.					
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None					
ASSUMPTIONS:	None					
OPEN ISSUES:	None					

NKAP BOLTING SYSTEM					
DATE:	03-06-21				
VERSION:	1.0	1.0			
AUTHOR(S)	Divya Bagratee				
USE CASE NAME:	View Profile USE CASE TYPE				
USE CASE ID:	3.5	Business Requirements:			
PRIORITY:	High	System Analysis:			
SOURCE:	NKAP Bolting System Requirements	System Design:	\square		

PRIMARY BUSINESS ACTOR:	Customer				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	This use case describes the event where a customer wants to view their profile. The customer will click on the "User" icon on the navigation bar and clicks the "View Profile" drop down menu item. The system will display the customer information from the Customer Entity . The use case ends when the system has successfully displayed the customer's profile.				
PRE-CONDITION:	The custom	er must b	e logged in		
TD100TD			ave an account o	on the syste	em
TRIGGER:	A customer war	nts to viev	·	DEGRAN) F
TYPICAL COURSE OF EVENTS:	ACTOR	Manual	Automated Acti	RESPONS	DE:
	ACTION:	Action	Addinated Add	011	
	Step 1: A customer wants to view their profile.				
	Step 2: The customer clicks on the "User" icon on the navigation bar and clicks the "View Profile" drop down menu item.		Step 3: The system makes use of a SQL Read query to retrieve the customer information from the Customer Entity. Step 4: The system displays the Profile Modal with the following html components:		
			Input Name	Control Type	Notes
			View Profile	Modal Heading	None
			Title	Label	Label for title
			Name	Label	Label for Name

			Surnar	ne	Label	Label for Surname
			Email Addres	ss	Label	Label for Email Address
			Contac Numbe		Label	Label for Contact Number
			VAT Regist Numbe		Label	Label for VAT Registration Number
			Busine Name	ess	Label	Label for Business Name
			Update		Button	Button which enables the customer to update their information when clicked
			Delete		Button	Button which enables customer to delete their account when clicked.
ALTERNATE	ALT STEP 4: The profile information					the Customer's
	Input Name Control Notes Type					
			None	lone		
	There was an unexpected error when trying to display Customer Profile.	unexpected error when trying to display Customer		Label to indicate the system failed to display the Customer Profile,		
	Ok	Buttor	า	None		
COURSES:						
CONCLUSION:	The use case concludes when the system has successfully displayed the Customer's profile.					
POST-CONDITION:	The customer can view their profile.					
BUSINESS RULES:	1. The customer must be logged in to view their profile.					

IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

NKAP BOLTING SYSTEM					
DATE:	03-06-21				
VERSION:	1.0				
AUTHOR(S)	Divya Bagra	tee			
USE CASE NAME:	Update Profi	ile	USE CAS	SE TYPE	
USE CASE ID:	3.6		Business Requirements:		
PRIORITY:	High		System Analysis:		
SOURCE:	NKAP Bolting System Requirement		System Design:	☑	
PRIMARY BUSINESS ACTOR:	Customer				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	This use case describes the event where a customer wants to update their profile. The customer will select the User Icon on the navigation bar and click the View Profile Drop-Down Menu Item. The system will enable the customer to update the customer information retrieved from the Customer Entity . The system will validate and confirm the changes. The use case ends when the system has successfully updated the Customer's information				
PRE-CONDITION:	The customer must be logged in.				
TRIGGER:	A customer wants to update their profile.				
TYPICAL COURSE	ACTOR SYSTEM RESPONSE:				
OF EVENTS:	ACTION:	Manual Action	Automated Action		
	Step 1: A customer wants to update their profile information.				
	Step 2: The customer selects the User Icon on the navigation bar and		Step 3: The system invokes Profile.	s Use case 3.5 View	

	clicks the						
	"View Profile"						
	Drop-Down Menu Item						
	Step 4: The customer clicks on the "Update"	Step 5: The system displays the Update Profile Modal with the following html components and enables the input controls:					
		Input Name	Control Type	Notes			
	Button.	Update Profile	Modal Heading	None			
		Title	Label	Label for title			
		Title	Textbox	Title_ID is retrieved from the <u>Customer</u> <u>Entity</u>			
		Name	Label	Label for Name			
		Name	Textbox	Customer_Name is retrieved from the Customer Entity			
		Surname	Label	Label for Surname			
		Surname	Textbox	Customer_Surname is retrieved from the Customer Entity			
		Cellphone Number	Label	Label for Cellphone Number			
		Cellphone Number	Textbox	Customer_ CellphoneNumber is retrieved from the Customer Entity			
		Email Address	Label	Label for Email Address			
		Email Address	Textbox	Customer_ EmailAddress is retrieved from the Customer Entity			
		VAT Registration Number	Label	Label for VAT Registration Number			
		VAT Registration Number	Textbox	Customer_VATReg is retrieved from the Customer Entity			
		Business Name	Label	Label for Business Name			

					Business Name	Textbox	is retrie	mer_ essName eved from the emer Entity	
			Update	Button	the cus	n which enables ustomer to e the their nation when d			
			Cancel	Button	None				
	Step 6: The customer		Step 7: The system information according the <u>Custome</u>	ording to the v					
	updates their information		Input Name	Attribute in Entity		Validation Requirements			
	and clicks		Title	Title_ID		• Required			
	the "Update" Button.		Name	Customer_ Name		RequiredMax 30 charactersAlphabet only			
			Surname	Customer_ Surname		RequiredMax 30 charactersAlphabet only			
			Cellphone Number	Customer_ CellPhoneNu	umber	 Required Max 10 characters Numeric characters only 			
			Email Address	Customer_ EmailAddres	SS	 Required Max 100 characters Must contain "@" and domain extension 			
			VAT Registration Number	Customer_ VATReg		Max 10 characters			

				Numeric characters only
	Business Name	Customer Businessl	_	Max 50 characters
	Step 8: The sy			stomer to confirm ion modal:
	Input Name	Control Type	Notes	
	Confirm Update	Modal Heading	None	
	Are you sure you want to update Profile?	Label		confirm the f the customer's on
	Cancel	Button	custome	icked when the r does not he update.
	Confirm	Button		icked when the r does confirm te.
Step 9: The Customer	Step 10: The save the upda as follows:			odate query to Customer Entity
clicks the "Confirm" button.	Details to be Updated	Attribute	in Table	Notes
	Customer ID	Customer	·_ID	Cannot be Updated (stays the same)
	Title	Title_ID		Can be updated
	Name	Customer	_Name	Can be updated
	Surname	Customer Surname	—	Can be updated
	Cellphone Number	Customer		Can be updated
	Email Address	Customer EmailAdd	_	Can be updated
	VAT Registration Number	Customer VATReg	-	Can be updated

		Business Name	Customer_ BusinessNa	ıme	Can be updated
		Step 11: The sy Customer's prof			
		Input Name	Control Type	Notes	
		Success	Modal Heading	None	
		Customer Profile was successfully updated.	Label	custome	indicate the er's profile ccessfully
		Ok	Button	None	

ALTERNATE COURSES:

ALT-STEP 7: The system fails to validate the updated information and displays a Validation Error Modal:

Input Name	Control Type	Notes
Validation Error	Modal Heading	None
The information entered is in the incorrect format. Please try again.	Label	None.
Ok	Button	None

ALT-STEP 9: The Customer does not confirm the update and clicks the "Cancel Button". Terminate Use Case.

ALT-STEP 11: The system fails to update the Customer Profile and displays an error modal:

Input Name	Control Type	Notes
Error	Modal Heading	None
There was an unexpected error when trying to update the customer Profile.	Label	Label to indicate the system failed to update the customer's Profile.
Ok	Button	None

CONCLUSION:	The use case concludes when the customer profile has been successfully updated.
POST-CONDITION:	The customer Profile is updated.
BUSINESS RULES:	The customer must be logged in to update their profile
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

2.1.4. EMPLOYEE SUBSYSTEM

NKAP BOLTING SYSTEM					
DATE:	13-06-2021				
VERSION:	1.0				
AUTHOR(S)	Sonali Badrinath				
USE CASE NAME:	Create Delivery Shift	USE CASE TYPE			
USE CASE ID:	4.1.	Business Requirements:			
PRIORITY:	High	System Analysis:			
SOURCE:	NKAP Bolting System Requirements	System Design: ☑			
PRIMARY BUSINESS ACTOR:	None				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	delivery shift on to the s	the event where the admin wants to add a new system. The use case begins when the admin very shift. The admin will enter all the delivery			

	NKAP B	OLTING S	YSTEM		
DATE:	03-06-21				
VERSION:	1.0				
AUTHOR(S)	Divya Bagratee				
USE CASE NAME:	Delete Profile		USE CASE	ТҮРЕ	
USE CASE ID:	3.7		Business Requirement	ts: □	
PRIORITY:	High		System Analysis:		
SOURCE:	NKAP Bolting System Requirements	n	System Design:	Ø	
PRIMARY BUSINESS ACTOR:	Customer				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	their account. The Cu bar and click the "Vie retrieve the Custome	ustomer w w Profile" r Informat The use c	ent where a customer wa ill select the User Icon on dropdown menu item. The on Customer Entity. The ase ends when the systemer's Account.	the navigation e system will e system will	
PRE-CONDITION:	The customer mu	st be logg	ed in.		
TRIGGER:	A customer wants to	delete the	ir profile.		
TYPICAL COURSE			SYSTEM RESPON	SE:	
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Action		
	Step 1: A customer wants to delete their account.				
	Step 2: The customer will select the User Icon on the navigation bar and click the "View Profile" dropdown menu item.		Step 3: The system will 3.5 View Profile.	invoke use case	
	Step 4: The customer will click the "Delete Button".		Step 5: The system will confirmation of the delet Confirmation modal:		

	Input Name Co	ntrol Notes oe				
	Confirmation Moderate	dal None ading				
	Are you sure you want to delete your account?	Label to request confirmation of deletion.				
	Confirm But	ton Button when clicked will proceed with the deletion.				
	Cancel But	ton Button when clicked will cancel the deletion.				
Step 6: The customer clicks the "confirm" button.	Step 7: The system of Delete query to delete information from the Attribute in Entity Title_ID Customer_Name	e the customer				
	Customer_Surnam					
	Customer_CellPho Customer_EmailA					
	-	Customer_VATReg				
	Customer_Busines	Customer_BusinessName				
	And the User Entity :	And the User Entity:				
	Attribute in table					
	User_ID	User_ID				
	User_Username					
	Step 8: The system modal to inform the a	admin that the				
	customer was succes	ssiully deleted.				

				C Pi	ustomer rofile was uccessfully eleted.	Modal Heading Label	Label to indicate the system successfully deleted the
							customer.
ALTERNATE				0		Button	None
ALTERNATE	ALT-STEP 6: The "Cancel" button.			_		e deletion a	and clicks the
COURSES:	ALT-STEP 8: The displays an Erro			o de	elete the Cus	tomer's ac	count and
	Input Name	Con	trol Ty _l	ре	Notes		
	Error	Mod Hea			None		
	There was an unexpected error when trying to delet the customer account.	to delete to delete to delete tomer			dicate the system failed ne customer's account.		
	Ok	Butte	on	None			
CONCLUSION:	The use case co		when t	he:	system has s	uccessfully	deleted the
POST-CONDITION:	The custome	er no lon	ger exis	sts o	on the systen	n.	
BUSINESS RULES:	1. The customer	must be	logged	d in	to delete the	ir account.	
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None						
ASSUMPTIONS:	None						
OPEN ISSUES:	None						
	shift details and use case ends v				•		
PRE-CONDITION:	The admin s	should b	e logge	d oı	nto the syste	m.	
TRIGGER:	The admin requesting to add a new delivery shift.						
TYPICAL COURSE	ACTOR			5	SYSTEM RESP	ONSE:	
OF EVENTS:	ACTION:	Manual Action	Auto	mat	ed Action		

		NKAP B	OLTING SYSTEM		
DATE:	03-06-21				
VERSION:	1.0				
AUTHOR(S)	Divya Bagratee				
USE CASE NAME:	Add User F	Role USE CASE TYPE			
USE CASE ID:	3.8		Business Requirements: □		
PRIORITY:	High		System Analysis: □		
SOURCE:	NKAP Bolti System Requireme		System Design: ☑		
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	Role On the navigation I system will provide the the User Ro	e system. bar and cl display th New Use ole in the l	bes the event where an Admin wants to add a User The admin will select the User Menu item on the lick the "Add User Role" dropdown menu item. The ne Add User Role Screen and prompt the admin to er Role Information. The system will validate and save UserRole Entity. The use case ends when the adds a User Role to the system.		
PRE-CONDITION:	The Ad	min must	be logged in to the system		
TRIGGER:	An admin w	vants to a	dd a User Role to the system.		
TYPICAL COURSE	ACTOR		SYSTEM RESPONSE:		
OF EVENTS:	ACTION:	Manual Action	Automated Action		
	Step 1: An admin wants to add a User Role on the system.				
	Step 2: The admin will		Step 3: The system will display the Add User Role Modal with the following html controls and will enable the input controls:		
	select the User Menu		Input Name Control Notes Type		

item on the	Add User Role	Heading	None
navigation bar and click the	User Role Name	Label	Label for User Role Name
"Add User Role" dropdown	User Role Name	Textbox	Enables the admin to enter the User Role Name.
menu item.	User Role Description	Label	Label for User Role Description
	User Role Description	Textbox	Enables admin to enter the User Role description.
	Add	Button	Button when clicked will proceed with the add user role process.
	Cancel	Button	Button when clicked will terminate the add user role process.
Step 4: The admin	•		the User Role details quirements of the User
provides the User Role	Input Name	Attribute in Entity	Validation Requirements
•	User Role Name		Requirements
the User Role details and clicks the "Add"	User Role	Entity	me • Required • Max 50 characters • Alphabet only
the User Role details and clicks the "Add"	User Role Name User Role Description	Entity UserRole_Na UserRole_De	me Required Max 50 characters Alphabet only scrip Required Max 200 characters Alphabet only
the User Role details and clicks the "Add"	User Role Name User Role Description	Entity UserRole_Na UserRole_De	Requirements me Required Max 50 characters Alphabet only scrip Required Max 200 characters Alphabet
the User Role details and clicks the "Add"	User Role Name User Role Description	Entity UserRole_Na UserRole_De	me Required Max 50 characters Alphabet only scrip Required Max 200 characters Alphabet only characters Alphabet only
the User Role details and clicks the "Add"	User Role Name User Role Description Step 6: The sysuser role details	Entity UserRole_Na UserRole_De stem requests and displays Control	Requirements me Required Max 50 characters Alphabet only scrip Required Max 200 characters Alphabet only confirmation of the a confirmation modal:

Step 1: The admin

		Are the User Role details correct?	Label	confi	el to request rmation of User details.
		Confirm	Button	Butto will p	on when clicked proceed with ang the User
		Cancel	Button	will c	on when clicked ancel the ng of the User
	Step 7: The admin	Step 8: The sys query to save the as follows:			SQL Insert JserRole Entity
	clicks the "confirm" button.	Details to be Saved	Attribute in Table		Notes
		User Role ID	UserRole_ID		The system generates a new UserRole_ID by retrieving the last UserRole_ID in the UserRole table and incrementing it by one.
		User Role Description	UserRole_De	SC	None
		UserRole Name	UserRole_Na	me	None
		Step 9: The system a		-	ds the User Role ss modal:
		Input Name	Control Type	Note	s
		Success	Modal Heading	None	9
		The User Role was successfully added.	Label	syste	I to indicate the m successfully d the User
		Ok	Button	None)
ALTERNATE COURSES:	ALT STEP 5: The sy displays a validation		late the User F	Role de	etails and
			Notes		

	Validation Error	Modal Heading	None
	The information entered is in the incorrect format. Please try again.	Label	None.
	Ok	Button	None
	ALT STEP 7: The	Admin clicks the '	'cancel" button. Terminate Use Case.
	ALT STEP 9: The san error modal:	system fails to sa	ve the User Role details and displays
	Input Name	Control Type	Notes
	Error	Modal Heading	None
	There was an unexpected error when trying to add the User Role.	Label	Label to indicate the system failed to Add the User Role.
	Ok	Button	None
CONCLUSION:	The Use Case con User Role.	cludes when the	system has successfully added the
POST-CONDITION:	The UserRole h	nas been added t	o the UserRole Entity.
BUSINESS	1. Only an admin c	an add a User Ro	ole
RULES:	2. The admin must	be logged in to a	idd a User Role.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None		
ASSUMPTIONS:	None		
OPEN ISSUES:	None		
	requests to add a new delivery shift. Step 2: The admin will click on the "Create	"Create	The system responds by loading the Delivery Shift" modal with the g controls:
	Delivery Shift" menu		

	NKAP B	OLTING S	SYSTEM		
DATE:	03-06-21				
VERSION:	1.0				
AUTHOR(S)	Divya Bagratee				
USE CASE NAME:	Search User Role		USE CASE TYPE		
USE CASE ID:	3.9		Business Requirements: □		
PRIORITY:	High		System Analysis: □		
SOURCE:	NKAP Bolting System Requirements		System Design: ☑		
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	Role. The admin will sel the "Search User Role" Search User Role Scree The system will retrieve	ect the Us dropdown en and pro and displa ty. The us	It where an Admin wants to search for a User ser Menu item on the navigation bar and click in menu item. The system will display the compt the admin to provide the Search Criteria. By the User Role matching the search criteria se case ends when the system has Role.		
PRE-CONDITION:	The User Role must	exist on t	he system		
TRIGGER:	An admin wants to sear	ch for a U	ser Role.		
TYPICAL COURSE			SYSTEM RESPONSE:		
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Action		
	Step 1: An admin wants to search for a user role.				
	Step 2: The admin will select the User Menu item on the Step 3: The system will display the sea user role screen with the following html components and enables the input cont				
	navigation bar and click the "Search User Role" dropdown menu		Input Control Notes Name Type		
	item.		Search Heading None User Role		
	item on the navigation bar.		ntrol Control Notes me Type		

	User Role Name	Textbox	Enak the a to er the U Role Nam	admin Iter Jser e.	
	Search	Bullon	where clicked search a material record the	ed will ch for ching rd in	
Step 4: The admin provides the search criteria and clicks the "Search" Button.	Step 5: The matching is making us	ecord in t	ne <u>Use</u>	rRole Er	ntity by
	Step 6: The matching to datagrid vice components	he search ew with th	criteria	provide	
	Input Na	me Con Type		Notes	
	Search User Rol	Mod e Hea		None	
	User Rol	e Tabl Colu		Column the data view	
	Action	Tabl Colu	e mn 2	Column the data view	
	Vertical Elipses & Pencil	Butte	ons	Buttons found in each ro the data view, we clicked allows admin to update delete to user ro	n ow of agrid then the to or the

	sele they	7: The Admin cts the User Role wish to view by ing on it.		Role"	-	stem displ	ays the "View User wing html
				Inpu	t Name	Control Type	Notes
				View Role	User	Modal Heading	None
				Go E	Back	Button	When clicked returns to the previous screen.
				User	Role	Label	Label for the UserRole_Name retrieved from the UserRole Entity.
					Role cription	Label	Label for the UserRole_Desc retrieved from the UserRole Entity.
ALTERNIATE	A 1 T	OTED A N. II					
ALTERNATE COURSES:		STEP 6: No User system displays ar					admin is found and
				al. Term			admin is found and
		system displays ar	n error moda	d. Term	ninate U		admin is found and
		system displays ar	Control Ty	d. Term	Notes None Label (matchi	se Case.	
		Input Name Error No User Role's matching the search criteria	Control Ty Modal Hea	d. Term	Notes None Label (matchi	displayed wing records	vhen no
CONCLUSION:	the	Input Name Error No User Role's matching the search criteria was found. Ok	Control Ty Modal Hea Label Button	/pe ding	Notes None Label of matchin the of	displayed wing records database.	vhen no
CONCLUSION: POST-CONDITION:	The	Input Name Error No User Role's matching the search criteria was found. Ok	Control Ty Modal Hea Label Button es when the	ype ding	Notes None Label of matching in the of the succession of the succe	displayed wing records database.	vhen no were found
CONCLUSION:	The 1. O	Input Name Error No User Role's matching the search criteria was found. Ok use case conclud	Control Ty Modal Hea Label Button es when the to view the lesearch for a	ype ding systen Jser Ro User R	Notes None Label of matching in the of successole tole.	displayed wing records database.	vhen no were found
CONCLUSION: POST-CONDITION: BUSINESS	The 1. O	Input Name Error No User Role's matching the search criteria was found. Ok use case conclud The admin is able nly an admin can he User Role mus	Control Ty Modal Hea Label Button es when the to view the lesearch for a	ype ding systen Jser Ro User R	Notes None Label of matching in the of successole tole.	displayed wing records database.	vhen no were found
CONCLUSION: POST-CONDITION: BUSINESS RULES: IMPLEMENTATION CONSTRAINTS AND	The 1. O 2. Ti	Input Name Error No User Role's matching the search criteria was found. Ok use case conclud The admin is able only an admin can the User Role must	Control Ty Modal Hea Label Button es when the to view the lesearch for a	ype ding systen Jser Ro User R	Notes None Label of matching in the of successole tole.	displayed wing records database.	vhen no were found

		NKAP B	OLTING SYSTEM
DATE:	03-06-21		
VERSION:	1.0		
AUTHOR(S)	Divya Bagı	ratee	
USE CASE NAME:	Update Us	er Role	USE CASE TYPE
USE CASE ID:	3.10		Business Requirements: □
PRIORITY:	High		System Analysis: □
SOURCE:	NKAP Bolti System Requireme		System Design: ☑
PRIMARY BUSINESS ACTOR:	Admin		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	None		
DESCRIPTION:	User Role. and click th display the	The admine "Search Update Uptails. The	bes the event where an Admin wants to update a n will select the User Menu item on the navigation bar user Role" dropdown menu item. The system will ser Role Screen and prompt the admin to provide the use case ends when the system has successfully le.
PRE-CONDITION:			be logged in to the system. ust exist on the system.
TRIGGER:	An admin w	vants to up	pdate a User Role
TYPICAL COURSE	ACTOR		SYSTEM RESPONSE:
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Action
	Step 1: The Admin wants to update a User Role Step 2: The		Step 3: The system will invoke Use Case 3.9 Search User Role.
	admin will select the User Menu item on		

	the navigation bar and click the "Search User Role" dropdown menu item. Step 4: The admin will	Step 5: The sy Role Modal with enable the inpu	h the followi		Update User components and
(click the "Update"	Input Name	Control Type	Notes	
l l	button.	Update User Role	Modal Heading	None	
		Name	Label		. Displayed extbox is d.
		Name	Textbox		s the admin to the User Role
		Description	Label		. Displayed extbox is d.
		Description	Textbox		s the admin to the User Role tion.
		Update	Button	Button when clicked proceeds with the update process.	
		Cancel	Button	termina	when clicked tes the process.
	Step 6:	Step 7: The sy	stem validat	es the U	ser Role
-	The admin		ording to the		ion requirements
t	provides the updated	Input Name	Attribute in Entity	n 	Validation Requirements
details and clicks the "Update" button.	and clicks the "Update"	User Role Name	UserRole_	Name	RequiredMax 50 charactersAlphabet only

	User Role Description	UserRole_[Required Max 200 characters Alphabet only
	Step 8: The sy user role detail			nation of the irmation modal:
	Input Name	Control Type	Notes	
	Confirm update	Modal Heading	None	
	Are you sure you want to update the User Role details.	Label		request ation of User stails.
	Confirm	Button	will prod	when clicked ceed with g of the User
	Cancel	Button	will can	when clicked cel the g of the User
Step 9: The admin clicks the	Step 10: The squery to Update Entity as follow Details to	te the User R	ole in the	
"Confirm" Button.	be Updated User Role ID	UserRole_l		Cannot be Updated (Stays the same)
	User Role Description	UserRole_	Desc	Can be updated
	User Role Name	UserRole_l	Name	Can be updated
	Step 11: The s	•		odates the User
	Input Name	Control Type	Notes	S

		Succe	ess	Modal Heading	None	
		The U Role v succe Updat	vas ssfully	Label	Label to indicate the system successfully updated the User Role.	
		Ok		Button	None	
ALTERNATE	ALT-STEP 7 : The displays a Validat			ate the Upda	ted details and	
	Input Name	Control 7	ype No	otes		
	Validation Error	Modal Heading	No	one		
	The information entered is in the incorrect format. Please try again.	Label	No	one.		
	Ok	Button	No	one		
COURSES:	ALT-STEP 9: The "Cancel" button.			irm the upda	te and clicks the	
	ALT-STEP 11: The error modal:	ne system fa	ils to upda	ate the User	Role and displays an	
	Input Name	Control T	ype No	otes		
	Error	Modal Heading	No	None		
	There was an error when trying to update the User Role.	Label		Label to indicate the system failed update the User Role.		
	Ok	Button	No	ne		
CONCLUSION:	The use case con User Role.	cludes whe	the syste	em has succ	essfully updated the	
POST-CONDITION:	The User Role	e has been ι	ıpdated.			
BUSINESS RULES:	 Only an admin The User Role 	•				
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None					
ACCUMPTIONS.						
ASSUMPTIONS:	None					

	NKAP B	OLTING S	YSTEM	
DATE:	03-06-21			
VERSION:	1.0			
AUTHOR(S)	Divya Bagratee			
USE CASE NAME:	Delete User Role		USE CASE TY	PE
USE CASE ID:	3.11		Business Requirements:	
PRIORITY:	High		System Analysis:	
SOURCE:	NKAP Bolting System Requirements	า	System Design:	☑
PRIMARY BUSINESS ACTOR:	Admin			
PRIMARY THE SYSTEM ACTOR:	None			
OTHER PARTICIPATING ACTORS:	None			
OTHER INTERESTED STAKEHOLDERS:	None			
DESCRIPTION:	User Role. The admine bar and click the "Sea	n will sele arch User e User Ro	ent where an Admin wants of the User Menu item on the Role" dropdown menu item of Screen. The use case ended the User Role.	e navigation . The system
PRE-CONDITION:	The admin must be	oe logged	in to the system.	
	The User Role mi	ust exist o	n the system.	
TRIGGER:	An admin wants to de	elete a Us		
TYPICAL COURSE OF EVENTS:	ACTOR ACTION:		SYSTEM RESPONSE	
OF EVENTS.	ACTOR ACTION:	Manual Action	Automated Action	
	Step 1: The Admin wants to delete a User Role			
	Step 2: The admin will select the User Menu item on the navigation bar and click the "Search User Role"		Step 3: The system will inv 3.9 Search User Role.	voke Use Case
	Step 4: The admin will click the "Delete" button.		Step 5: The system reque confirmation of the deletion a confirmation modal:	

	Input Name	Control Type	Notes
	Confirmation	Modal Heading	None
	Are you sure you want to delete the User Role?	Label	Label to request confirmation of deletion of of User Role.
	Confirm	Button	Button when clicked will proceed with deletion of the User Role.
	Cancel	Button	Button when clicked will cancel the deletion of the User Role.
Step 6: The admin clicks the "Confirm" Button.	Step 7: The sys Delete Query to the <u>UserRole E</u>	Delete the	
	Attribute in Er	ntity	
	UserRole_ID		
	UserRole_Nar UserRole_Des		
	Step 8: : The sy modal to inform customer was si	stem displays the admin	that the
	Input Name	Control Type	Notes
	Success	Modal Heading	None
	The User Role was successfully deleted.	Label	Label to indicate the system successfully deleted the User Role.
	Ok	Button	None

ALTERNATE	ALT STEP 6: The admin clicks the "cancel" button. Terminate Use Case.					
COURSES:	ALT STEP 8: The system fails to delete the User Role and displays an error modal.					
	Input Name	Input Name Control Type Notes				
	Error	Modal Headin	g	None		
	There was an error when trying to delete the User Role.	Label		Label to indicate to delete the Us	e the system failed er Role.	
	Ok	Button		None		
CONCLUSION:	The use case concludes when the system has successfully deleted the User Role.					
POST-CONDITION:	The User Role no longer exists on the system.					
BUSINESS RULES:	Only an admin can Delete a User Role.					
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None					
ASSUMPTIONS:	None					
OPEN ISSUES:	None					
			Create Deliver Shift	Heading Y	None	
			Start Time:	Placeholder Text	This is the Placeholder Text for the start time of the delivery shift	
			Start Time:	Dropdown	None	
			End Time:	Placeholder Text	This is the Placeholder Text for the end time	

				of the delivery shift
		End Time:	Dropdown	None
		Date:	Placeholder Text	This is the Placeholder Text for the date of the delivery shift
		Date:	Calendar	None
		Add	Button	This button is used when the admin decides to proceed with the addition of the delivery shift.
		Cancel	Button	This button is used when the admin decides to terminate the process of adding a new delivery shift.
Step 4: The admin enters the details in the respective places:				
Start TimeEnd TimeDate				
Step 5: The admin clicks on the "Add"				ys a modal for the tion of the delivery
button [Alt]		Input Name	Control Type	Notes

	Would like to proceed with the creation a new delivery shift?	you d e n of	Placeholder Text Placeholder Text	This serves as a header for the modal. This is to allow the admin to finalize the decision to submit the creation of the delivery shift with the details provided.
	Yes		Button	Proceeds to the next step (Step 7)
	No		Button	Allows user to enter delivery shift details again.
Step 7: The admin selects the "Yes" button. [Alt]	informat	ion e	stem validate ntered by the as a delivery sl	admin does not
	Input Name	Attri table	ibute in	Notes
	Start Time	attri Time End	tTime bute in e table Time	Retrieve the Time_ID if the StartTime and EndTime are atrributes
	Time		bute in e table	associated with a Time_ID. If there is a Time_ID associated with both times, then retrieve the Shift_ID

			that contains the specific Time_ID from the Shift table.
	Date	DayOfTheWeek	Retrieve the Date_ID and retrieve the Shift_ID that contains the specific Date_ID from the Shift table.
	Time_ID matched Therefore	_ID associated wi and the Shift_ID a Date_ID are not t e, the creation for shift does not alre	essociated with the the same. The requested
	save the	•	SQL insert query to tion by performing
	Time tab	le:	
	Informa to be save	in table	Details
	Time ID	Time_ID	Generated by the system by reading the last Time_ID from the Time table and adding it by one.
	from the	em saves the follo inputted times to generated Time_I	be associated

Input Name	Attribute in table	Validation requirements
Start Time	StartTime	Required
End Time	EndTime	Required

The **Time_ID** is then stored as an attribute associated with the generated **Shift_ID** in the **Shift** table.

Date table:

Information to	Attribute in table	Details
be saved:		
Date ID	Date_ID	Generated by the system by reading the last Date_ID from the Date table and adding it by one.

The system saves the following information from the inputted date to be associated with the generated Date_ID.

Input	Attribute in	Validation
Name	table	requirements
Date	DayOfTheWeek	Required

The **Date_ID** is then stored as an attribute associated with the generated **Shift_ID** in the **Shift** table.

Shift table:

Information	Attribute	Details

be saved:

		Shift ID	Shift_ID	Generated by the system by reading the last Shift_ID from the Shift table and adding it by one.
		Date ID	Date_ID	Retrieved from the Date table.
		Time ID	Time_ID	Retrieved from the Time table.
		Step 10: The delivery shift entities and continued linear transfer in the step	details in th	
			Туре	
		Success	Modal Heading	None
		Delivery Shift was successfully created.	Placeholde Text	r Placeholder Text to indicate the delivery shift information is saved.
		Ok	Button	None
ALTERNATE COURSES:	[Alt Step 5]: The admin this use case.	clicks on the "Ca	incel" button	which terminates
	[Alt Step 7]: The admin admin Return to step 4.		o" button wh	ich makes the
	[Alt Step 8]: The Shift_I the Shift_ID associated Therefore, the creation on the system.	with the matche	d Date_ID ar	e the same.
	Input Name	Control Type	Notes	

VALIDATION	Placeholder	This serves as a header for
UNSUCCESSFUL	Text	the modal.
The information	Placeholder	This is to show that the
provided in the date and	Text	delivery shift already
time already exists in a		exists with the parameter
delivery shift.		the admin entered.
BACK	Button	This button allows the
		admin the return to the
		'Create Delivery Shift'
		screen to change the
		entered information.

the delivery shift information was unsuccessful.

Input Name	Control Type	Notes
Unsuccessful	Placeholder Text	This serves as a header for the modal.
Failed to create delivery shift. Please try again later.	Placeholder Text	This is to show that the saving of the delivery shift information was unsuccessful.
ОК	Button	None

CONCLUSION:

This use case concludes when the delivery shift has been successfully added on the system.

POST-CONDITION:	The delivery shift has been added to the Shift table.
BUSINESS RULES:	Only an admin can add a new delivery shift to the system
IMPLEMENTATION	
CONSTRAINTS AND	None
SPECIFICATIONS:	
ASSUMPTIONS:	None

OPEN ISSUES: None

NKAP BOLTING SYSTEM					
DATE:	13-06-2021				
VERSION:	1.0				
AUTHOR(S)	Sonali Badrinath				
USE CASE NAME:	Search Delivery Shift	USE CASE TYPE			
USE CASE ID:	4.2	Business Requirements:			
PRIORITY:	High	System Analysis:			
SOURCE:	NKAP Bolting System Requirements	System Design: ☑			
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	for a delivery shift. Th	es the event where the admin wishes to search ne admin will enter the date. The system will ry shift based on the date the admin entered. This			

	use case concludes when the delivery shift is shown the results of the search query.					
PRE-CONDITION:	 The admin should be logged on to the system. The delivery shift should be already exist in the system. 					
TRIGGER:	The admin wishes to search for a delivery shift.					
TYPICAL COURSE	ACTOR	SYSTEM RESPONSE:				
OF EVENTS:	ACTION:	Manual Action	Automated	Action		
	Step 1: The admin would like to search for a delivery shift.					
	Step 2: The admin clicks on the "Search Delivery Shift" screen.		Step 3: The system responds by loading the "Search Delivery Shift" screen with the following controls:			
		Search elivery hift"	Control Name	Control Type	Notes	
			Search Delivery Shift	Heading	None	
			Search Date:	Placeholder Text	This is the Placeholder Text to prompt the user to select the delivery date of the delivery shift they want to search.	
			Search Date	Textbox	None	
			Search icon	Icon button	None	

Delivery Shift Table	Table	This is to show the Delivery Shifts that are saved in the database in a table.
Date	Column	This is the column for the Delivery Shift date.
Time	Column	This is the column for the delivery shift start time and end time.
Assign To	Column	This is the column is to allow the admin to assign a delivery shift to an employee. (Use Case 4.10. Assign Delivery Shift)
Options	Icon	This is to show a drop down for the options to update and delete information on each row.
Update	Button	Present in Options dropdown. Used to update the information of a delivery shift such as the date, start time and end time.
Delete	Button	Present in Options dropdown. Used to delete shift details such as date, start time, end time and employee information if it was assigned.

Step 4:		Cancel Step 5: Th	Button e system validat	Present in Options dropdown. Used to cancel a shift that was assigned to an employee.		
The admin		against the Date table [Alt] :				
inputs the		Input Name	Attribute Name	Validation Requirements		
search criteria and clicks on the search icon.	cks	Search Date	DayOfTheWeek	Required, Date format must be entered as shown in the placeholder text.		
		Step 6: A S	SOL read guery is	s used to read the		
		Step 6 : A SQL read query is used to read the following attributes based on the search criteria:				
		The system will use the DayOfTheWeek associated with the Date_ID attribute from the Date table to retrieve all shifts that contain that specific Date_ID from the Shift table: • Date_ID to retrieve and display the associated DayOfTheWeek attribute from the Date Table in the Date column.				
		•	retrieve and dis	he Time table to splay the StartTime Start Time column stribute in the End		
		Shift_ID the and display have been the request	nat match the se y the names of t n assigned to the	vill use the identified arch criteria to check he employees that delivery shifts with estem will match the		

		•		ft_ID of the shift that
			a search result	
			EmployeeShift EmployeeShift	_
		•	The system wi Employee_ID a EmployeeShift EmployeeShift	associated with that _ID in the
		•	· · · · —	me attribute from the e that is associate with
		•		_Name will be ne Assign To Column n to Assign another
		-	•	now the search results form of a table. [Alt]
ALTERNATE	Step 5: The details		_	
				ta on the requested unsuccessful modal
	to the admin.	rstern win dis	olay a validatioi	i diisuccessidi iiiodai
	Input Name		Control Type	Notes
	Error		Placeholder	This serves as a
			Text	header for the
				modal.
	No matches found		Placeholder	modal. This is to show that
	No matches found check the format of entered information	of the		modal.
	check the format	of the	Placeholder	modal. This is to show that there were no
	check the format	of the	Placeholder	modal. This is to show that there were no
CONCLUSION:	check the format of entered information	of the on.	Placeholder Text Button	modal. This is to show that there were no matches found. None
CONCLUSION: POST-CONDITION:	OK The use case conclusearched criteria.	of the on. udes when the ives the resulting the contract of the resulting the resulting the contract of	Placeholder Text Button e admin receive	modal. This is to show that there were no matches found. None
	OK The use case conclusearched criteria. The admin rece	of the on. udes when the ives the resulting the contract of the resulting the resulting the contract of	Placeholder Text Button e admin receive	modal. This is to show that there were no matches found. None

ASSUMPTIONS:	None
OPEN ISSUES:	None

	NKA	AP BOLTII	NG SYSTEM			
DATE:	13-06-2021					
VERSION:	1.0					
AUTHOR(S)	Sonali Badrinath	1				
USE CASE NAME:	Update Delivery	/ Shift	USE CASE TYPE			
USE CASE ID:	4.3.		Business Requirements:			
PRIORITY:	3		System Analysis:			
SOURCE:	NKAP Bolting Sys Requirements	System Design:				
PRIMARY BUSINESS ACTOR:	Admin					
PRIMARY THE SYSTEM ACTOR:	None					
OTHER PARTICIPATING ACTORS:	None					
OTHER INTERESTED STAKEHOLDERS:	Employee					
DESCRIPTION:	This use case describes the event in which an admin wants to update the details of a delivery shift. This use case ends when the information has been updated successfully.					
PRE-CONDITION:	The admin should be logged on to the system.					
TRIGGER:	The admin wishes update the details of the delivery shift.					
TYPICAL COURSE		Actor Action:				
OF EVENTS:		Manual Action	Automated Action			

Step 1: The admin requests to update the details of a delivery shift by selecting the "Update" option in the Option column dropdown of the line of information	Step 2: The Search Deli		es use case 4.2.
of which they want to update.	"Update Defollowing control Name Update Delivery	elivery Shift " n	nds by loading the nodal with the Notes None
	Shift Start Time: Start	Placeholder Text Dropdown	This is the Placeholder Text for the start time of the delivery shift None
	Time: End Time:	Placeholder Text	This is the Placeholder Text for the end time of the delivery shift
	End Time:	Dropdown	None

	Date:	Placeholder Text	This is the Placeholder Text for the date of the delivery shift
	Date:	Calendar	None
	Done	Button	This button is used for when the admin is done updating the information and wants to proceed to update the new information on the system.
	Cancel	Button	This button is used when the admin decides to terminate the process of adding a new delivery shift.
Step 4: The admin enters the new information and clicks on	informati	ne system valida on entered by th xist as a delivery	e admin does not
the " Done " button. [Alt]		Attribute in table	Notes
	Time	StartTime attribute in Time table	Retrieve the Time_ID if the StartTime and EndTime are
	Time	EndTime attribute in Time table	atrributes associated with a Time_ID. If there is a Time_ID associated with both times, then retrieve

					the Shift_ID that contains the specific Time_ID from the Shift table.
		Date	DayOfTheWeek	Retrieve the Date_ID and retrieve the Shift_ID that contains the specific Date_ID from the Shift table.	
			Time_ID matched Therefor	:_ID associated wi and the Shift_ID a I Date_ID are not re, the creation fo shift does not alre	associated with the the same. If the requested
					• •
			Input	Attribute in	Validation
			Name	table	requirements
			Start Time	StartTime	Required
		End Time	EndTime	Required	
			Date tab	ıle:	
			Input	Attribute in	Validation
		1			
			Name	table	requirements

Step 7: The system saves the new delivery
shift details in the relevant entities and
displays a success modal:

Input Name	Control Type	Notes
Success	Modal Heading	None
Delivery Shift information was successfully updated.	Placeholder Text	Placeholder Text to indicate the employee information is saved.
Ok	Button	None

ALTERNATE COURSES:

[Alt Step 4]: The admin clicks on the "Cancel" button which terminates this use case.

Return to **step 1**.

[Alt Step 5]: The Shift_ID associated with the matched Time_ID and the Shift_ID associated with the matched Date_ID are the same.

Therefore, the update for the requested delivery shift already exists on the system.

Input Name	Control Type	Notes
VALIDATION UNSUCCESSFUL	Placeholder Text	This serves as a header for the modal.
The information provided in the date and time already exists in a delivery shift.	Placeholder Text	This is to show that the delivery shift already exists with the parameters the admin entered.
BACK	Button	This button allows the admin the return to the 'Update Delivery Shift' screen to change the entered information.

After the admin clicks on the "Back" button the admin will Return to [Alt Step 7]: The updated delivery shift information could not be added to the system. The system will display a modal informing the admin that the saving of the delivery shift information was unsuccessful. Input Name Control Type Notes Placeholder This serves as a header for Error the modal. Text Placeholder This is to show that the Failed to update delivery shift Text saving of the delivery shift information was information. Please try unsuccessful. again later. OK Button None This use case concludes when the delivery shift information has been **CONCLUSION:** successfully updated on the system. **POST-CONDITION:** The delivery shift has been updated in the **Delivery Shift** table. 1. Only an admin can update the information of the delivery shift in **BUSINESS RULES:** the system. **IMPLEMENTATION CONSTRAINTS AND** None **SPECIFICATIONS:**

ASSUMPTIONS:

OPEN ISSUES:

None

None

		NKAF	BOLTING SYSTEM			
DATE:	13-06-2021					
VERSION:	1.0					
AUTHOR(S)	Sonali Badrina	ıth				
USE CASE NAME:	Delete Deliver Shift	ry	USE CASE TYPE			
USE CASE ID:	4.4.		Business Requirements:			
PRIORITY:	High		System Analysis:			
Source:	NKAP Bolting S Requirements	•	System Design: ☑			
PRIMARY BUSINESS ACTOR:	Admin					
PRIMARY THE SYSTEM ACTOR:	None					
OTHER PARTICIPATING ACTORS:	None					
OTHER INTERESTED STAKEHOLDERS:	None					
DESCRIPTION:	This use case describes the process where the admin wants to delete the delivery shift from the system. This use case begins when the Administrator requests to delete a delivery shift. This use case concludes when the admin gets notified that the delivery shift has been successfully removed from the system.					
Pre-condition:			be logged in to the system. hould already exist on the systems database.			
TRIGGER:	The admin requesting to delete a delivery shift.					
TYPICAL COURSE			SYSTEM RESPONSE:			
OF EVENTS:		Manua Automated Action				
	Action:	l Action				
	Step 1: The admin request to delete a	Piccion	Step 2: The system invokes Use case 4.2. Search Delivery Shift.			

	delivery shift from the system.			
Step 3: The admin requests to update the	Step 4: The syst following contro	ols:		
	details of an	Input Name	Control Type	Notes
	employee by selecting the "Delete" option in the Option column dropdown of the line of informatio	Confirm to Proceed	Placeholder Text	This serves as a header for the modal.
		Would you like to proceed with the deletion of the selected delivery shift?	Text	This is to allow the admin to finalize the decision to submit the deletion of the delivery shift.
	n of which they want to delete.	Yes	Button	Proceeds to the next step (Step 6)
		No	Button	Allows user to return to the Search Delivery Shift screen.
	Step 5: The admin clicks on the "Yes" button to	Step 6: The syst remove the deli [Alt]:		delete query to mation from the
	confirm the deletion of	From the Shift t		
	the delivery shift . [Alt]	n to e	Attribut Deta e in able	nils
		Shift ID S	-	will automatically te the:

					-	Date_ID and the DayOfTheWeek associated with the ID in the Date table. Time_ID and the StartTime, EndTime atrributes associated with the ID in the Time table. If the selected shift was assigned the EmployeeShift_I D associated with the Shift_ID will be deleted.
						deleted delivery n Delivery Shift
			Step 8: The sys	stem display	s a suc	ccess modal:
			Input Name	Control Type	l	Notes
			Success	Modal Headin	g	None
			Delivery Shift information was successfully deleted.	Placeho Text	older	Placeholder Text to indicate the delivery shift information is deleted.
			Ok	Button		None
ALTERNATE	[Alt Step 5]: Terminate th		clicks on the "	No" button.		
Courses:	[Alt Step 6]: The delivery shift could not be deleted from the system. The system will display a modal informing the admin that the deletion of the delivery shift was unsuccessful.					

	Input Name	Control Type	Notes			
	UNSUCCESSFUL	Placeholder Text	This serves as a header for the modal.			
	Failed to delete delivery shift. Please try again later.	Placeholder Text	This is to show that the deletion of the delivery shift information was unsuccessful.			
	ОК	Button	None			
Conclusion:	This use case ends when the delivery shift has been successfully deleted from the system.					
Post- condition:	The details of the delivery shift have been removed from the Delivery Shift table.					
BUSINESS RULES:	1. Only the admin can remo	ove a delivery sl	hift.			
IMPLEMENTATIO N CONSTRAINTS AND SPECIFICATIONS:	None					
Assumptions:	None					
OPEN ISSUES:	None					

	NKAP BOLTING SYSTEM						
DATE:	23-06-21						
VERSION:	1.0						
AUTHOR(S)	Sonali Badrina	ath					
USE CASE NAME:	Add Employe	e		USE CASE TYPE			
USE CASE ID:	4.5.			Business Requireme	ents:		
PRIORITY:	High			System Analysis:			
SOURCE:	NKAP Bolting	System Requirer	ments	System Design:			
PRIMARY BUSINESS ACTOR:	Administrator						
PRIMARY THE SYSTEM ACTOR:	Employee						
OTHER PARTICIPATIN G ACTORS:	None						
OTHER INTERESTED STAKEHOLDER S:	None						
DESCRIPTION:	This use case describes the event where an admin wants to add an employee on the system. The system will prompt the admin to provide the new employee information by displaying the Add Employee Screen. The system will validate the information. Once the system has successfully validated the Information, it will be captured and stored in the User Entity and the Employee Entity respectively. The use case ends when the employee has successfully added on the system.						
PRE- CONDITION:	The admin should be logged onto the system.						
TRIGGER:	The admin red	questing to add a	new emplo	yee.			
TYPICAL COURSE OF EVENTS:	ACTOR ACTION:	SYSTEM RESPONSE: Manual Automated Action Action					
	Step 1: The admin requests to						

	<u> </u>				
add a new					
employee.					
Step 2: The				s the Employee	
admin will		dropdown m	enu item		
click on the					
"Employee"					
menu item					
on the					
navigation					
bar.					
Step 4: The		Stop E: Thos	vetom rocnon	ds by loading the	
-				ds by loading the	
customer			/ee modal wi	th the following	
clicks the		controls:			
"Add					
Employee"		Control	Control	Notes	
dropdown		Name	Туре	Notes	
menu item.		Name	Турс		
		Add	Heading	None	
		Employee			
		Title:	Textbox	This is the	
		Title:	rexedex	Placeholder	
				Text for the	
				title of the	
				employee.	
		Title:	Dropdown	Populated	
			·	with different	
				titles to	
				select.	
				Jeicet.	
		Name:	Placeholder	This is the	
			Text	Placeholder	
				Text for the	
				name of the	
				employee.	
		Name:	Textbox	None	
		Surname:	Placeholder	This is the	
			Text	Placeholder	
				Text for the	
				surname of	
				the	
				employee.	
				cilipioyee.	
1	ı	1		ı	

Surname:	Textbox	None
Cell Number:	Placeholder Text	This is the Placeholder Text for the cell number of the employee
Cell Number:	Textbox	None
Address:	Placeholder Text	This is the Placeholder Text for the address of the employee.
Address:	Textbox	None
Date of Birth:	Placeholder Text	This is the Placeholder Text for the date of birth of the employee.
DOB:	Calendar	None
ID Number:	Placeholder Text	This is the Placeholder Text for the ID Number of the employee
ID Number:	Textbox	None
Next	Button	This button is used when the admin decides to proceed with the addition of the employee.

User Role	Placeholder Text	This is the Placeholder Text for the User Role to be selected
User Role	Dropdown (disabled)	Populated with all existing user roles from the database.
Username	Placeholder Text	This is the Placeholder Text for the Username of the employee
Username	Textbox (disabled)	None
Password	Placeholder Text	This is the Placeholder Text for the Password of the employee account.
Password	Textbox (disabled)	None
Confirm Password	Placeholder Text	This is the Placeholder Text for the Password Confirmation.
Confirm Password	Textbox (disabled)	None

		Add Employee	Button (disabled)	This button is used when the admin has to create the new login details for the employee.
		Cancel	Button	This button is used when the admin decides to terminate the process of adding a new employee.
Step 6: The	Step 7: The	Step 8: The s	ystem validate	s the captured
employee · ·	admin enters		formation acco	_
communicat es their	the details in		quirements of	tne <u>Employee</u>
details to	the respective	Entity:		
the admin.	places:			
		Input	Attribute in	Validation
	• Title	Name	table	requireme
	• Name			nts
	• Surname	Employee	Employee_ID	Generated
	• Cell Number	_ID		by the
	• Address			system by
	• Date of			reading the last
	Birth			Employee
	• ID Number			_ID from
				the
	The admin			Employee
	clicks on the			table and
	"Next"			adding it
	button			by one.
		Title	Title_ID	Required
		Name	Employee_Na	m Required,
			е	maximum

				of 200 characters
	Surname	Employee_Su me	ırna	Required, 10 digits
	Cell Number:	Employe_Nur er	mb	Required, maximum of 200 characters
	Address:	Employee_Ac	ddr	Required, maximum of 200 characters
	Date of Birth:	Employee_D0	ЭB	Required, Date
	ID Numb er:	Employee_ID mber	Nu	Required, 13 digits
		ystem enables	and	disables the
	following cor	ntrols:		
	following cor Control Name	Control Type	Not	es
	Control	Control	Non	
	Control Name	Control Type	Non This Place Text title	
	Control Name Add Employee	Control Type Heading	Non This Place Text title emp	is the seholder to for the poloyee. ulated to different is to

Name:	Textbox (disable)	None
Surname:	Placeholder Text	This is the Placeholder Text for the surname of the employee
Surname:	Textbox (disable)	None
Cell Number:	Placeholder Text	This is the Placeholder Text for the cell number of the employee
Cell Number:	Textbox (disable)	None
Address:	Placeholder Text	This is the Placeholder Text for the address of the employee.
Address:	Textbox (disable)	None
Date of Birth:	Placeholder Text	This is the Placeholder Text for the date of birth of the employee.
DOB:	Calendar (disable)	None
ID Number:	Placeholder Text	This is the Placeholder Text for the ID Number of the employee

	ID Number:	Textbox (disable)	None
	Next	Button (disable)	This button is used when the admin decides to proceed with the addition of the employee.
	User Role	Placeholder Text	This is the Placeholder Text for the User Role to be selected
	User Role	Dropdown (enabled)	Populated with all existing user roles from the database.
	Username	Placeholder Text	This is the Placeholder Text for the Username of the employee
	Username	Textbox (enabled)	None
	Password	Placeholder Text	This is the Placeholder Text for the Password of the employee account.
	Password	Textbox (enabled)	None

	Confirm Password	Placeholder Text	This is the Placeholder Text for the Password Confirmation.
	Confirm Password	Textbox (enabled)	None
	Add Employee	Button (enabled)	This button is used when the admin has to create the new login details for the employee.
	Cancel	Button	This button is used when the admin decides to terminate the process of adding a new employee.
Step 10: The admin enters the	information	system validate according to the s of the User E	
temporary login details for the	Name	Attribute in Entity	Validation Requirements
employee by providing the following details: • Usern ame • Passw ord		User_Userna me	 Required Max 50 characters Alphanume ric characters only
• Confir m		User_Passwo rd	Required

	Passw ord User Role The admin clicks on the "Add Employee" button.			 Max 50 characters Alphabet characters only
		admin to conf employee info	irm the addition:	
		Input Name	Control Type	Notes
		Confirm to Proceed	Placeholder Text	This serves as a header for the modal.
		Would you like to proceed with the addition of a new employee?	Placeholder Text	This is to allow the admin to finalize the decision to submit the addition of the employee with the details provided.
		Yes	Button	Proceeds to the next step (Step 14)
		No	Button	Allows user to enter employee details again.
	Step 13: The admin selects the "Yes" button. [Alt]		ew employee i	SQL insert query nformation in

Input Name	Attribute in table	Validation requireme nts
Employee _ID	Employee_ID	Generated by the system by reading the last Employee _ID from the Employee table and adding it by one.
Title	Title_ID	Required
Name	Employee_Nam e	Required, maximum of 200 characters
Surname	Employee_Surna me	Required, 10 digits
Cell Number:	Employe_Numb er	Required, maximum of 200 characters
Address:	Employee_Addr ess	Required, maximum of 200 characters
Date of Birth:	Employee_DOB	Required, Date
ID Numb er:	Employee_IDNu mber	Required, 13 digits
Details to be saved	Attribute in No	tes

			User ID	User_ID	The system generates a new User_ID by retrieving the last User_ID in the User table and incrementing it by one.
			Username	User_ Username	None
			Password	User_ Password	None
			UserRole ID	UserRole_ID	None
			=		ne new vant entities and
			Input Name	Control Type	Notes
			Success	Modal Heading	None
			Employee information was successfully created.		Placeholder Text to indicate the employee information is saved.
			Ok	Button	None
ALTERNATE COURSES:	[Alt Step 13]: Return to step	The admin clicks o 7.	on the "No" k	outton which m	akes the admin
		d 11: The system Validation Unsu		· ·	d information
	Input Name		Control I Type	Notes	

VALIDATION UNSUCCESSFUL	Placeholder Text	This serves as a header for the modal.	
The information provided in the given parameters is not in the correct format.	Placeholder Text	This is to show that the employee information was incorrectly entered.	
BACK	Button	This button allows the admin the return to the 'Add Employee' screen to change the entered information.	

[Alt Step 15]: The new employee could not be added to the system. The system will display a modal informing the admin that the saving of the employee information was unsuccessful.

Input Name	Control Type	Notes
Unsuccessful	Placeholder Text	This serves as a header for the modal.
Failed to create employee information. Please try again later.	Placeholder Text	This is to show that the saving of the employee information was unsuccessful.
ОК	Button	None

This use case concludes when the employee has been successfully added on the system.

CONCLUSION:	The employee has been added to the Employee table and <u>User</u> table.
POST-	Only an admin can add a new employee to the system
CONDITION:	The employee must reset their password after the use case.
BUSINESS RULES:	None
IMPLEMENTA TION CONSTRAINTS	None

AND	
SPECIFICATIO	
NS:	
ASSUMPTION	None
S:	None
OPEN ISSUES:	

NKAP BOLTING SYSTEM					
DATE:	13-06-2021				
VERSION:	1.0	1.0			
AUTHOR(S)	Sonali Badrinath				
USE CASE NAME:	Search Employee	USE CASE TYPE			
USE CASE ID:	4.6.	Business Requirements:			
PRIORITY:	High	System Analysis:			

SOURCE:	NKAP Boltin	g				
SOURCE.	System	0	System Design: ☑			
	Requiremen	ts				
PRIMARY BUSINESS ACTOR:	Admin					
PRIMARY THE SYSTEM ACTOR:	None					
OTHER PARTICIPATING ACTORS:	None					
OTHER INTERESTED STAKEHOLDERS:	None					
DESCRIPTION:	an employee's employee's employee b	This use case describes the event where the admin wishes to search for an employee. The admin will enter the search parameters like the employee's name and surname. The system will search for the employee based on the parameters the admin entered. This use case concludes when the employee is shown the results of the search query.				
PRE-CONDITION:	The adm	in should	be logged on to the system.			
	The emp	loyee sho	ould be already exist in the system.			
TRIGGER:	The admin v	vishes to s	search for a employee.			
TYPICAL COURSE			SYSTEM RESPONSE:			
OF EVENTS:	ACTOR	Manua	Automated Action			
	ACTION:	1				
		Action				
	Step 1: The admin would like to search for an employee.					
	Step 2: The admin clicks on the "Search Employee " screen.		Step 3: The system responds by loading the "Search Employee" screen with the following controls: Control Control Notes Name Type			

Search Employee	Heading	None
Name:	Placeholder Text	This is the Placeholder Text to prompt the user to enter a search query.
Search Name	Textbox	None
Surname:	Placeholder Text	This is the Placeholder Text to prompt the user to enter a search query.
Search Surname	Textbox	None
Search Icon	Icon	Used to give consent to perform the search.
Employee Table	Table	This is to show the Employees that are saved in the database in a table.
ID Number	Column	This is the column for the Employee ID Number
Title	Column	This is the column for the employee title.
Fullname	Column	This is the column for the employee name and surname.
Date of Birth	Column	This is the column for the employee date of birth

	Address	Column		s the column e employee	
	Number	Column		s the column e employee er	
	Options	Icon	drop option	nation on	
	Update	Button	dropo updat	nation of an	
	Delete	Button	dropo	nt in Options lown. Used to e shift details.	
Step 4:	Step 5: The system captures and validates the information against the Employee table [Alt] :				
The admin					
inputs the	Input	Attribute Nam	е	Validation	
search	Name			Requirment	
criteria and clicks				S	
on the	Search	Employee Nai	me	Maximum	
search	Name:			of 50	
button.				characters	
	Search	Employee_Sur	nam	Maximum	
	Surname	е		of 50	
	:			characters	
	following at	L read query is tributes from the e search criteria	ne Emp		
	Employee table:				
	' '				

CONCLUSION: POST-CONDITION:	searched criteria.The admin receives the result searched criteria	ts of the employ	yees with the same	
CONCLUSION:	searched criteria.			
	The use case concludes when the admin receives the results of the searched criteria.			
	OK The second of the second	Button	None	
	No matches found. Please check the format of the entered information.	Placeholder Text	This is to show that there were no matches found.	
	Error	Placeholder Text	This serves as a header for the modal.	
	Input Name	Control Type	Notes	
ALTERNATE	Step 5: The details entered were validation failed or there are not information. The system will disp the admin.	not in the right any existing da	ta on the requested	
			now the search results	
	•	–	B attribute information ne DOB column	
		· · · -	dress attribute isplayed in the Address	
	•	· · · · —	mber attribute isplayed in the Number	
	•	· · · · —	rname attribute isplayed in the Surname	
	•	Employee_Na information di column	me attribute isplayed in the Name	
	•	_	ute information ne Title column	
		Number colun	Number in the ID nn	

IMPLEMENTATIO N CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

		NKAP BO	OLTING SYSTEM
DATE:	13-06-2021		
VERSION:	1.0		
AUTHOR(S)	Sonali Badrina	ath	
USE CASE NAME:	Update Empl	oyee	USE CASE TYPE
USE CASE ID:	4.7.		Business Requirements:
PRIORITY:	3		System Analysis:
SOURCE:	NKAP Bolting Requirements	=	System Design: ☑
PRIMARY BUSINESS ACTOR:	Admin		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	Employee		
DESCRIPTION:	the details of information t	an emplo hat was a ion. This	s the event in which an admin wants to update oyee. The system will display the previous added before and then the admin will add the use case ends when the information has been
PRE-CONDITION:	The admir	n should	be logged on to the system.
TRIGGER:	The admin wi	shes upd	ate the details of the employee.
TYPICAL COURSE			Actor Action:
OF EVENTS:	Actor Action:	Manua I Action	Automated Action
	Step 1: The admin requests to update the details of an employee by selecting		Step 2: The system invokes use case 4.6. Search Employee.

the "Update" option in the Option column dropdown of the line of informatio n of which they want to update.			
			s by loading the with the following
	Control Name	Control Type	Notes
	Update Employee	Heading	None
	Title:	Textbox	This is the Placeholder Text for the title of the employee.
	Title:	Dropdown	Populated with different titles to select.
	Name:	Placeholder Text	This is the Placeholder Text for the start time of the delivery shift
	Name:	Textbox	None
	Surname:	Placeholder Text	This is the Placeholder Text for the end time of

		the delivery shift
Surname:	Textbox	None
Name:	Placeholder Text	This is the Placeholder Text for the start time of the delivery shift
Name:	Textbox	None
Cell Number:	Placeholder Text	This is the Placeholder Text for the cell number of the employee
Cell Number:	Textbox	None
Address:	Placeholder Text	This is the Placeholder Text for the address of the employee.
Address:	Textbox	None
Date of Birth:	Placeholder Text	This is the Placeholder Text for the date of birth of the employee.
DOB:	Calendar	None
ID Number:	Placeholder Text	This is the Placeholder Text for the ID Number of the employee

		ID Number:	Textbox	None	9	
		Done	Button	used wher admi done upda infor and v proce upda new	n the in is ting the mation wants to eed to te the mation he	
		Cancel	Button	This I used the a decid term proce addir	button is when idmin des to inate the ess of ng a new loyee.	
	Step 4: The admin enters the new informatio	informatio	e system captures n entered by the cable [Alt]:			
n and clicks on the	n and clicks	Input Name	Attribute in tabl	le	Validation requirem s	
	button. [Alt]	Title	Title_ID		Required	
	[rait]	Name	Employee_Nam	e	Required, maximum 200 character	n of
		Surnam e	Employee_Surn	ame	Required digits	, 10

Cell Number :	Employe_Number	Required, maximum of 200 characters
Address :	Employee_Address	Required, maximum of 200 characters
Date of Birth:	Employee_DOB	Required, Date
ID Numb er:	Employee_IDNumb er	Required, 13 digits
	e system uses a SQL ins captured information table [Alt]:	
Information to be saved:		Details
Title	Title_ID	Match the Title_ID of the selected Title_Des c
Name	Employee_Name	None

Surname	Employee_Surname	None
Cell Number:	Employe_Number	None
Address:	Employee_Address	None
Date of Birth:	Employee_DOB	None
ID Numb er:	Employee_IDNumbe r	None

Step 7: The system saves the new employee
details in the relevant entities and displays a
success modal.

Input Name	Control Type	Notes
Success	Modal Heading	None
Employee information was successfully updated.	Placeholder Text	Placeholder Text to indicate the employee information is saved.
Ok	Button	None

ALTERNATE COURSES:

[Alt Step 4]: The admin clicks on the "Cancel" button which terminates this use case.

Return to step 1.

[Alt Step 5]: The details entered were not in the right format, so the validation failed. The system will display a validation unsuccessful modal to the admin.

Input Name	Control Type	Notes
VALIDATION UNSUCCESSFUL	Placeholder Text	This serves as a header for the modal.
The information provided in the given parameters is not in the correct format.	Placeholder Text	This is to show that the employee information was incorrectly entered.
BACK	Button	This button allows the admin the return to the 'Update Employee' screen to edit the entered information in the correct format.

After the admin clicks on the "**Back**" button the system will show the admin where the validation errors occurred by showing a red outline where the validation failed.

	[Alt Step 6]: The updated employee information could not be added to the system. The system will display a modal informing the admin that the saving of the employee information was unsuccessful.						
	Input Name	Notes					
	UNSUCCESSFUL	Placeholder Text	This serves as a header for the modal.				
	Failed to update employee information. Please try again later.	Placeholder Text	This is to show that the saving of the employee information was unsuccessful.				
	ОК	Button	None				
CONCLUSION:	This use case concludes when the employee information has been successfully updated on the system.						
POST- CONDITION:	The employee has been updated in the Employee table.						
BUSINESS RULES:	1. Only an admin can update the information of the employee in the system.						
IMPLEMENTATIO N CONSTRAINTS AND	None						

SPECIFICATIONS:

ASSUMPTIONS:

OPEN ISSUES:

None

None

NKAP BOLTING SYSTEM					
DATE:	13-06-2021				
VERSION:	1.0				
AUTHOR(S)	Sonali Badrinath				
USE CASE NAME:	Delete Employee	Use case type			
USE CASE ID:	4.8.	Business Requirements:			

PRIORITY:	High		System Analysis:				
Source:	NKAP Bolting Requirement		System Design:	\square			
PRIMARY BUSINESS ACTOR:	Admin						
PRIMARY THE SYSTEM ACTOR:	None						
OTHER PARTICIPATING ACTORS:	None						
OTHER INTERESTED STAKEHOLDERS:	None						
DESCRIPTION:	This use case describes the process where the admin wants to delete the employee from the system. This use case begins when the Administrator requests to delete a employee. This use case concludes when the admin gets notified that the employee has been successfully removed from the system.						
PRE-CONDITION:	 The admin should be logged in to the system. The employee should already exist on the systems database. 						
TRIGGER:	The admin requesting to delete an employee.						
TYPICAL			SYSTE	M RESPONSE:			
COURSE	ACTOR ACTION:	Manua	Automated Action				
OF EVENTS:	ACTION.	Action					
	Step 1: The admin request to delete a employee from the system.		Step 2: The system Employee.	invokes Use	case 4.6. Search		
	Step 3: The admin requests to update the details of an		Step 4: The system displays a modal, with the following controls: Input Name Control Notes Type				

employee by selecting the "Delete" option in the Option column dropdown of the line of informatio	Confirm to Proceed		Placehol Text	der	This serves as a header for the modal.		
		, ,		Placeholder Text		This is to allow the admin to finalize the decision to submit the deletion of the employee.	
	n of which they want to delete.		Yes B		Button		Proceeds to the next step (Step 6)
			No		Button		Allows user to return to the Search Employee screen.
	Step 5: The admin clicks on		Step 6: The system uses a SQL delete query to remove the employee information from the Employee table [Alt]:				
the "Yes" button to confirm the deletion of the employee. [Alt]		Informatio Attribute in Details n to table be deleted:					
		Employee ID	Emp D	delet attrik Title_ Empl Empl r, Em Empl Empl of the		ete all associated ibutes such as the e_Desc, bloyee_Name, bloyee_Surname, bloyee_CellNumbe mployee_DOB and bloyee_IDNumber ne selected bloyee.	

		Step 7: The system removes the deleted employee from the table in the Search Employee screen.			
		Step 8: The system displays a success modal:			
		Inj	put Name	Control Type	Notes
		Su	ccess	Modal Heading	None
		inf wa su	nployee formation as ccessfully eleted.	Placeholder Text	Placeholder Text to indicate the employee information is deleted.
		Ok	(Button	None
ALTERNATE	[Alt Step 5]: The admin clicks on the "No" button. Terminate this use case.				
COURSES:	[Alt Step 8]: The employee could not be deleted from the system. The system will display a modal informing the admin that the deletion of the employee was unsuccessful.				-
	Input Name		Control Type	Notes	
	UNSUCCESSFUL		Placeholder Text	This serves a modal.	s a header for the
	Failed to delete employee. Please try again later.				he employee was unsuccessful.
	OK Button None				
Conclusion:	This use case ends when the employee has been successfully deleted from the system.				
POST- CONDITION:	The details of the employee have been removed from the Employee table.				
BUSINESS RULES:	1. Only the admin can	1. Only the admin can remove a employee.			
IMPLEMENTATIO N CONSTRAINTS	None				

AND	
SPECIFICATIONS:	
Assumptions:	None
OPEN ISSUES:	None

	N	IKAP BOI	LTING SYSTEM	
DATE:	13-06-2021			
VERSION:	1.0			
AUTHOR(S)	Sonali Badrinath			
USE CASE NAME:	Assign Order for Delivery/Courier		USE CASE TYPE	
USE CASE ID:	4.9.		Business Requirements:	
PRIORITY:	High		System Analysis:	
SOURCE:	NKAP Bolting Syst Requirements	em	System Design: ☑	
PRIMARY BUSINESS ACTOR:	Administrator			
PRIMARY THE SYSTEM ACTOR:	None			
OTHER PARTICIPATIN G ACTORS:	None			
OTHER INTERESTED STAKEHOLDER S:	CustomerEmployeeCourier			
DESCRIPTION:	This use case describes the event where the admin wants to assign an order to be delivered to the customer. The use case begins when the admin requests to assign an order for a delivery or courier depending on the distance. Details of the assigned delivery will be stored in the Delivery table. The use case ends when the been successfully assigned.			
PRE- CONDITION:	The admin must be logged in.			
TRIGGER:	The admin wants to assign an order to be delivered to the customer.			
TYPICAL			SYSTEM RESPONSE:	
COURSE OF EVENTS:	ACTOR ACTION:	Manu al Actio n	Automated Action	
	Step 1: The admin wants to assign an order			

to be delivered			
to the customer.			
Step 2: The admin will click on the "Assign	Step 3: The syst following:	em respo	onds by loading the
Order Delivery"	Screen with the	following	g controls:
sub menu item of the 'Admin'	Control	Contr	Notes
on the	Name	ol	
navigation bar.		Type	
	Assign Order Delivery	Headi ng	None
	Unscheduled Deliveries	Table	None
	Sale ID	Colum n	Only sales that have:
			 Have a Collection_St atus of 'Available' Have a Sale_Receive Type of 'Delivery' Have a Sale_Assign attribute that is 'False'
	Customer Name	Colum n	None
	Address	Colum n	None
	Distance	Colum n	None
	Delivery/Cou rier	Colum n	This each column cell contains both Delivery and Courier buttons in each row.
	Delivery	Butto n	This button is used when the admin wants to assign an

				order for a local delivery.
		Courier	n ,	This button is used when the admin wants to assign an order for courier.
the	9 4: Based on distance in Distance		tem respon	ids by loading the following controls:
colu adm dete	ımn the	Control Name Courier Courier	Control Type Heading Placehold	
be couradment the butter the	delivered via rier. The nin clicks on "Courier" ton that is in row of ch the admin	Name	Text	Placeholder Text for the searching the courier's name of which to assign the order
wan	nts to assign order, that is	Search Courier Name	Textbox	
"De er" (the "Un	livery/Couri column of scheduled veries" table	Courier Name:	Placehold Text	er This is the Placeholder Text for the name of the courier
[ALT		Search Icon Instruction Text	Icon Butto	This informs the user as to how to assign the order. The label Text will read "Click on email link to compose courier request"
		Sale ID Sale ID	Label Textbox (disabled	

Delivery Show the search results of Couriers that are saved in the database. Name Column This is the column for the Courier name Type Column This is the column for the courier type Number Column This is the column for the number of the courier type This is the column for the number of the courier type Email Column This is the column for the mail of the courier. The email links are functional. This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning an order to	Courier	Table	This is to	
results of Couriers that are saved in the database. Name Column This is the column for the Courier name Type Column This is the column for the courier type Number Column This is the column for the number of the courier Email Column This is the column for the number of the courier The email of the courier. The email links are functional. Done Button Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning	Delivery		show the	
Name Column This is the column for the mail links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning to column the column for the order.			search	
Name Column This is the column for the Courier name Type Column This is the column for the courier type Number Column This is the column for the number of the number of the courier Email Column This is the column for the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning			results of	
Name Column This is the column for the Courier name Type Column This is the column for the courier type Number Column This is the column for the courier type Number Column This is the column for the number of the courier Email Column This is the column for the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning			Couriers	
Name Column This is the column for the Courier name Type Column This is the column for the courier type Number Column This is the column for the number of the courier the number of the courier the number of the mail for the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning				
Name Column This is the column for the Courier name Type Column This is the column for the courier type Number Column This is the column for the number of the courier Email Column This is the column for the enail of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning				
Type Column This is the column for the Courier name Type Column This is the column for the courier type Number Column This is the column for the number of the courier Email Column This is the column for the email of the email of the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning				
Type Column This is the column for the courier type Number Column This is the column for the number of the courier The email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning	Name	Column		
Type Column This is the column for the courier type Number Column This is the column for the number of the courier This is the column for the number of the courier This is the column for the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning				
Type Column This is the column for the courier type Number Column This is the column for the number of the courier Email Column This is the column for the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button In button is used when the admin decides to terminate the process of assigning				
Number Column Number Column This is the column for the number of the courier Email Column Email Column This is the column for the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning				
Number Column This is the column for the number of the courier Email Column This is the column for the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning	Туре	Column		
Number Column This is the column for the number of the courier. Email Column Email Column This is the column for the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning				
Number Column This is the column for the number of the courier Email Column This is the column for the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning				
column for the number of the courier Email Column This is the column for the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning			• •	
Email Column This is the column for the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning	Number	Column		
Email Column This is the column for the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning				
Email Column This is the column for the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning				
Email Column This is the column for the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning				
column for the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning	F 11	Callana		
the email of the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning	Email	Column		
the courier. The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning				
The email links are functional. Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning				
Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning				
Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning				
Done Button This button is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning				
is clicked when the admin is done assigning the order. Cancel Button This button is used when the admin decides to terminate the process of assigning	Dono	Rutton		
Cancel Button This button is used when the admin decides to terminate the process of assigning	Done	Button		
Cancel Button This button is used when the admin decides to terminate the process of assigning				
Cancel Button This button is used when the admin decides to terminate the process of assigning				
Cancel Button This button is used when the admin decides to terminate the process of assigning				
Cancel Button This button is used when the admin decides to terminate the process of assigning				
Cancel Button This button is used when the admin decides to terminate the process of assigning				
is used when the admin decides to terminate the process of assigning	Cancel	Button		
when the admin decides to terminate the process of assigning	G 0.11 G 0.	20.000		
admin decides to terminate the process of assigning				
decides to terminate the process of assigning				
terminate the process of assigning				
of assigning				
of assigning			the process	
a courier.			a courier.	

Step 6: Admin wants to search for the courier. [ALT] Step 8: The	Step 7: The sys	r. [ALT]		ie.
admin clicks on the email address of the courier they want to assign the order to. [ALT]	Step 9: The system directs the admin to the web browser whereby the admin can compose the email for the order and the email address of the courier will be automatically filled in the "To:" section. [ALT]			pose ess
Step 10: The admin will enter the delivery information for the order in the email to the courier. [ALT]				
Step 11: Once, done the admin sends the email. The admin navigates back to the "Courier" modal. [ALT]				
Step 12: The admin clicks on 'Done' button. [ALT]	Step 13: The s for the admin been assigned	to confirm tha		
[Input Name	Control Type	Notes	
	Attention Clicking on the Done	Placeholder Text Placeholder Text	This serves as a header for the modal. This is to remind the	
	button will update the order information		admin that by clicking the Done button that the	

	to be assigned.		order will be saved on the system as being assigned.	
	Ok	Button	Proceeds to the next step (Step 15)	
Step 14: The admin clicks on the OK button.	Back Step 15: The sysave the captu		Return the admin to the Courier Screen and allow the user to assign an order to a courier or use the Cancel button to terminate the courier assignment process.	<i>r</i> to
[ALT]	Delivery table			
	Information to be saved:	Attribute in table	Details	
	Delivery ID Courier ID	Delivery_ID Courier ID	Generated by the system by reading the last Delivery_ID from the Delivery table and adding it by one.	,

Sale table		as w se Cc St Cc w Do	ourier_ID ssociated ith the elected ourier_Email. sore the ourier_ID ith the elivery _ID nat was enerated.
Information		Attribute in	n Details
be saved:		table	
Sale_Assign=e Courier Emai	Tru	Sale_Assign Courier_Em	Sale_ID to match the Sale_Assig n attribute and update informatio n to true. This will mean that the sale is recorded as assigned a None
assignment de modal: [ALT]			
Input Name		ontrol Type	Notes
Success		odal Heading	
Saving of assigned order information has been successful.	Pla	ceholder Tex	t Placeholder Text to indicate the order has been updated to be assigned

			on the system.
			system.
	Ok	Button	None

ALTERNATE COURSES:

[Alt Step 4(delivery option)]: Based on the distance in the Distance column the admin determines that the order must be delivered locally. The admin clicks on the "Delivery" button that is in the row of which the admin wants to assign the order, that is in the "Delivery/Courier" column of the "Unscheduled Deliveries" table.

[Alt Step 5 (delivery option)]: The system responds by loading the "Delivery" modal with the following controls:

Control	Control	Notes
Name	Type	
Delivery	Heading	None
Select	Placeholder	This is the
Employee	Text	Placeholder Text
		to allows
		selection of an
		employee that
		will be assigned
		to the order.
Employee	Dropdown	Populated
		selecting all the
		Employee_Name
		data from the
-	5	Employee table.
Done	Button	This button is clicked when the
		admin is done
		assigning the
		order.
Cancel	Button	This button is
Caricei	Button	used when the
		admin decides
		to terminate the
		process of
		assigning an
		order to an
		employee.
		Returns user to

	"Assign Order
	For Delivery"
	screen

[Alt Step 6 (courier option)]: The admin selects the "Cancel" button. Return to Step 3.

[ALT Step 6a (delivery option)]: The admin selects the employee from the Employee dropdown to assign the delivery of the order. Then clicks on the 'Done' button. Proceed to ALT Step 7(delivery option)

[ALT Step 6b (delivery option)]: The admin clicks on the 'Cancel' button. Return to Step 5.

[ALT Step 7(delivery option)]: The system displays a modal for the admin to confirm the assigning of the order to an employee:

Input Name	Control Type	Notes
Confirm to Proceed	Placeholder Text	This serves as a header for the modal.
Would you like to proceed with the assigning of this order to this employee?	Placeholder Text	This is to allow the admin to finalize the decision to submit the assigning of the order to an employee.
Yes	Button	Proceeds to the save information (ALT Step 8(delivery option))
No	Button	Allows user to change the selected employee name again.

[ALT Step 8a (delivery option)]: The admin selects the "Yes" button. Proceed to ALT Step 9(delivery option)

[ALT Step 8b (delivery option)]: The admin selects the "No" button. Return to Step 6.

[ALT Step 9 (delivery option)]: The system uses a SQL insert query to save the captured information in the:

Delivery table

Information to be saved:	Attribute in table	Details
Delivery ID	Delivery_ID	Generated by the system by reading the last Delivery_ID from the Delivery table and adding it by one.

Sale table

Information to be saved:	Attribute in table	Details
Sale_Assign=True	Sale_Assign	Use the Sale_ID to match the Sale_Assign attribute and update information to true. This will mean that the sale is recorded as assigned

Employee table

Input Name	Attribute in table	Details
Employee Name	Employee_Name	Identify the Employee_ID of the selected Employee_Name.

EmployeeShift table

Information to be	Attribute in	Details
saved:	table	
Delivery ID	Delivery_ID	The system will
		look for

EmployeeShift_ID that has the selected Employee_ID to check if the selected employee has a delivery shift. The system will check if the delivery shift date is on the date of which the employee is assigned to do the delivery. The system will check that the Shift_ID in the EmployeeShift table has a Date_ID that has a DayOfTheWeek attribute that is from the current day and onwards. The system saves the Delivery_ID with the Employee ID.

[ALT Step 10a (delivery option)]: The system saves the new delivery assignment details and displays a success modal:

Input Name	Control Type	Notes
Success	Modal Heading	None
Saving of assigned order information has been successful.	Placeholder Text	Placeholder Text to indicate the order has been updated to be assigned to the employee on the system.
Ok	Button	None

[ALT Step 10b (delivery option)]: The order could not be assigned for delivery on the system. The system will display a modal:

Input Name	Control Type	Notes
Error	Placeholder Text	This serves as a header for the modal.
There was an error while saving the assigning information. Please try again later.	Placeholder Text	This is to show that the saving of the assignment information was unsuccessful.
ОК	Button	None

[ALT Step 10c (delivery option)]: The system could not find an assigned delivery shift to the selected employee that is on the current date or a date onwards from the current date. The employee cannot be assigned to the delivery of the order. The system will display the following modal:

Input Name	Control Type	Notes
Error	Placeholder Text	This serves as a header for the modal.
Employee cannot be assigned to deliver order as they are not assigned to an appropriate shift.	Placeholder Text	This is to show that the order cannot be assigned to deliver the order as they are not assigned to a shift within the required date parameter.
ОК	Button	None

[Alt Step 16(courier option)]: The order could not be assigned for delivery on the system. The system will display a modal:

Input Name	Control Type	Notes
Error	Placeholder Text	This serves as a header for the modal.

	There was an error while	Placeholder	This is to show that the	
	saving the assigning	Text	saving of the assignment	
	information. Please try		information was	
	again later.		unsuccessful.	
		_		
	OK	Button	None	
CONCLUSION:	This use case concludes when on the system.	the employee	has been successfully added	
POST- CONDITION:	The employee has been added to the Employee table.			
BUSINESS RULES:	Only an admin can add a new employee to the system			
IMPLEMENTAT ION CONSTRAINTS AND SPECIFICATION S:	None			
ASSUMPTIONS :	None			
OPEN ISSUES:	None			

NKAP BOLTING SYSTEM			
DATE:	13-06-2021		
VERSION:	1.0		
AUTHOR(S)	Sonali Badrinath		
USE CASE NAME:	Assign Delivery Shift	USE CASE TYPE	
USE CASE ID:	4.10.	Business Requirements:	
PRIORITY:	High	System Analysis:	
SOURCE:	NKAP Bolting System Requirements	System Design: ☑	
PRIMARY BUSINESS ACTOR:	Admin		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS :	Employee		
DESCRIPTION:	An Admin will be able to view the specific shift and assign it to an employee. The system will update the shift to be assigned to the employee on the shift schedule.		
PRE- CONDITION:	The admin should be on the Search Delivery Shift screen		
TRIGGER:	The admin requesting to assign a delivery shift.		
		SYSTEM RESPONSE:	

		Manu	Automated Action				
		al	Natomated Netion				
		Action					
	Step 1: The admin wants to assign a delivery shift to an employee		Step 2: The system invokes use case 4.2. Search Delivery Shift.				
	Step 3: The admin			Step 4: The system responds by loading the " Assign Delivery Shift " modal with the following controls:			
	will click		Control	Control	Notes		
	on the "Assign"		Name	Туре	Notes		
	button present in the Assign To column of the Delivery		Assign Shift	Heading	None		
			Select Employee	Placeholder Text	None		
		Delivery	Delivery			Employee Name	Dropdown
	the Search Delivery Shift Screen.	e arch livery ift	Proceed	Button	This button is used for when the admin is done selecting the employee name of which to assign the shift.		
			Cancel	Button	This button is used when the admin decides to terminate the process of adding a new employee.		
	Step 5: The admin selects the						

	employee name.			
	Step 6: The admin clicks on	Step 7: The system following controls:		dal, with the
	the " Proceed	Input Name	Control Type	Notes
	" button [Alt]	Confirm to Proceed	Placeholder Text	This serves as a header for the modal.
		Would you like to proceed with assigning this employee to the selected delivery shift?	Placeholder Text	This is to allow the admin to finalize the decision to submit the assigning of the delivery shift.
		Yes	Button	Proceeds to the next step (Step 8)
		No	Button	Allows user to return to the Assign Shift screen.
	Step 8: The admin clicks on	Step 9: The system the captured inform [Alt]:		ert query to save mployeeShift table
	the " Yes " button to confirm	Information to be saved:	Attribute in table	Details
the assigning of the delivery shift. [Alt]	EmployeeShift _ID	EmployeeShift _ID	Generated by the system by reading the last EmployeeShift	

		_ID from the EmployeeShift table and adding it by one.
Employee ID	Employee_ID	The system will retrieve the Employee_ID associated with the Employee_Na me that was selected and save it as an attribute in the EmployeeShift table with the chosen Shift's ID.

Step 10: The system displays a success modal:

Input Name	Control Type	Notes
Success	Modal Heading	None
Delivery shift was successfully assigned.	Placeholder Text	Placeholder Text to indicate the delivery shift is assigned to the employee and saved on the system.
Ok	Button	None

ALTERNATE COURSES:

[Alt Step 6]: The admin clicks on the "Cancel" button which terminates this use case.

Return to step 2

[Alt Step 8]: The admin clicks on No button. Return to Step 4.

[Alt Step 10]: Failed to assign employee to delivery shift. The system will display the following unsuccessful modal.

	Leavit Name	Control Time	Notes
	Input Name	Control Type	Notes
	UNSUCCESSFUL	Placeholder Text	This serves as a header for the modal.
	Failed to assign delivery shift to employee. Please try again later.	Placeholder Text	This is to show that the assignment of the delivery shift information was unsuccessful.
	ОК	Button	None
CONCLUSION:	This use case concludes whe	en the employed	e has been successfully added
POST- CONDITION:	The employee has been	added to the E i	mployee table.
BUSINESS RULES:	Only an admin can add a new	w employee to	the system
IMPLEMENTATI ON CONSTRAINTS AND SPECIFICATIONS :	None		
ASSUMPTIONS:	None		
OPEN ISSUES:	None		

	ı	NKAP BOL	TING SYSTEM
DATE:	13-06-2021		
VERSION:	1.0		
AUTHOR(S)	Sonali Badrina	ath	
USE CASE NAME:	Cancel Delive	ry Shift	USE CASE TYPE
USE CASE ID:	4.11.		Business Requirements:
PRIORITY:	High		System Analysis: 🗆
SOURCE:	NKAP Bolting Requirements	=	System Design: ☑
PRIMARY BUSINESS ACTOR:	Administrator		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	Employee		
DESCRIPTION:	An Admin will be able to view the specific shift and cancel it. The system will update the assigned shift information using the EmployeeShift table.		
PRE-CONDITION:	The admir	should b	e on the Search Delivery Shift screen
TRIGGER:	The admin red	questing t	o cancel an assigned delivery shift.
TYPICAL COURSE OF EVENTS:	ACTOR ACTION:	Manual Action	SYSTEM RESPONSE: Automated Action
	Step 1: The admin wants to cancel an assigned delivery shift. Step 3: The		Step 2: The system invokes Use case 4.2. Search Delivery Shift. Step 4: The system will display a confirmation
	admin requests to cancel the assigned		popup to confirm the cancellation.

shift of an employee	Input Name	Control Type	Notes
by selecting the "Cancel" option in the Assign To column that in the line of the delivery shift information of which they want to cancel.	Confirm to Proceed	Placeholder Text	This serves as a header for the modal.
	Would you like to proceed with the cancellation of the selected assigned delivery shift?	Placeholder Text	This is to allow the admin to finalize the decision to submit the cancel the delivery shift.
	Yes	Button	Proceeds to the next step (Step 6)
	No	Button	Allows user to return to the Search Delivery Shift screen.
Step 5: The admin clicks on the	Step 6: The systeremove the emp EmployeeShift to	loyee informa	• •
button to confirm the cancellation		ttribute in able	Details
of the delivery shift. [Alt]	Employee El	, , =	The Employee_ID present in the EmployeeShift table is removed.

	Step 7: The system displays:				
		Inp	ut Name	Control Type	Notes
		Suc	ccess	Modal Heading	None
		of t ass was	igned shift	Placeholder Text	Placeholder Text to indicate the cancellation is done on the system.
		Ok		Button	None
		nam		the Assign to	e the employee column of the
ALTERNATE	[Alt Step 5]: The admin clicks on the "Cancel" button which terminates				
COURSES:	this use case.				
	Return to step 4				
	[Alt Step 7]: The cancellation could not be performed. The system display a modal informing the admin. The admin will Return to 9				
	Input Name		Control Type	Notes	
	Cannot be Cancelled		Placeholder Text	This serves	s as a header dal.
	This delivery shift cannot be cancelled as it is not yet assigned to an employee. Please try again later.		Placeholder Text	admin that	how that the t the delivery assigned and cannot be
			Button	None	
	No		Button	None	
CONCLUSION:	This use case concludes when the employee has been successfully added on the system.				

POST-CONDITION:	The employee no longer exists in the EmployeeShift table.
BUSINESS RULES:	Only an admin can add a new employee to the system
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

	NK	AP BOLT	ING SYSTEM	
DATE:	13-06-2021			
VERSION:	1.0			
AUTHOR(S)	Sonali Badri	nath		
USE CASE NAME:	Cancel Deli	very	USE CASE TYPE	
USE CASE ID:	4.11.		Business Requirements: □	
PRIORITY:	High		System Analysis: □	
SOURCE:	NKAP Boltin System Requiremen	_	System Design: ☑	
PRIMARY BUSINESS ACTOR:	Administrato	r		
PRIMARY THE SYSTEM ACTOR:	None			
OTHER PARTICIPATING ACTORS:	None	None		
OTHER INTERESTED STAKEHOLDER S:	Employee			
DESCRIPTION:	An Admin will be able to view the specific shift and cancel it. The system will update the assigned shift information using the EmployeeShift table.			
PRE- CONDITION:	The admin should be on the Search Delivery Shift screen			
TRIGGER:	The admin re	equesting	to cancel an assigned delivery shift.	
TYPICAL			SYSTEM RESPONSE:	
COURSE	ACTOR ACTION:	Manu	Automated Action	
OF EVENTS:	ACTION.	al Action		
	Step 1: The admin wants to cancel an assigned delivery shift.		Step 2: The system invokes Use case 4.2. Search Delivery Shift.	

Step 3: The admin requests to cancel the	Step 4: The system will display a confirmation popup to confirm the cancellation.
assigned shift of an	Input Name Control Notes Type
employee by selecting the "Cancel" option in the Assign To column that in the line of the delivery shift information of which they want to cancel.	Confirm to Label This serves as a header for the modal.
	Would you like to proceed with the cancellation of the selected assigned delivery shift? Label This is to allow the admin to finalize the decision to submit the cancel the delivery shift.
	Yes Button Proceeds to the next step (Step 6)
	No Button Allows user to return to the Search Delivery Shift screen.
Step 5: The admin clicks on	Step 6 : The system uses a SQL delete query to remove the employee information from the EmployeeShift table [Alt]:
the "Yes" button to confirm the cancellatio n of the delivery shift. [Alt]	Informatio Attribute in Details n to table be deleted:
	Employee Employee_I The Employee_I D present in the EmployeeShi

					ft table is removed.
		Step	7: The sy	stem displa	ys:
		Inpu	ıt Name	Control Type	Notes
		Suc	cess	Modal Heading	None
		of the	gned shift	Label	Label to indicate the cancellation is done on the system.
		Ok		Button	None
		empl	oyee nam	•	move the the Assign to ft table. [Alt]
ALTERNATE COURSES:	[Alt Step 5]: The adm terminates this use ca		cks on the	"Cancel" bu	itton which
	Return to step 4 [Alt Step 7]: The cancellation could not be performed. system will display a modal informing the admin. The a Return to Step 3				
	Input Name		Control Type	Notes	
	Cannot be Cancelled		Label	This serves as a header for the modal.	
	This delivery shift cannot be cancelled as it is not yet assigned to an employee. Please try again later.		Label	This is to show that the admin that the delivery shift is not assigned and therefore cannot be cancelled.	
	Yes		Button	None	
	No		Button	None	
CONCLUSION:	This use case concludes when the employee has been successfully added on the system.				

POST- CONDITION:	The employee no longer exists in the EmployeeShift table.
BUSINESS RULES:	Only an admin can add a new employee to the system
IMPLEMENTATI ON CONSTRAINTS AND SPECIFICATION S:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

2.1.5. ADMINISTRATION BACK-OFFICE SUBSYSTEM

NKAP BOLTING SYSTE	ΞM			
DATE:	05-06-21			
VERSION:	1.0			
AUTHOR(S)	Divya Bagratee			
USE CASE NAME:	Search Online Sales		USE CASE TYPE	
USE CASE ID:	5.1		Business Requirements:	
PRIORITY:	High		System Analysis:	
SOURCE:	NKAP Bolting Systen Requirements	n	System Design:	Ø
PRIMARY BUSINESS ACTOR:	Employee/Admin			
PRIMARY THE SYSTEM ACTOR:	None			
OTHER PARTICIPATING ACTORS:	None			
OTHER INTERESTED STAKEHOLDERS:	None			
DESCRIPTION:	This use case describes the event where an employee would like to view a Sale. The employee/admin will select the Administration Back Office Menu item on the navigation bar and click the "Search Online Sales" dropdown menu item. The system will display the Search Online Sale screen and request the search criteria from the employee/admin. The system will search for a matching record in the database and display it to the employee/admin in a datagrid view. The use case ends when the system successfully displays the Sale to the admin/employee.			
PRE-CONDITION:	The Sale must exThe employee/ad			
TRIGGER:			earch for a Sale Order	
TYPICAL COURSE	7 tr Employee// tarriir		RESPONSE:	
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Action	
	Step 1: An employee/admin wants to search for an online Sale.			
	Step 2: The employee/admin will select the Administration Back Office Menu item on the navigation bar and click the "Search Online		Step 3: The system will dis Online Sale screen with the components and enables the Input Control Name Type	following html

	Sales" dropdown			A.I.
	menu item.	Search Online Sales	Heading	None
		Filter	Label	Label for filter
		Filter	dropdown	Enables the employee/admin to select the filter they want to search for an online sale by
		Select Date	Label	Displayed if the employee/admin chooses the search by date filter option.
		Calendar	Calendar	Enables the employee/admin to choose the date. Displayed if the search by date filter option was chosen.
		Select Month	Label	Displayed if the employee admin chooses the search by month filter option.
		Month	Dropdown	Enables the employee/admin to choose the month. Displayed if the search by month filter option was chosen.
		Select Order Status	Label	Displayed if the employee/admin chooses the search by order status filter option.
		Order Status	Dropdown	Enables the employee/admin to choose the Order Status. Displayed if the search by order status filter option was chosen.
		Search	Button	Button when clicked will search for a matching record in the database.

			Cancel	Button	Button when clicked will terminate the search process.		
	Step 4: The employee/admin selects the filter option and click "Search" Button.		Step 5: The system makes use of a SQL read query to retrieve the matching Sale records from the Sale Entity.				
			Step 6: The system will display the Online Sales in a datagrid view with the following html components:				
			Input Control Notes Name Type				
			Online Sales	Heading	None		
			Sale Number	Table Column 1	Column in the data grid view		
	Step 7: The admin selects the sale they want to view by clicking the "View" button.		Sale Date	Table Column 2	Column in the data grid view		
			Order Status	Table Column 3	Column in the data grid view		
			Actions	Table Column 4	Column in the data grid view		
			View	Button	Button when clicked will view the full details of a specific sale.		
			Back	Button	None		
			Step 8: The system displays the View Sale Modal with the following html components:				
			Input Control Notes Name Type		l Notes		
			View Sale	1100000	g None		
			Sale Number	Label	Label for Sale Number		
			Sale Number	Text	Sale_ID retrieved from Sale Entity		
			Sale Date	Label	Label for Sale Date		
			Sale Date	Text	SaleOrder_Date retrieved from Sale Entity		

Order Status	Label	Label for Order Status
Order Status	Text	OrderStatus_ID retrieved from Sale Entity
Customer	Label	Label for Customer
Customer	Text	Customer_ID retrieved from Sale Entity
Payment Type	Label	Label for Payment Type
Payment Type	Text	PaymentType_ID retrieved from Sale Entity
Payment Date	Label	Label for Payment Date
Payment Date	Text	Payment_Date retrieved from Sale Entity
Payment Amount	Label	Label for Payment Amount
Payment Amount	Text	Payment_Amount retrieved from Sale Entity
Recieval Type	Label	Label for Recieval Type
Recieval Type	Text	SaleOrder_ RecievalType from Sale Entity
Items	Label	Label for Items Table
Items	Column 1 in table	ProductItem_Name from the ProductItem Entity
Quantity	Column 2 in table	SaleLine_Quantity from the SaleLine Entity
Back	Button	Button when clicked will return to the Sale Screen.
Pack Order	Button	Button when clicked allows the order Status to be updated.

ALTERNATE	ALT-STEP 6: The system fails to find matching records in the database and displays an error modal:				
	Input Name	Control Type	Notes		
	Error	Modal Heading	None		
	No sales matching the search criteria was found.	Label	Label to indicate the system failed to find a matching Sale record in the database.		
	Ok	Button	None		
COURSES:	ALT-STEP 8: The s	system fails to disp	play the sale and displays an error modal:		
	Input Name	Control Type	Notes		
	Error	Modal Heading	None		
	There was an unexpected error when trying to display the sales.	Label	Label to indicate the system failed to display the Sales.		
	Ok	Button	None		
CONCLUSION:	The use case concl	ludes when the sy	stem has successfully displayed the sales.		
POST-CONDITION:	The employee/admin is able to view the sales.				
BUSINESS RULES:	Only Employees and Admins can search for a Sale.				
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None				
ASSUMPTIONS:	None				
OPEN ISSUES:	None				

NKAP BOLTING SYSTE	ΞM					
DATE:	05-06-21					
VERSION:	1.0					
AUTHOR(S)	Divya Bagratee	Divya Bagratee				
USE CASE NAME:	Pack Order USE CASE TYPE					
USE CASE ID:	5.2		Business Requirements:			
PRIORITY:	High		System Analysis:			
SOURCE:	NKAP Bolting System Requirements		System Design:	Ø		
PRIMARY BUSINESS ACTOR:	Employee					
PRIMARY THE SYSTEM ACTOR:	None					
OTHER PARTICIPATING ACTORS:	None					
OTHER INTERESTED STAKEHOLDERS:	None					
DESCRIPTION:	This use case describes the event where an employee searches for orders which require packing and proceeds to pack the order. Once the employee has packed the order, they will indicate so on the system and the system will update the order status of the order appropriately in the Sale Entity . The system will notify the customer that their order is ready for collection. The use case concludes when the Order has been packed and the status has been updated accordingly.					
PRE-CONDITION:	 The Employee must The sale must exist Packing". 	• • •	in to the system. em and have an order stat	us of "Needs		
TRIGGER:	An employee wants to p	ack an ord	er.			
TYPICAL COURSE		SYSTEM	RESPONSE:			
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Action			
	Step 1: An employee wants to pack an order.					
	Step 2: The employee will select the Administration Back Office Menu item on the navigation bar and click the "Pack Order" dropdown menu item.		Step 3: The system invok Search Online Sales.	es Use Case 5.1		
	Step 4: The employee clicks the "Pack Order" screen with the following html component and enables the input controls:					

	Input Name	Control Type	Notes
	Pack Order	Heading	None
	Has this order been packed?	Label	Label to prompt confirmation that the order has been packed.
	Has this order been packed?	Checkbox	Checked by employee to confirm order is packed
	For Delivery	Label	Label to indicate order is for delivery.
	For Delivery	Checkbox	Checked by employee if order is for delivery
	For Collection	Label	Label to indicate order is for Collection
	For Collection	Checkbox	Checked by employee if order is for collection.
	Confirm	Button	Button when clicked will update the order status appropriately.
	Cancel	Button	Button when clicked will terminate the Use Case
Step 6: The employee confirms the order has been packed and that		ollection" by n	es Order Status to naking use of a
the order is for collection and clicks the "Confirm" button.	Details to be Updated	Attribute in E	ntity Notes
		OrderStatus_I Sale Entity	D in the None

			Step 8: The system notifies the customer via SMS that their order is ready for collection by retrieving the		
			 Custom 	er_CellPhone	Number
			from the Custon	mer Entity.	
			Step 9: The system successfully completes the Pack Order Process and displays a success Modal:		
			Input Name	Control Type	Notes
			Success	Modal Heading	None
			Order has been Packed.	Label	Label to indicate the Order was packed.
			Ok	Button	None
ALTERNATE	ALT-STEP 6: The a	dmin clicks the	'Cancel" Button	. Terminate U	se Case.
	ALT STEP 7: The o "Ready for Delivery				
	Details to be Updated	Attribute in Er	ntity Note	es	
	Order Status	OrderStatus_ Sale Entity	ID in the No	ne	
COURSES:	ALT-STEP 9: The s modal:	ystem fails to up	date the Order	Status and di	splays an error
	Input Name	Control Type	Notes		
	Error	Modal Heading	None		
	There was an unexpected error when trying to complete the pack order process.	Label		ate the syste pack order p	
	l L ·				

	Ok	Button	None				
CONCLUSION:	The use case concludes when the Order has been packed and the status has been updated accordingly						
POST-CONDITION:	The Order Statu	The Order Status has been updated					
BUSINESS RULES:	Once an order has been packed and the status has been updated, it cannot be edited.						
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None						
ASSUMPTIONS:	None						
OPEN ISSUES:	None						

NKAP BOLTING SYS	STEM							
DATE:	05-06-21							
VERSION:	1.0							
AUTHOR(S)	Divya Bagratee							
USE CASE NAME:	Do Stock-Take		USE CASE TYPE					
USE CASE ID:	5.3		Business Requirements:					
PRIORITY:	High		System Analysis:					
SOURCE:	NKAP Bolting System Requirements	1	System Design:	Ø				
PRIMARY BUSINESS ACTOR:	Admin							
PRIMARY THE SYSTEM ACTOR:	None							
OTHER PARTICIPATING ACTORS:	None							
OTHER INTERESTED STAKEHOLDERS:	None							
DESCRIPTION:	item. The admin will s navigation bar and cli- record the stock take the <u>ProductItem_Stoc</u>	elect the ack the "Dod date in the kTake Enductitem"	vent where an Admin condu Administration Back Office Notes Stock-Take" dropdown mer e StockTake Entity and the country. The system will then adjusted Entity. The Use Case Ends work take.	fenu item on the nu item. The admin will quantity of the stock in just the quantity on hand				
PRE-CONDITION:	The Admin must kThe Admin must		value of stock on hand for an	item				
TRIGGER:	An Admin wants to co	nduct a s	tock-take.					
TYPICAL COURSE		SYSTEM	M RESPONSE:					
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Action					
	Step 1: An admin wants to do a Stock- Take							
	Step 2: The admin will select the Administration Back Office Menu item on the navigation bar and click the "Do Stock-Take" dropdown menu item.		Step 3: The system invoke Product	s Use Case 9.10 Search				

<u> </u>	ı	<u> </u>					
Step 4: The admin clicks the "Do Stock- Take Button"		Step 5: The system displays the Do-stock Take Modal with the following html components and enables the input controls:					
		Input Nam	ne Control Type	Notes			
		Stock-Tak	Modal Heading	None			
		Date	Label	Label for Stock Take Date			
		Stock Tak Date	Date Picker	Enables the admin to select the date of the stock take.			
		Quantity	Textbox	Enables the admin to enter the stock take quantity			
		Save	Button	Button when clicked proceeds with the stock-take			
		Cancel	Button	Button when clicked terminates the stock- take process			
Step 6: The admin provides the stock-take details and		Step 7: The system validates the stock take detail according to the validation requirements of the relevant entities:					
clicks the "Save" button.		Input Name	Attribute in Ent		ion ements		
		Stock Take Date	StockTake_Date StockTake Enti	4.7			
		Stock Take Quantity	StockTake_Quain ProductItem StockTake	-	juired gers		

		;	Step 8: The system requests confirmation for the stock take from the admin and displays a confirmation modal.						
			Input Nam	ie i	Control Type	Notes			
			Confirmati	ion	Modal Heading	None			
			Do you wa to proceed with the stock take	t	Label	Label to prompt confirmation for the stock take			
			Confirm		Button	Button when clicked will proceed with the stock take			
			Cancel		Button	Button when clicked will terminate the stock take process.			
	0. 0.7		O: 10 TI			(001 :			
	Step 9: The admin clicks the "Confirm" button.				em s makes use stock take detail				
			Details to be saved	Attri	bute in Entity	Notes			
			Stock Take ID		kTake_ID in : kTake Entity	The system will generate a unique StockTake_ID by retrieving the previous StockTake_ID and incrementing it by on1.			
			Stock Take Date	e StockTake		None			
			Stock Take Quantity	in <u>Pr</u>	kTake_Quantity coductItem kTake	None			
			Step 11: The	e syst	em will update th	е			

			Quantity_on_hand					
			in the ProductItem Entity for the specific product.					
			Step 12: The system successfully completes to stock take and displays a success modal:					
			Input Name		Control Type	Notes		
			Su	ccess	Modal Heading	None		
		was		ock Take s mpleted ccessfully.	Label	Label to indicate the system successfully completed the stock take.		
			Ok		Button	None		
ALTERNATE	ALT-STEP 9: The A	dmin clicks	the "d	cancel" butto	n. Terminate	Use Case.		
	ALT-STEP 7: The s Validation Error Mo		o vali	date the sto	ck take detail	ls and displays a		
	Input Name	Control Type Notes						
	Validation Error	Modal Hea	ding					
	The information entered is in the incorrect format. Please try again.	Label	Label		None.			
	Ok	Button		None				
COURSES:	ALT-STEP 12: The modal:	system fails	to co	mplete the s	stock-take an	d displays an error		
	Input Name	Control Ty	ype Notes					
	Error	Modal Hea	ding	g None				
	There was an unexpected error when trying to do the stock-take,	Label		Label to indicate the system failed to do the stock-take.				
	Ok	Button		None				
CONCLUSION:	The Use Case Concludes when the system has successfully completed the stock take.							
POST- CONDITION:	The quantity on hand for a product is accurate.							

BUSINESS RULES:	Only an admin can do a stock take The admin must be logged in.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

NKAP BOLTING SYSTE	EM						
DATE:	06-06-21						
VERSION:	1.0						
AUTHOR(S)	Divya Bagratee						
USE CASE NAME:	View Delivery Shift Sched	lule	USE CASE TYPE				
USE CASE ID:	5.4		Business Requirements:				
PRIORITY:	High		System Analysis:				
SOURCE:	NKAP Bolting System Requirements		System Design:	Ø			
PRIMARY BUSINESS ACTOR:	Employee						
PRIMARY THE SYSTEM ACTOR:	None						
OTHER PARTICIPATING ACTORS:	None						
OTHER INTERESTED STAKEHOLDERS:	None						
DESCRIPTION:	The Employee will select to navigation bar and click the system will retrieve the sh	the Admin ne "View S nift schedu e use case	where employee views the SI nistration Back Office Menu it Shift Schedule" dropdown meule information from the Shift e ends when the system has employee.	tem on the enu item. The <u>Entity</u> and			
PRE-CONDITION:	The employee must be	e logged i	n				
TRIGGER:	An Employee wants to vie	w the shif	ft schedule				
TYPICAL COURSE		SYSTEN	M RESPONSE:				
OF EVENTS:	ACTOR ACTION:	Manual Action					
	Step 1: An employee wants to view the Delivery shift schedule.						

Step 2: The employee will select the Administration Back Office Menu item on the navigation bar and click the "View Delivery Shift Schedule" dropdown menu item.	Step 3: The system makes use of a SQL Read query to retrieve the Delivery Shift Schedule information from the Shift Entity and EmployeeShift Entity. Step 4: The system displays the Shift Schedule screen with the following html				
	components:		-		
	Input Name	Control Type	Notes		
	Schedule	Heading	None		
	Employee	Column 1 in table	Employee_ID retrieved from the Employee_Shift Entity. Shows the Employee Names.		
	Monday	Column 2 in table	Date_ID retrieved from the Shift Entity. Shows the day of the week.		
	Tuesday	Column 3 in table	Date_ID retrieved from the Shift Entity. Shows the day of the week.		
	Wednesday	Column 4 in table	Date_ID retrieved from the Shift Entity. Shows the day of the week.		
	Thursday	Column 5 in table	Date_ID retrieved from the Shift Entity. Shows the day of the week.		
	Friday	Column 6 in table	Date_ID retrieved from the Shift Entity. Shows the day of the week.		

			Column Rows	Column Rows	Time_ID retrieved from the Shift Entity. Shows the time slot the employee works on a specific day	
ALTERNATE COURSES:	ALT-STEP 4: The s modal:	system fails to disp	olay the shift so	chedule an	d displays an error	
	Input Name	Control Type	Notes			
	Error	Modal Heading	None			
	There was an unexpected error when trying to display the Delivery Shift Schedule.	Label Label to indicate the sy display the shift schedu				
	Ok	Button	None			
CONCLUSION:	The Use Case Con schedule to the em		system has suc	ccessfully o	displayed the shift	
POST-CONDITION:	The employee k	knows when their	next shift is			
BUSINESS RULES:	Only an Employee and admin can view the shift schedule					
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None					
ASSUMPTIONS:	None					
OPEN ISSUES:	None					

NKAP BOLTING SYSTEM							
DATE:	06-06-21	06-06-21					
VERSION:	1.0	1.0					
AUTHOR(S)	Divya Bagratee	Divya Bagratee					
USE CASE NAME:	Write-off Stock	USE CASE TYPE					
USE CASE ID:	5.5	Business Requirements:					
PRIORITY:	High	System Analysis:					
SOURCE:	NKAP Bolting System Requirements	System Design:	Ø				
PRIMARY BUSINESS ACTOR:	Employee						

PRIMARY THE	l.,							
SYSTEM ACTOR:	None							
OTHER PARTICIPATING ACTORS:	None							
OTHER INTERESTED STAKEHOLDERS:	None							
DESCRIPTION:	This use case describes the event where the Admin wants to write off stock. The admin will select the Administration Back Office Menu item on the navigation bar and click the "Write-Off Stock" dropdown menu item. The system will write-off stock according to the write off information provided by the admin and update the relevant entities.							
PRE-CONDITION:	The admin must be	logged in	to the system					
TRIGGER:	An admin wants to write	off stock.						
TYPICAL COURSE		SYSTEM	ARESPONSE:					
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Ad	ction				
	Step 1: An admin wants to write off stock.							
	Step 2: The admin will select the Administration Back Office Menu item on the navigation bar and click the "Write-Off Stock" dropdown menu item.		Step 3: The system will invoke use case 9.10 Search Product					
	Step 4: The admin will click the "write-off stock" button.		Step 5: The si Stock screen components:		isplay the Write Off owing html			
			Input Name	Control Type	Notes			
			Write-Off Stock	Heading	None			
			Date	Label	Label for Write-Off Date			
		DateDateEnables adrPickerselect write-						
			Quantity	Textbox	Enables admin to enter write-off quantity			
			Reason	Textbox	Enables admin to enter write-off reason			

			Save	E	Button	proc	on when clicked eeds with write- rocess
			Cancel	E	Button	Button when clicked terminates write-off process	
	Step 6: The admin provides the write-off information and clicks		Step 7: The information requiremen	acco	ording to the	he val	idation
	the "save" button.		Input Name	Attr Enti	ibute in ity		Validation Requirements
			Write- off quantity	in <u>P</u> Writ	WriteOff_Quantity in ProductItem_ WriteOffStock Entity		RequiredIntegers only
			Write- off Reason	in <u>P</u> Writ	WriteOff_Reason in <u>ProductItem_</u> WriteOffStock Entity		RequiredAlphabet only
			Write- off Date Write-Off Entity			RequiredMust be in Date Format	
			Step 8: The confirm the modal:				e admin to vs a confirmation
			Input Nar	ne	Control Type		Notes
			Confirmation		on Modal Heading		None
			Do you w to procee with the Write-off?	ed	Label		Label to prompt confirmation for the stock take
		Confirm		Button		Button when clicked will proceed with the write-off	
			Cancel		Button		Button when clicked will terminate the write-off process.

Step 9: The admin will click the "Confirm" button.		•	e system makes use the write-off deta		
		Details to be Saved	Attribute in Entity	None	
		Write-Off ID	WriteOff_ID in Write-Off Entity	The system will generate a unique Write-off_ID by retrieving the previous Write-Off_ID and incrementing it by one.	e O S O
Write-off quantity	WriteOff_Quantity in ProductItem WriteOffStock Entity	y None			
		Write-off Reason	WriteOff_Reason in ProductItem_ WriteOffStock Entity	None	
		Write-off Date	WriteOff_Date. in Write-Off Entity	None	
		Cton 44. Th		an the n	
		Quantity_Or	e system decrease n_Hand in the Prod quantity specified	ductItem Entity b	by
		Step 12: The system successfully completes to Write-off and displays a success modal:			the
		Input Nam	e Control Type	Notes	
		Success	Modal Heading	None	
		Write-Off was completed successfu		Label to indicate the system successfully completed the write-off.	

			Ok	Button	None		
ALTERNATE	ALT-STEP 9: The a	dmin clicks the "d	cancel" button.	Terminate U	se Case.		
COURSES:	ALT-STEP 7: The signal validation error mode		date the write	-off details an	d displays a		
	Input Name	Control Type	ontrol Type Notes				
	Validation Error	Modal Heading	None				
	The information entered is in the incorrect format. Please try again.	Label	None.				
	Ok	Button	None				
	ALT-STEP 12: The modal:	system fails to co	mplete the wr	ite-off and dis	plays an error		
	Input Name	Control Type	Notes				
	Error	Modal Heading	None				
	There was an unexpected error when trying to do the write-off, Label to indicate the system fa the write-off.		em failed to do				
	Ok	Button	None				
CONCLUSION:	There is record of the for sale.	ne stock being wr	itten-off and th	e stock is no	longer available		
POST-CONDITION:	The stock is wri	tten-off and is not	available for	sale anymore			
BUSINESS RULES:	Only an Admin can write-off stock.						
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None						
ASSUMPTIONS:	None						
OPEN ISSUES:	None						

2.1.6. CUSTOMER SUBSYSTEM

NKAP Bolting System

Chardé Devine Vries	Date: <u>06/06/2021</u>
	Chardé Devine Vries

Version:

USE CASE NAME:	View Products		USE CASE TYP	PE
USE CASE ID:	6.1		Business Requirements:	
PRIORITY:	High		System Analysis:	
Source:	NKAP Bolting system		System Design:	\square
PRIMARY BUSINESS	Customer			
ACTOR:				
PRIMARY THE	None			
SYSTEM ACTOR:				
OTHER	None			
PARTICIPATING				
ACTORS:				
OTHER INTERESTED	None			
STAKEHOLDERS:				
DESCRIPTION:	This use case describes the event was the use case starts when the custon product details" use cases in order the "ProductItem" entity. The systoniew and browse through all availations.	omer click to select em displa	s on the products tab, they can inv a product. The system will retriev ys all the relevant information. The	voke the "view e information from
PRE-CONDITION:	The customer must have an action of the customer must have action of t	count an	d be logged into the system	
TRIGGER:	A user wishes to browse the produ	cts on the	e system	
TYPICAL COURSE			SYSTEM RESPONSE:	
OF EVENTS:		MAN	AUTOMATED ACTION	
	ACTOR ACTION:	UAL		
		ACTI		
		ON		
	Step 1: The customer			
	request to browse through			
	the products by category.			

Step 2: The customer clicks on the 'Products' tab on the menu-strip and selects a product category from the menu-items:

- Anchors
- Bolts
- Clamps
- Fasteners
- Nuts
- Pins
- Rivets
- Screws
- Shackles
- Thread Rods
- Turnbuckle
- Wall Fixers
- Washers

Step 3: The system loads the **Products** screen with the following elements(showing the selected product category):

Element	Element	Notes:
Name	Туре	
Products	Modal Heading	To indicate what the screen is for
lblSearch	Label	This is the label to prompt the user to enter a search query.
txtSearch	Textbox	To enter the search criteria
btnSearch icon	Icon button	This will submit the search criteria
cardProdu ctItems	Card	To show each product item. Each card contains: ProductItem_Name ProductItem_I mage Retrieved from the Productitem entity
The followin cardProduct	g elements a Items:	ire in
imgProduc tItem	Img	To display the product item image
lblProductI temName	Label	To display the name of the Product Item
lblProductI tem Cost	Label	To display the cost of the

product item

ALTERNATE COURSES:	ALT Step 3: There system for Element Name Error There was an error loading the Products page. Please try again later. OK Use Case Terminated	Fails to load Element Label Label Button		This serve for the notes to loading to view was	res as a header nodal. show that the Products information accessful.
Conclusion:		when the relevant product details are displayed			
Post-condition:	The customer can see the	Products s	creen		
Business Rules:	None				
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None				
Assumptions:	None				
OPEN ISSUES:	None				

Author (s): Chardé Devine Vries Date: 06/06/2021

Version:

USE CASE NAME:	Search Products Details		USE CASE TYPE		
USE CASE ID:	6.2		Business Requirements:		
PRIORITY:	High		System Analysis:		
Source:	NKAP Bolting system		System Design:		
PRIMARY BUSINESS ACTOR:	Customer				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	This use case describes the process where the customer searches for a specific Product. Once the customer enters the search criteria the system will retrieve the matching results from the ProductItem entity the use case ends once the retrieved results have been displayed to the admin.				
PRE-CONDITION:	The customer must have an account	unt and be	e logged into the system		
TRIGGER:	A user wishes to browse the products	on the sys	stem		
TYPICAL COURSE			SYSTEM RESPONSE:		
OF EVENTS:	Actor Action:	MAN AUTOMATED ACTION UAL ACTI ON			
	Step 1: The customer requests to search for a product.		Step 2: The system invokes Search Products.	s Use Case 6.1	
	Step 3: The customer enters the product search criteria into the search textbox.				

Step 4: The customer clicks on the search icon.	Step 5: The system captures and validates the information against the		
	<u>Productite</u>	m table	
	Input Name	Attribute Type	Validation Requirem ents
	Name:	ProductItem_ Name	Maximum of 50 characters
	ID:	ProductItem_ ID	Characters must be integers
	ALT		
	read the m	e system uses a natching entries f ntegory entity. Th	rom the
		m_table: ductItem _ID ductItem_Name	
	-	e system displays earch results.	s the

Step 8: The customer clicks on the product they were searching for.	Step 9: The system loads the Product Details view with the following elements(showing the selected product category):
	Element Element Notes: Name Type
	Product Modal To indicate Details Header what the view is for
	The following elements are Retrieved from the ProductCategory entity
	IblProduct Label To display the description of the Product Category Category Type_Desc
	IblProductI Label To display the name of the Product Item O ProductIt em_Nam e
	IblProductI temDesc the description of the product item ProductIt em_Desc
	imgProduc Img To display the product item image

					ProductItem_Image
			IblProductI tem_Cost	Label	To display the cost of the product item o ProductIt em_Cost
			btnShowLe ss	Button	To display the less of the description information of the product item
			btnAddCar tItem	Button	To add an item to the cart.
			[ALT]		
EDNATE	ALT Step 5: The system fails to capture and validate the information and displays				

ALTERNATE

ALT Step 5: The system fails to capture and validate the information and displays the error message.

Element Name	Element Type	Notes
Error	Label	This serves as a header for the modal.
Please ensure that the details you have entered are valid	Notification/Pop-up	To inform the user that the details entered are invalid.
ОК	Button	None

Use Case Terminated

Courses:	ALT Step 9: There system fails to retrieve the product items:					
	Element Name	Element Type	Notes			
	Error	Label	This serves as a header for the modal.			
	There was an error loading the Products page. Please try again later.	Label	This is to show that the saving of the product category information was unsuccessful.			
	OK Use Case Terminated	Button	None			
Conclusion:	The use case concludes when the relevant product details are displayed successfully.					
Post-condition:	User sees Products detail view					
BUSINESS RULES:	None					
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None					
Assumptions:	None					
OPEN ISSUES:	None					

Author (s): Chardé Devine Vries Date: 06/06/2021

Version:

USE CASE NAME:	Search Specials and Promo	tions	USE CASE TYPE			
USE CASE ID:	6.3		Business Requirements:			
PRIORITY:	High		System Analysis:			
Source:	NKAP Bolting system		System Design:	\checkmark		
PRIMARY BUSINESS	Customer					
ACTOR:						
PRIMARY THE	None					
SYSTEM ACTOR:						
OTHER	None					
PARTICIPATING						
ACTORS:						
OTHER INTERESTED	None					
STAKEHOLDERS:						
DESCRIPTION:	This use case describes the event where a customer would like to browse all available specials and promotions. The use case starts when the customer clicks on the Promotions tab, they can invoke the "view product details" use cases in order to view a selected product. The system will retrieve information from the ProductItem entity and Product_Special entity The system displays all the relevant information. The customer is able to view and browse through all available product.					
PRE-CONDITION:	The customer must have	The customer must have an account and be logged into the system				
TRIGGER:	A user wishes to browse the	e products with	specials and promotions on the s	ystem		
Typical Course			SYSTEM RESPONSE:			
OF EVENTS:	Actor Action:	MANUAL ACTION	AUTOMATED ACTION			
	Step 1: The customer request to browse through the specials and promotions.					

Step 2: The customer clicks on the 'Promotions' menu item on the menustrip.

Step 3: The system loads the **Promotions** screen with the following elements(showing the selected product category):

Element Name	Element Type	Notes:
Specials and Promotion s	Header	To indicate what the screen is for
lblSearch	Label	This is the label to prompt the user to enter a search query.
txtSearch	Textbox	To enter the search criteria
btnSearch icon	Icon button	This will submit the search criteria
cardProductItems	Card	To show each product item. Each card contains: ProductItem_Name ProductItem_I mage Special_Price Special_EndD ate Retrieved from the: ProductItem entity Product_Special entity Special entity
The followin	g elements a	are in
btnAddCar tItem	Button	To add an item

	lbIProductI temName	Label	To display to name of the Product Iter	2
	imgProduc tItem	Image	To display to product iter image	
	lblSpecialP rice	Label	To display to price of the Product Iter	
	lblSpecialE nds	Label	To display to end date of Special price	the
	[ALT]			
Step 4: The customer enters the product search criteria into the search textbox.				
Step 5: The customer	Step 6: The system captures and validates the information against the Productitem table			tes the
clicks on the search		igainst the <u>Pr</u>	oductitem ta	
=	information a	gainst the <u>Pr</u> Attribute Typ		
clicks on the search	Input Name		oe	Validation Requirent Maximur
clicks on the search	Input Name Name:	Attribute Typ	oe _Name	Validation Requirement Maximum characte
clicks on the search	Input Name Name:	Attribute Typ	oe _Name	ble Validation
clicks on the search	Input Name Name:	ProductItem ProductItem ystem uses a entries from	_Name _ID read query t	Validation Requirement of the character
clicks on the search	Input Name Name: ID: ALT Step 7: The sthe matching Category entited the matching Category entitle the matching Category ent	ProductItem ProductItem ProductItem system uses a entries from ity. The syste	_Name _ID read query t	Validation Requirement Maximum character Character be integer

	Step 8: The sys		ays the retrieved
Step 9: The admin clicks on the product special they were searching for.	Details view w	ith the fol	ds the SpecialProduct lowing elected product
	Element Name	Elemen t Type	Notes:
	Special Product Details	Header	To indicate what the view is for
	_	n entity ,Pro	are Retrieved from oduct Special, Discount
	imgProductI tem	Img	To display the product item image O Produ ctItem _Imag e
	lblProductC ategoryDes c	Label	To display the description of the Product Category CategoryTy pe_Desc ProductCategory ctCategory Desc
	lblProductIt emName	Label	To display the name of the Product Item O ProductItem _Nam e

lblSpecialPri ce	Label	To display the price of the Product Item Specia I_Pric e
lblProductIt emDesc	Label	To display the description of the product item o ProductitemDesc
lblSpecialEn ds	Label	To display the end date of the Special price Special I_End Date
btnShowLes s	Button	To display the less of the description information of the product item
btnAddCartI tem	Button	To add an item to the cart.

ALTERNATE

ALT Step 3: The system fails to capture and validate the information and displays the error message.

Element Name	Element Type	Notes
Error	Label	This serves as a header for the modal.
Please ensure that the details you have entered are valid	Notification/Pop-up	To inform the user that the details entered are invalid.

	ОК	Button	None		
Courses:	Use Case Terminated ALT Step 10: There system	fails load Promotions pag	e:		
	Element Name	Element Type	Notes		
	Error	Label	This serves as a header for the modal.		
	There was an error loading the Promotions page. Please try again later.	Label	This is to show that the saving of the product category information was unsuccessful.		
	ОК	Button	None		
	Use Case Terminated				
CONCLUSION:	The use case concludes when the relevant product details are displayed successfully.				
Post-condition:	User sees Promotions view				
BUSINESS RULES:	None				
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None				
Assumptions:	None				
OPEN ISSUES:	None				

Author (s): Chardé Devine Vries Date: 06/06/2021

Version:

USE CASE NAME:	Add item to cart USE CASE TYPE				
USE CASE ID:	6.4		Business Re	quirements	s: 🗆
PRIORITY:	High		System Ana	lysis:	
Source:	NKAP Bolting system		System Desi	gn:	\square
PRIMARY BUSINESS ACTOR:	Customer				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	This use case describes the process where the customer searches for a specific Product. Once the customer enters the search criteria the system will retrieve the matching results from the ProductItem entity the use case ends once the retrieved results have been displayed to the admir				
PRE-CONDITION:	The customer must have an account	st have an account and be logged into the system			
TRIGGER:	A user wishes to browse the products of	s on the system			
Typical Course			SYST	EM RESPONS	E:
OF EVENTS:	ACTOR ACTION:	MAN UAL ACTI ON	AUTOMATE	D ACTION	
	Step 1: The customer requests to add a product item to cart.				
	Step 2: The customer clicks the 'add to cart' button on either the Product view, Product Details view or Promotions		customer we cart by displ	ould like to aying a mes	firm that the add the item to ssage box:
	view		Element name	Element type	Notes:

	Confirm Add to cart	Header	To indicate what the view is for
	IblConfir mMessag e	Label	Text asking if they would like to add the item to cart
	btnAdd	Button	To proceed and add item to cart.
	btnCancel	Button	To cancel the process and add item to cart.
	<u> </u>		

Step 4: The customer clicks the Add button

ALT

Step 5: The system uses a SQL insert to add the Product Item to the cart the **CartLine table** and generates a new cart if one does not exist yet.

Element Name	Element Type	Notes
CartLineID	CartLineID	The system generates a new CartLine_I D by retrieving the last CartLine_I D in the CartLine table and incrementing it by one.
ProductIte m_ID	Productite m_ID	The system captures and adds the

CartLine Quantity CartLine Quantity CartLine Quantity The system updates the CartLine Quantity in the CartLine table by incrementi ng it by one. ALT Step 6: The system displays a successfully added to cart message box Flement Name Type Added to cart To indicate what the view is for Added to cart To inform the user that the litem was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the				1	T
Quantity Quantity updates the CartLine Quantity in the CartLine table by increment in git by one. ALT Step 6: The system displays a successfully added to cart message box Element Name Type Added to Header To indicate what the view is for Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the					ProductIte m ID
Quantity Quantity updates the CartLine Quantity in the CartLine table by increment in git by one. ALT Step 6: The system displays a successfully added to cart message box Element Name Element Type Added to Header To indicate what the view is for Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the			CartLine	CartLine	The system
the CartLine Quantity in the CartLine table by incrementing it by one. ALT Step 6: The system displays a successfully added to cart message box Element Name Type Notes Type Added to Cart what the view is for Added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the item was successfully added to Cart the user that the user that the item was successfully added to Cart the user that					
CartLine Quantity in the CartLine table by incrementi ng it by one. Step 6: The system displays a successfully added to cart message box Element Name Type Added to cart Added to cart Added to cart Added to cart To indicate what the view is for Added to cart To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the			Quartity	Quantity	
ALT Step 6: The system displays a successfully added to cart message box Element Name Type Added to Cart message box Added to Label To inform the user that the item was successfull y added to cart be item was successfull y added to cart the user that the item was successfull y added to cart be item was successfull y added to cart be item was successfull be item was successfull and the item was successfull y added to cart be item was successfull y added to cart when the user that the item was successfull y added to cart message box					
the CartLine table by incrementing it by one. ALT Step 6: The system displays a successfully added to cart message box Element Rame Type Added to Cart what the view is for Added to Cart Label To inform the user that the item was successfull y added to Cart the bitnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the					
ALT Step 6: The system displays a successfully added to cart message box Element Name Ilement Notes Name Type Added to cart what the view is for Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the					=
ALT Step 6: The system displays a successfully added to cart message box Element Name Type Notes Added to cart What the view is for Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the					
ALT Step 6: The system displays a successfully added to cart message box Element Name Type Added to Header To indicate what the view is for Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the					CartLine
ALT Step 6: The system displays a successfully added to cart message box Element Rame Type Notes Name Type Notes Added to Header To indicate what the view is for Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the					table by
ALT Step 6: The system displays a successfully added to cart message box Element Name Type Notes Added to Header To indicate what the view is for Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the					incrementi
ALT Step 6: The system displays a successfully added to cart message box Element Name Type Notes Added to Header To indicate what the view is for Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the					ng it by
ALT Step 6: The system displays a successfully added to cart message box Element Name Type Notes Added to cart What the view is for Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the					
Step 6: The system displays a successfully added to cart message box Element Name Type					
Step 6: The system displays a successfully added to cart message box Element Name Type			ΔΙΤ		
successfully added to cart message box Element Name Type Added to Header To indicate what the view is for Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the					
Element Name Type Added to Header To indicate what the view is for Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the					
Added to cart Header To indicate what the view is for Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the			successfully a	dded to cart r	nessage box
Added to cart To indicate what the view is for Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the			Element	Element	Notes
cart what the view is for Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the			Name	Type	
Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the			Added to	Header	To indicate
Added to Label To inform the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the			cart		what the
cart the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the					view is for
cart the user that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the			Added to	Label	To inform
that the item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the					
item was successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the			Care		
successfull y added to cart btnOkay Button To proceed Step 7: The system updates the cart icon in the page header. Using the					
Step 7: The system updates the cart icon in the page header. Using the					
Step 7: The system updates the cart icon in the page header. Using the					
Step 7: The system updates the cart icon in the page header. Using the					_
Step 7: The system updates the cart icon in the page header. Using the					cart
in the page header. Using the			btnOkay	Button	To proceed
in the page header. Using the					
in the page header. Using the			Step 7: The sv	stem update	s the cart icon
CartLine_Quantity attribute in the					
CartLine entity to show the amount of			-	=	
items in the cart.					. amount of
ALT Step 4: The user clicks the cancel button.					
ALT Step 4: The user clicks the cancel button.	 ALT Ston A. The user clicks the ex	ancal hut	on		

Courses:	Use Case Terminated				
	ALT Step 6: There system fa	ils to add the product items	to cart:		
	Element Name	Element Type	Notes		
	Error	Label	This serves as a header for the modal.		
	There was an error while added the product to cart. Please try again.	Label	This is to show that adding the item to cart was unsuccessful.		
	ОК	Button	None		
	Use Case Terminated				
Conclusion:	The use case concludes when the relevant product details are displayed successfully.				
Post-condition:	User sees Products detail view				
Business Rules:	None				
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None				
Assumptions:	None				
OPEN ISSUES:	None				

Author (s): Chardé Devine Vries Date: 06/06/2021

Version:

USE CASE NAME:	View Cart		USE CASE TYPE						
USE CASE ID:	6.5		Business Requirements:						
PRIORITY:	High		System Analysis:						
Source:	NKAP Bolting system		System Design:	\checkmark					
PRIMARY BUSINESS ACTOR:	Customer	Customer							
PRIMARY THE SYSTEM ACTOR:	None								
OTHER PARTICIPATING ACTORS:	None								
OTHER INTERESTED STAKEHOLDERS:	None	None							
DESCRIPTION:	This use case describes the event where a customer would like to see all the items in their cart. The use case starts when the customer clicks on the Cart icon in the page header. The system will retrieve information from the CartLine entity . The system displays all the relevant information. The customer is able to view all cart items, remove items or checkout.								
PRE-CONDITION:	The customer must have an account and be logged into the system								
TRIGGER:	A user wishes to view items	ms in cart.							
TYPICAL COURSE		SYSTEM RESPONSE:							
OF EVENTS:	Actor Action:	MANUAL ACTION	AUTOMATED ACTION						
	Step 1: The customer request to see the cart.								
	Step 2: The customer clicks on the 'Cart' icon on the menu-		Step 3: The system loads the Cart view with the following elements						
	strip.		Element Elemen N Name t Type	otes:					

Cart Header what the screen is for tblCart Table To show each product item in cart using a SQL insert Each row contains: Productite not not not make the productite not	 Ţ	Ţ	T		
each product item in cart using a SQL Insert Each row contains: Productite m_Name Productite m_Name Productite m_Name Productite m_Special_Price is special_Price is shown) Retrieved from the: Product special entity Product entity			Cart	Header	what the
m_Name Producttle m_Image Special_Pr ice or Producttle m_Cost Special_E nidbate (if Special_Pr ice is shown) Retrieved from the: Productt em entity Product s pecial entity Subtotal Label To display subtotal label IblSubto tal VAT Label To display vAT label IblVAT Label To display vAT total			tblCart	Table	each product item in cart using a SQL Insert Each row
Subtotal Label To display subtotal label					m_Name ProductIte m_Image Special_Pr ice or ProductIte m_Cost Special_E ndDate (if Special_Pr ice is shown) Retrieved from the: ProductIt em entity
subtotal label IblSubto Label To display subtotal VAT Label To display VAT label IblVAT Label To display VAT total Total Label To display					pecial entity • Special
tal subtotal VAT Label To display VAT label IbIVAT Label To display VAT total Total Label To display			Subtotal	Label	subtotal
VAT label IbIVAT Label To display VAT total Total Label To display				Label	
VAT total Total Label To display			VAT	Label	To display
			lbIVAT	Label	
			Total	Label	

		IblTotal	Label	To display total price
		Checkou	Button	To go to checkout screen
		Back	Button	
		The follow	ing eleme	nts are
		displayed ietm:	in tblCart	for each
		imgProd uctItem	Image	To display the product item image
		lblProdu ctItemN ame	Label	To display the name of the Product Item
		IbIQuant ity	Label	To display the name of the Product Item
		nudQua ntity	Numeri c Up Down	To select the quantity of the Product Item
		SpecialP rice	Label	To display the label price of the Product Item
		IbISpecia IPrice	Label	To display the price of the Product Item
		ProductI tem_Cos t	Label	To display the label cost of the product item
<u> </u>	<u> </u>	L		<u> </u>

				btnRa oveC	m_ em	Label Button	To display the cost of the product item To remove an item from the cart.	
				[ALT]				
ALTERNATE	ALT Step 3: There syster	n fa	ils to retrieve the ca	rt items	5:			
Courses:	Element Name		Element Type		Not	es		
	Error		Label		This serves as a header for the modal.			
	There was an error loading the cart page. Please try again later.		Label	This is to show that there was an error while loading the cart.			error while	
	ОК		Button		None			
	Use Case Terminated							
Conclusion:	The use case concludes v	whe	en the relevant cart i	s displa	yed s	uccessfull	у.	
Post-condition:	User sees Cart view							
BUSINESS RULES:	None							
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None							
Assumptions:	None							
OPEN ISSUES:	None							

Author (s): Chardé Devine Vries Date: 06/06/2021

USE CASE NAME:	Remove Cart item	USE CASE TYPE						
USE CASE ID:	6.6		Business Re	quirements	s: 🗆			
PRIORITY:	High	System Ana	lysis:					
Source:	NKAP Bolting system		System Desi	gn:	$\overline{\checkmark}$			
PRIMARY BUSINESS	Customer							
Actor:								
PRIMARY THE	None							
SYSTEM ACTOR:								
OTHER	None							
PARTICIPATING								
ACTORS:								
OTHER INTERESTED	None							
STAKEHOLDERS:								
DESCRIPTION:	This use case describes the event where a customer would like remove an items from the cart. The use case starts when the customer clicks on the Remove icon found in the cart view. The system will remove information from the CartLine entity . The system displays all the relevant information. The use case concludes when the customer sees the successfully removed from cart message box.							
Pre-condition:	The customer must have an account a	nd be logged ir	nto the system					
TRIGGER:	A user wishes to remove an item from the	cart.						
Typical Course	SYSTEM RESPONSE:							
OF EVENTS:	Actor Action:	AUTOMATE	D ACTION					
	Step 1: The customer request to							
	remove an item from the cart.							
	Step 2: The customer clicks on the		Step 3: The system confirm that the					
	'Remove icon found next to the	customer wo	ould like to	remove the				
	item in the Cart view, cart table.	item from cart by displaying a						
		message box	K :					
			Element	Element	Notes:			
			name	type				
			Confirm	Header	То			
			Remove		indicate			
the state of the s			Kemove		illuicate			

	ProductIte m_ID	Product m_ID		The syste m remo ves the Produ ctlte m ID
Step 4: The customer clicks the Remove button ALT	Step 5: The system uses a SQL Delete to remove the Product Item from the cart in the CartLine table Element Element Notes Name Type			
	btnRemo ve	Button	they wou like rem the from To proceed and rem tem cart	ng if y uld to love item n cart ceed love n to cancel cess
			viev for	v is

	CartLine Quantity	CartLin	
	ALT Step 6: The successfully message bo	removed f	
	Element Name	Element Type	Notes
	Removed from cart	Header	To indicate what the view is for
	Removed from cart	Label	To inform the user that the item was successfull y removed from cart
	btnOkay	Button	To proceed

ALTERNATE	ALT Step 4: The user clicks	the cancel button.	Step 7: The system updates the cart icon in the page header. Using the CartLine_Quantity attribute in the CartLine entity to show the amount of items in the cart.		
Courses:	Use Case Terminated ALT Step 5: There system fa	m fails to remove the product item from cart:			
	Element Name	Element Type	Notes		
	Error	Label	This serves as a header for the modal.		
	There was an error while removing the product from cart.	Label	This is to show that removing the item from cart was unsuccessful.		
	Please try again. OK	Button	None		
	Use Case Terminated				
CONCLUSION:	The use case concludes who successfully.	en the removed cart	item message box is displayed		
Post-condition:	Cart is updated.				
BUSINESS RULES:	None				
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None				
Assumptions:	None				
OPEN ISSUES:	None				

NKAP Bolting System

Author (s): Chardé Devine Vries Date: 06/06/2021

Version:

___1.0___

USE CASE NAME:	Checkout Order USE CASE TYPE					
USE CASE ID:	6.7		Business Requirements:			
PRIORITY:	High		System Analysis:			
Source:	NKAP Bolting system		System Design:			
PRIMARY BUSINESS	Customer					
ACTOR:						
PRIMARY THE	None					
SYSTEM ACTOR:						
OTHER	Virtual Card Service					
PARTICIPATING						
ACTORS:						
OTHER INTERESTED	None					
STAKEHOLDERS:						
DESCRIPTION:	This use case describes the event where a customer would like to checkout all the items in their cart. The system checks if the customer is logged in. The system asks if they would like to pick up the order or if it should be delivered, which is followed by a delivery form that the customer has to fill in if they want it to be delivered. The system displays a modal with the order total. The customer requests to pay through the Virtual Card Service (PayPal). The transaction takes place on their system and once complete we save the transaction details on our system. The use case ends when the system displays a Payment successful popup.					
PRE-CONDITION:	The customer should h	ave product items in their ca	rt.			
TRIGGER:	A customer clicks the Check	cout button				
TYPICAL COURSE		S	YSTEM RESPONSE:			
OF EVENTS:	Actor Action:	MANUAL ACTION	AUTOMATED ACTION			
	Step 1: The customer request to go to checkout.					
	Step 2: The customer clicks the 'Checkout' button located in the Cart view.		Step 3: The system uses a S read in the Audit Trail entity if the customer is logged in	y to see		

	Element Name	Element Type	Notes
	AuditTrai I Desc	AuditTra il Desc	The system reads.
	ALT		
	Step 4: The customer to delivery and	select bet	ween
	Element Name	Element Type	Notes
	Checkout - Delivery or Collect	Label	This serves as a header for the modal.
	Delivery	Radio Button	None
	Collection	Radio Button	None
	Proceed	Button	If Delivery proceed to delivery form, if not proceed to step 9.
	Cancel	Button	None
	[ALT]		
Step 5: The customer selects the delivery	Step 6: The delivery for		olays

option and clicks the	Element	Elemen	Notes
proceed button.	Name	t Type	
ALT	Delivery form	Form	Saved to the: Address entity Provinc e Entity City Entity Suburb Entity
		ving eleme	ents are livery form:
	lblAddre ssLine1	Label	To display the Address_Lin e1 label
	txtAddr ess_Line 1	Textbo x	To enter the Address Line
	IblAddre ssLine2	Label	To display the Address_Lin e2 label
	txtAddr ess_Line 2	Textbo x	To enter the Address Line
	IblAddre ssLine3	Label	To display the Address_Lin e3 label
	txtAddr ess_Line 3	Textbo x	To enter the Address Line
	lblPostal Code	Label	To display the Postal code label

	txtPosta ICode	Textbo x	To enter the Postal Code
	IblProvi nce	Label	To display the label Province
	sltProvin ce	Dropdo wn	To select the Province
	IblCity	Label	To display the label City
	sltCity	Dropdo wn	To select the City
	lblSubur b	Label	To display the label Suburb
	sltSubur b	Dropdo wn	To select the Suburb
	btnProc eed	Button	Proceed to checkout
Step 7: The customer enters their address information and click the proceed button.	Step 8: The system uses a SQL insert to capture the AddressID using the Delivery table and generates a new deliveryID, the system also uses SQL Update to update the SaleOrder_ReceiveType in the Sale Entity.		
	Eleme nt Name	Attribute	Notes
	Addres sID		the Delivery table

				
		Deliver	Delivery_	The system
		yID	ID	generates
				a new
				Delivery_I
				D by
				retrieving
				the last
				Delivery_I
				D in the
				Delivery
				table and
				incrementi
				ng it by
				one in
				Delivery
				entity.
		SaleOr	SaleOrde	The system
		der_Re	r_Receiv	captures
		ceiveT	eType	and adds
			Стурс	the new
		ype		SaleOrder_
				ReceiveTyp
				e in Sale
				entity
				entity
				m validates
			formation u	ising a SQL
			rom the	
			y entity s entity	
			ce entity	
		• City en		
		• <u>Suburb</u>		
		Input	Attrib	ute Validatio
		Name	in Tab	
				Require
				ments
		txtAddre		_
		s_Line1	ine1	characters
	Į.			
		txtAddre s_Line2	es Addres ine2	s_L Max 100 characters

txtAddresss_Line3
txtPostalC Address ode Postal Code sltProvinc Province Must le Description selection n sltCity City_Desc Must le ription selection selec
ode Postal Code sltProvinc Province Must I e Descriptio selection sltCity City_Desc ription selection sltSuburb Suburb_D Must I escription selection ALT Step 9: The system displays a message box with the amount duand prompts the user to use
ode Postal Code sltProvinc Province_ Descriptio selection sltCity City_Desc ription selection sltSuburb Suburb_D Must I escription selection ALT Step 9: The system displays a message box with the amount duand prompts the user to use
sltProvinc e Descriptio selection n sltCity City_Desc ription selection sele
e Descriptio selected in sltCity City_Desc ription selected sltSuburb Suburb_D Must I escription selected secription selected sltSuburb Suburb_D are scription selected sltSuburb selected sltSuburb selected sltSuburb selected secription secription selected secription secription selected secription secr
e Descriptio selection sltCity City_Desc ription selection sltSuburb Suburb_D Must I escription selection ALT Step 9: The system displays a message box with the amount duand prompts the user to use
ALT Step 9: The system displays a message box with the amount duand prompts the user to use
ALT Step 9: The system displays a message box with the amount duand prompts the user to use
ALT Step 9: The system displays a message box with the amount duand prompts the user to use
SItSuburb Suburb_D Must I escription selected Step 9: The system displays a message box with the amount duand prompts the user to use
ALT Step 9: The system displays a message box with the amount du and prompts the user to use
ALT Step 9: The system displays a message box with the amount du and prompts the user to use
Step 9: The system displays a message box with the amount du and prompts the user to use
Step 9: The system displays a message box with the amount du and prompts the user to use
message box with the amount du and prompts the user to use
message box with the amount du and prompts the user to use
and prompts the user to use
available payment method.
Flowers Flowers Notes
Element Element Notes
Name Type
IblSubtot Label To
al display
subtota
Subtota
IbIVAT Label To
display
VAT
total
IbiTotal Label To
display
total
nrice
price
price Subtotal Label To
Subtotal Label To
Subtotal Label To display
Subtotal Label To display subtotal
Subtotal Label To display
Subtotal Label To display subtotal
Subtotal Label To display subtotal label VAT Label To
Subtotal Label To display subtotal label VAT Label To display of display subtotal label and display subtotal label to dis
Subtotal Label To display subtotal label VAT Label To

	Total	Label	To display total label	
	Checkout -Payment Method	Button	Virtual Card Services - Bank as option	
	Exit icon	Button	Invoke Use Case 6.1 Search Product s	
Step 10: The user clicks the 'Checkout with PayPal' button.	Step 11: The system loads the payment gateway.			
Step 12: The customer makes a payment through the Virtual card services	insert to rec	The system uses a SQL record the transaction in able and generates a new		
(PayPal).	Element		Notes	
	Name	e SaleID	The	
	SaleID	Juicib	The system generate s a new SaleID by retrievin g the last SaleID D in the Sale table and incremen	

			ting it by one.
	Customer ID	Custom er ID	The system captures and adds the SaleOrde r Desc
	Payment Type ID	Paymen tType ID	The system captures the Payment Type ID
	OrderStat us ID	OrderSt atus ID	The system captures the OrderSta tus ID
	SaleOrder Desc	SaleOrd er Desc	The system captures and adds the SaleOrde r Desc
	SaleOrder Date	SaleOrd er Date	The system captures

	Element Name	Element Type	Notes
	Step 14: The Payment su		
	ALT		
	Payment Amount	Paymen t Amount	current Date. The system captures and adds the Payment Amount by calculati ng the total of the cart items.
	Payment Date	Paymen t Date	using the current Date. The system captures and adds the Payment Date using the
			and adds the SaleOrde r Date

	Paymen	Label	To inform
	t		the user
	Success		that the
	full		payment
			was
			successful.
	btnOka	Button	То
	У		proceed.
	ALT		

ALTERNATE COURSES:

ALT Step 2: The customer is not logged in, the system displays the following:

Element Name	Element Type	Notes
Hi, please sign in or register as a new user.	Label	This serves as a header for the modal.
Sign in	Button	Invoke Use Case 1.1 Login
Register	Button	Invoke Use Case 3.1 Register as new customer
Exit icon	Button	Invoke Use Case 6.1 Search Products

ALT Step 4: The customer selects the cancel button

Terminate Use Case

ALT Step 8: The customer chooses Collection radio button.

The system uses SQL Update query to update SaleOrder_ReceiveType to 'collection' in the Sale Entity.

Proceed to step 7.

ALT Step 13: There system fails to update the delivery details:

Element Name	Element Type	Notes
Error	Label	This serves as a header for the modal.

	There was an error while updating the delivery details. Please try again. OK Use Case Terminated	Label	This is to show that updating the delivery details was unsuccessful.		
	ALT Step 14: There was an	error on the VCS system a	nd the payment failed.		
	Element Name	Element Type	Notes		
	Error	Label	This serves as a header for the modal.		
	Payment unsuccessful Please try again.	Label	This is to show that the payment was unsuccessful.		
	ОК	Button	Invoke Use Case 6.5		
Conclusion:	The use case concludes wh	nen the system displays a Pa	ayment successful message		
Post-condition:	Checkout successful – upd	ated SQL etc Sale entity			
BUSINESS RULES:	Customer needs to be logged in in order to make the payment.				
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None				
ASSUMPTIONS:	None				
OPEN ISSUES:	None				

2.1.7. REPORTS SUBSYSTEM

	NKAF	BOLTING	G SYSTEM			
DATE:	9/06/2021					
VERSION:	1.0					
AUTHOR(S)	Jerome Ame	enigy				
USE CASE NAME:	Generate fast- product list	selling	USE CAS	E TYPE		
USE CASE ID:	7.1		Business Requiremen	ts:□		
PRIORITY:	High		System Analysis:			
SOURCE:	NKAP Bolting Requirements		System Design:	₫		
PRIMARY BUSINESS ACTOR:	Admin					
PRIMARY THE SYSTEM ACTOR:	None					
OTHER PARTICIPATING ACTORS:	None					
OTHER INTERESTED STAKEHOLDERS:	None					
DESCRIPTION:	on the fast-se to the produc the report's m which the list	elling produts tab and nenu item i should be items which	s the event whereby a list acts by the admin. The act select the "fast-selling profucts tab and er generated. The system with had the most sales in the select in the sele	Imin will first navigate oduct list" report in ter the period for will then retrieve and		
PRE-CONDITION:		55	in to the system I on the system			
TRIGGER:	The admin was	ants to rev	iew the fastest selling iter	ns available on the		
TYPICAL COURSE	ACTOR		SYSTEM RESPO	NSE:		
OF EVENTS:	ACTION:	Manual	Automated Action			
	Step 1: The requests to generate a report					
	Step 2: The admin clicks on the Reports		Step 3: The system dis screen with the following on it:			

menu	Control	Control	Notes
option	Name	Туре	
	Reports	Heading	None
	Generate Report	Label	None
	Report type	Label	A dropdown list containing all the different reports available
	Period start date	Date/time selector	The control to select the report start date
	Period end date	Date/time selector	The control to select the report end date
	Generate	Button	Button used to generate the selected report.
	Report Name	Label	A label where the name of the generated report will be displayed.
	Download	Button	Button used to download the generated reports
	Report type	Dropdown	A dropdown list containing all the different reports available
	Please select the	Label	None.

	report		
	period		
	Report Panel	Panel	Area where the generated report will be displayed.
	Start Date	Label	None.
	End Date	Label	None.
Step 4: The admin selects the "Fast-selling product list"			
Step 5: The admin selects the required period.			
Step 6: The admin clicks the "Generate" button			
	SQL Read. Production: Product Product Categor Sale: Sale_ID SaleOrd SaleOrd	m the databas Item_ID Item_Name Item_Desc ryType_ID	s the required e tables using a
	• SaleLine • SaleLin • Product	e_ID :ltem_ID	

		Sale_IDSaleLine_Qunatity
		Step 8: The system compiles and generates the retrieved information into the Fast-selling product list.
		Step 9: The system displays the Fast-selling product list in the report panel
ALTERNATE COURSES		
CONCLUSION:	A list based on the admin	the fastest selling items is generated and displayed to
POST-CONDITION:	The admin h	as viewed the generated fast-selling product list
BUSINESS RULES:	1. The list is bas	sed on the sales made through the system
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None	
ASSUMPTIONS:	None	
OPEN ISSUES:	None	

NKAP BOLTING SYSTEM					
DATE:	9/06/2021				
VERSION:	1.0				
AUTHOR(S)	Jerome Amenigy				
USE CASE NAME:	Generate slow-selling product list	USE CASE TYPE			
USE CASE ID:	7.2	Business Requirements:□			
PRIORITY:	High	System Analysis: □			
SOURCE:	NKAP Bolting System Requirements	System Design: ☑			
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				

DESCRIPTION:	This use case describes the event whereby a list is generated based on the slow-selling products by the admin. The admin will first navigate to the products tab and select the "slow-selling product list" report in the report's menu item in the products tab and enter the period for which the list should be generated. The system will then retrieve and display those items which had the lowest amount sales in the period the admin has selected.					
PRE-CONDITION:			on the system. in to the syster			
TRIGGER:		ants to review n to other item		n are not being	sold very often	
TYPICAL COURSE	ACTOR		SYSTEM	RESPONSE:		
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Act	ion		
	Step 1: The requests to generate a report					
	Step 2: The admin clicks on the Reports menu option			the "Reports" uts and controls		
			Control Name	Control Type	Notes	
	option		Reports	Heading	None	
			Generate Report	Label	None	
			Report type	Label	A dropdown list containing all the different reports available	
			Period start date	Date/time selector	The control to select the report start date	
			Period end date	Date/time selector	The control to select the report end date	
			Generate	Button	Button used to generate	

			the selected report.
	Report Name	Label	A label where the name of the generated report will be displayed.
	Download	Button	Button used to download the generated reports
	Report type	Dropdown	A dropdown list containing all the different reports available
	Please select the report period	Label	None.
	Report Panel	Panel	Area where the generated report will be displayed.
	Start Date	Label	None.
	End Date	Label	None.
Step 4: The admin selects the "Slow- selling product list"			
Step 5: The admin selects the			

	required period.		
	Step 6: The admin clicks the "Generate" button		
			Step 7: The system retrieves the required information from the database tables using a SQL Read.
			ProductItem:
			 ProductItem_ID
			 ProductItem_Name
			 ProductItem_Desc
			CategoryType_ID
			Sale:
			Sale_ID
			SaleOrder_NumberSaleOrder_Date
			SaleOrder_Date SaleOrder_Desc
			SaleLine
			SaleLine_ID
			ProductItem_IDSale_ID
			Sale_ib SaleLine_ Qunatity
			Step 8: The system compiles and generates the retrieved information into the Slow-selling product list.
			Step 9: The system displays the Slow-selling product list in the report panel
ALTERNATE COURSES			
CONCLUSION:	A list based of admin	on the slowest	selling items is generated and displayed to the
POST-CONDITION:	• The ac	dmin has view	ed the generated slow-selling product list
BUSINESS RULES:	1. The list is l	pased on the s	sales made through the system
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None		
ASSUMPTIONS:	None		
OPEN ISSUES:	None		

	NKAP BOL	TING SYS	TEM		
DATE:	9/06/2021				
VERSION:	1.0				
AUTHOR(S)	Jerome Amenigy				
USE CASE NAME:	Generate most frequent be report	ouyers	U	SE CASE TY	PE
USE CASE ID:	7.3		Business Ro	equirements	s :□
PRIORITY:	High		System Ana	lysis:	
SOURCE:	NKAP Bolting System Requirements		System Des	ign:	
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	This use case describes based on the Customers will retrieve information display a list of the custo the past month	s who hav from the C	e placed the n Customer and	nost orders. Sale tables a	The system and then
PRE-CONDITION:	Sales have occurThe admin is logo		•		
TRIGGER:	The admin wants to view from the business.	w who the	most frequent	customers a	are buying
TYPICAL COURSE			SYSTEM	RESPONSE	:
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Ad	ction	
	Step 1: The requests to generate a report				
	Step 2: The admin clicks on the Reports menu option Step 3: The system displays the "Reports" screen with the following inputs and controls on it:				
			Control Name	Control Type	Notes
			Reports	Heading	None

Generate Report	Label	None
Report type	Label	A dropdown list containing all the different reports available
Period start date	Date/time selector	The control to select the report start date
Period end date	Date/time selector	The control to select the report end date
Generate	Button	Button used to generate the selected report.
Report Name	Label	A label where the name of the generated report will be displayed.
Download	Button	Button used to download the generated reports
Report type	Dropdown	A dropdown list containing all the different

					reports available
			Please select the report period	Label	None.
		Report Panel	Panel	Area where the generated report will be displayed.	
		Start Date	Label	None.	
			End Date	Label	None.
	Step 4: The admin selects the "Most frequent buyer report"				
	Step 5: The admin clicks the "Generate" button		 SaleO Delive Courie Payme Custor 	D rder_Date rder_Desc ry_ID er_ID ent_Date mer_ID mer_ID mer_Surname mer_Cellphor mer_Busines	neNumber Idress name
			From the Salusing a SQL		<u>ner</u> tables

			Step 7: The system sorts through the retrieved information and filters for the top results by using a count
			Step 8: The system displays the most frequent buyers report
ALTERNATE COURSES			
CONCLUSION:	The list of the most frequency	ient buye	rs has been generated and displayed.
POST-CONDITION:	The admin has view.	ewed the	generated report
BUSINESS RULES:	1. None		
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None		
ASSUMPTIONS:	None		
OPEN ISSUES:	None		

NKAP BOLTING SYSTEM				
DATE:	9/06/2021	9/06/2021		
VERSION:	1.0			
AUTHOR(S)	Jerome Amenigy			
USE CASE NAME:	Generate most popular location report	USE CASE TYPE		
USE CASE ID:	7.4	Business Requirements:□		
PRIORITY:	High	System Analysis: □		
SOURCE:	NKAP Bolting System Requirements	System Design: ☑		
PRIMARY BUSINESS ACTOR:	Admin			
PRIMARY THE SYSTEM ACTOR:	None			
OTHER PARTICIPATING ACTORS:	None			
OTHER INTERESTED STAKEHOLDERS:	None			
DESCRIPTION:		ess where the admin generates a list nost purchases were made through the		

PRE-CONDITION:	Sales must have occurred over the past month				
TRIGGER:	The admin wants to vie customers are ordering		based on the	locations wh	ere
TYPICAL COURSE			SYSTEM	RESPONSE	:
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Ad	ction	
	Step 1: The requests to generate a report				
	Step 2: The admin clicks on the Reports menu option		Step 3: The "Reports" scrinputs and co	reen with the	
			Control Name	Control Type	Notes
			Reports	Heading	None
			Generate Report	Label	None
			Report type	Label	A dropdown list containing all the different reports available
			Period start date	Date/time selector	The control to select the report start date
			Period end date	Date/time selector	The control to select the report end date
			Generate	Button	Button used to generate the selected report.
			Report Name	Label	A label where the name of the

			generated report will be displayed.
	Download	Button	Button used to download the generated reports
	Report type	Dropdown	A dropdown list containing all the different reports available
	Please select the report period	Label	None.
	Report Panel	Panel	Area where the generated report will be displayed.
	Start Date	Label	None.
	End Date	Label	None.
Step 4: The admin selects the "most popular location			
report" Step 5: The admin clicks the "Generate" button	CustorCustorCustorCustor	om the <u>Custor</u> mer_ID mer_Name mer_Surname mer_Cellphor mer_EmailAd mer_Busines	ner table e ne Idress

	T	
		SaleOrder_Date SaleOrder_Date
		SaleOrder_Desc
		From the <u>Sale</u> table,
		Address ID
		Customer_ID
		Province_ID
		Address Line1
		Address_Line2
		 Address_Line3
		 Address_PostalCode
		From the Address table,
		Province_ID
		 Province_Description
		From the Province entity,
		City_ID
		Province_ID
		City_Description
		From the city entity,
		Trom the only entity,
		Suburb_ID
		City_ID
		Suburb_Description
		From the Suburb table using a SQL
		Read
		Step 7: The system sorts through the
		retrieved information and filters for the
		areas which had the most sales in the
		past month
		Step 8: The system displays the most popular location report
ALTERNATE		
CONCLUSION:		has been generated and displayed
POST-CONDITION:	The admin views the most	popular location report
BUSINESS RULES:	1. None	
IMPLEMENTATION		
CONSTRAINTS	None	
AND SPECIFICATIONS:		
ASSUMPTIONS:	None	
OPEN ISSUES:	None	

	NKAP BOL	TING SYS	TEM		
DATE:	9/06/2021				
VERSION:	1.0				
AUTHOR(S)	Jerome Amenigy				
USE CASE NAME:	Generate Weekly Sale Ord	der List	USE	CASE TY	PE
USE CASE ID:	7.5		Business Req	uirements	::□
PRIORITY:	High		System Analys	sis:	
SOURCE:	NKAP Bolting System Requirements		System Desig	n:	Ø
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	This use case describes the event where the admin generates a report on the sale orders which have occurred over the past week made by customers. The admin will first navigate to the reports tab and then select reports and select the "weekly sale order list" report and then the system will retrieve and display the list based on all the sale orders which have taken place over the past week.			de by d then select n the system	
PRE-CONDITION:	Sales have occurThe admin is logg		•	ast week	
TRIGGER:	The admin wants to view past week.	wall the sa	ale which have ta	aken place	over the
TYPICAL COURSE			SYSTEM RI	ESPONSE	:
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Action	on	
	Step 1: The requests to generate a report				
	Step 2: The admin clicks on the Reports menu option		Step 3: The sy "Reports" screet inputs and cont	en with the	•
				Control Type	Notes

	Reports	Heading	None
	Generate Report	Label	None
	Report type	Label	A dropdown list containing all the different reports available
	Period start date	Date/time selector	The control to select the report start date
	Period end date	Date/time selector	The control to select the report end date
	Generate	Button	Button used to generate the selected report.
	Report Name	Label	A label where the name of the generated report will be displayed.
	Download	Button	Button used to download the generated reports
	Report type	Dropdown	A dropdown list containing all the

		Please	Label	different reports available
		select the report period	Labor	Trone.
		Report Panel	Panel	Area where the generated report will be displayed.
		Start Date	Label	None.
		End Date	Label	None.
Step 4: The admin selects the "Weekly Sale Order List"				
Step 5: The admin clicks the "Generate" button		SaleOCustorPayme		:
		CustorCustorCustorCustor	mer_ID mer_Name mer_Surnam mer_Cellphoi mer_EmailAo mer_Busines	ne Idress
		From the Salusing a SQL		ner tables
		Step 7: The s Sales which I week and put correspondin those sale or	have occurre ts them in a li g customers	d in the past ist with the who made
		Step 8: The sweekly sale of		

ALTERNATE	
CONCLUSION:	The weekly sale order list has been generated and displayed
POST-CONDITION:	The admin views the generated sale order list
BUSINESS RULES:	1. None
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

	NKAP BOLTING SYS	STEM_	
DATE:	9/06/2021		
VERSION:	1.0		
AUTHOR(S)	Jerome Amenigy		
USE CASE NAME:	Generate Monthly Sale Order List	USE CASE TYPE	
USE CASE ID:	7.6	Business Requirements:□	
PRIORITY:	High	System Analysis: □	
SOURCE:	NKAP Bolting System Requirements	System Design: ☑	
PRIMARY BUSINESS ACTOR:	Admin		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATING ACTORS:	None		
OTHER INTERESTED STAKEHOLDERS:	None		
DESCRIPTION:	This use case describes the event where the admin generates a report on the sale orders which have occurred over the past month made by customers. The admin will first navigate to the reports tab and then select reports and select the "monthly sale order list" report and then the system will retrieve and display the list based on all the sales which have taken place over the past month		
PRE-CONDITION:	Sales have occurred over tThe admin is logged in to t	'	

TRIGGER:	The admin wants to view all the sales which have taken place over the past month.						
TYPICAL COURSE			SYSTEM RESPONSE:				
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Action				
	Step 1: The requests to generate a report			The system displays the screen with the following d controls on it: Control Notes Type			
				een with the	the following		
			Control Name		Notes		
			Reports	Heading	None		
			Generate Report	Label	None		
			Report type	Label	splays the he following it: Notes None None A dropdown list containing all the different reports available The control to select the report start date The control to select the report end date Button used to generate the selected report. A label where the		
				Period start date		control to select the report	
			Period end date		control to select the report end		
			Ger	Generate	Button	used to generate the selected	
			Report Name	Label			

			be displayed.	
	Download	Button	Button used to download the generated reports	
	Report type	Dropdown	A dropdown list containing all the different reports available	
	Please select the report period	Label	None.	
	Report Panel	Panel	Area where the generated report will be displayed.	
	Start Date	Label	None.	
	End Date	Label	None.	
Step 4: The admin selects the "Monthly Sale Order List"				
Step 5: The admin clicks the "Generate" button	Sale_ISaleOSaleOCustor	Customer_IDPayment_Date		
	CustorCustor			

		Customer_EmailAddressCustomer_Businesname			
		From the Sale and Customer tables using a SQL read			
		Step 7: The system then selects the Sales which have occurred in the past month and puts them in a list with the corresponding customers who made those sale orders also in the list.			
		Step 8: The system then displays the monthly sale order list to the admin			
ALTERNATE					
CONCLUSION:	The weekly sale order list has been generated and displayed				
POST-CONDITION:	The admin views the generated sale order list				
BUSINESS RULES:	1. None				
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None				
ASSUMPTIONS:	None				
OPEN ISSUES:	None				

NKAP BOLTING SYSTEM					
DATE:	9/06/2021				
VERSION:	1.0				
AUTHOR(S)	Jerome Amenigy				
USE CASE NAME:	Generate Stock-Level Report USE CASE TYPE				
USE CASE ID:	7.7	Business Requirements:□			
PRIORITY:	High	System Analysis: □			
SOURCE:	NKAP Bolting System Requirements	System Design: ☑			
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				

OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	This use case describes the event where a stock-level report is generated by the admin. The admin navigates to the reports tab and chooses the generate stock-level report. The system will retrieve information from the ProductItem entity and display it to the admin. The use case ends once the report has been displayed to the admin.				
PRE-CONDITION:	 Products have been added to the system The admin is logged in 				
TRIGGER:	The admin wants to view	w the stoc	k available in t	he system	
TYPICAL COURSE			SYSTEM	RESPONSE	
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Ad	tion	
	Step 1: The requests to generate a report				
	Step 2: The admin clicks on the Reports menu option		Step 3: The system displays the "Reports" screen with the following inputs and controls on it:		
			Control Name	Control Type	Notes
			Reports	Heading	None
			Generate Report	Label	None
			Report type	Label	A dropdown list containing all the different reports available
			Period start date	Date/time selector	The control to select the report start date
			Period end date	Date/time selector	The control to select the report end date

	Generate	Button	Button used to generate the selected report.
	Report Name	Label	A label where the name of the generated report will be displayed.
	Download	Button	Button used to download the generated reports
	Report type	Dropdown	A dropdown list containing all the different reports available
	Please select the report period	Label	None.
	Report Panel	Panel	Area where the generated report will be displayed.
	Start Date	Label	None.
	End Date	Label	None.
Step 4: The admin selects the "Stock-Level Report"			

	Step 5: The admin clicks the "Generate" button	Step 6: The system retrieves ProductItem_ID ProductItem_Name ProductItem_Desc ProductItem_Image CategoryType_ID Quantity on hand CategoryType_Desc ProductCategory_ID ProductCategory_ID ProductCategory_Desc From the ProductItem, CategoryType and ProductCategory tables using a SQL Read. Step 7: The system categorizes each ProductItem into its Category type and each category type into its Product Category. Step 8: The system then displays the Stock-level report with the necessary		
ALTERNATE		columns.		
ALTERNATE	The steel level remark is seen and to	ad and diaplayed		
CONCLUSION: POST-CONDITION:	The admin views the steel			
BUSINESS RULES:	The admin views the stock-level report A Name			
IMPLEMENTATION	1. None			
CONSTRAINTS AND SPECIFICATIONS:	None			
ASSUMPTIONS:	None			
OPEN ISSUES:	None			

NKAP BOLTING SYSTEM				
DATE:	9/06/2021			
VERSION:	1.0			
AUTHOR(S)	AUTHOR(S) Jerome Amenigy			
USE CASE NAME:	Packing Report	USE CASE TYPE		

USE CASE ID:	7.8		Business Re	quirements	:□	
PRIORITY:	High		System Ana	lysis:		
SOURCE:	NKAP Bolting System Requirements		System Desi	ign:	Ø	
PRIMARY BUSINESS ACTOR:	Admin					
PRIMARY THE SYSTEM ACTOR:	None					
OTHER PARTICIPATING ACTORS:	None					
OTHER INTERESTED STAKEHOLDERS:	None					
DESCRIPTION:	This use case describes the event whereby an admin generates a report based on the Sales which still need to be packed. Firstly, the admin requests to generate a packing report on the system. The system will then retrieve all the Sales which have a status of ready to be packed. The system will then populate and display a list with the SaleOrder details.					
PRE-CONDITION:	Sales have taken	place on	the system			
	The admin is logg	ged in to th	ne system			
TRIGGER:	The admin wants to view	w all the or	rders that still r	needs to be p	acked	
TYPICAL COURSE			SYSTEM	RESPONSE:		
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Ac	tion		
	Step 1: The requests to generate a report					
	Step 2: The admin clicks on the Reports menu option		Step 3: The s "Reports" scr inputs and co	een with the	•	
			Control Name	Control Type	Notes	
			Reports	Heading	None	
			Generate Report	Label	None	
			Report type	Label	A dropdown list containing all the different	

Period Date/time The	e
	_
start date selector control to select the report start date.	ne
Period Date/time The end date selector control to select the report end date.	ne
Generate Button Button used to generate the selected report.	e
Report Name Label A label where the name of the generate report where the displayer displayer is a second control of the second con	f ed vill
Download Button Button used to download the generate reports	ad
Report type Dropdown A dropdow list containing all the different reports available	ing t
Please Label None. select the report period	
Report Panel Area where the	he

			Start Date End Date	Label Label	generated report will be displayed. None.
	Step 4: The admin selects the Packing Report"				
	Step 5: The admin clicks the "Generate" button		SaleOSaleOSaleOSaleOPaymePaymeAndOrderS	D Status_ID rder_Desc rder_Date rder_Assign rder_Receive ent_Date ent_Amount Status_ID Status_Desc e and OrderS have the ord cking" using a system sorts e first made to	Etatus der status of a SQL Read. the retrieved to the most
ALTERNATE			<u> </u>		
CONCLUSION:	The Packing report is generated from the Sales which have a status of "ready for packing"				
POST-CONDITION:	The system generates and displays the packing report				
BUSINESS RULES:	The packing report is generated from the Sale Orders which have a status of "ready for packing"				
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None				

ASSUMPTIONS:	None
OPEN ISSUES:	None

	NKAP BOL	TING SYS	STEM		
DATE:	9/06/2021				
VERSION:	1.0				
AUTHOR(S)	Jerome Amenigy				
USE CASE NAME:	Delivery Report		USE CASE TYPE		
USE CASE ID:	7.9		Business Requirements:□		
PRIORITY:	High		System Analysis: □		
SOURCE:	NKAP Bolting System Requirements		System Design: ☑		
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	This use case describes the event whereby an admin generates a report based on the Sales which still need to be delivered. The admin will first request to generate a delivery report on the system. The system will then retrieve the Sale Orders with an Order Status of "ready for delivery"				
PRE-CONDITION:	Sales have taken place on the system				
	The admin is logged in to the system				
TRIGGER:	The admin wants to view delivered	wall the s	ale orders that still needs to be		
TYPICAL COURSE	SYSTEM RESPONSE:				
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Action		
	Step 1: The requests to generate a report				
	Step 2: The admin clicks on the Reports menu option		Step 3: The system displays the "Reports" screen with the following inputs and controls on it:		
			Control Control No Name Type	tes	

Reports	Heading	None
Generate Report	Label	None
Report type	Label	A dropdown list containing all the different reports available
Period start date	Date/time selector	The control to select the report start date
Period end date	Date/time selector	The control to select the report end date
Generate	Button	Button used to generate the selected report.
Report Name	Label	A label where the name of the generated report will be displayed.
Download	Button	Button used to download the generated reports
Report type	Dropdown	A dropdown list containing all the

	Please select the report period Report Panel	Label	different reports available None. Area where the generated report will be displayed.
	Start Date	Label	None.
	End Date	Label	None.
Step 4: The admin			
selects the Delivery Report"			
Step 5: The admin clicks the "Generate" button	 SaleO SaleO SaleO SaleO Payme Payme OrderS OrderS OrderS OrderS Addres Custom Provinc Addres Addres Addres Addres Addres 	D Status_ID rder_Desc rder_Date rder_Assign rder_Receive ent_Date ent_Amount Status_ID Status_Desc ry_ID r_ID ry_Date s_ID ner_ID	еТуре

		Province_ID		
		Province_Description		
		, City_ID		
		• Province_ID		
		City_Description		
		,		
		Suburb_ID		
		• City_ID		
		 Suburb_Description 		
		And		
		Customer_ID		
		Customer_Name		
		from the Sale, OrderStatus, Delivery,		
		Address, Province, City, Suburb and Customer entities which have the		
		order status of "ready for delivery"		
		using a SQL Read.		
		Step 7: The system then filters		
		through the retrieved sales and sorts		
		them according to SaleOrder_Date.		
		Step 4: The system then displays the		
		list of retrieved Sales to the admin in		
		the correct order as the Delivery Report		
ALTERNATE		ιτεροιτ		
CONCLUSION:	The delivery report is ge tables and displayed to	nerated from the Sale, Delivery and OrderStatus the admin.		
POST-CONDITION:	The system has generated and displayed the delivery report			
BUSINESS RULES:	The Sales displayed in the report must have an order status of "ready for delivery"			
IMPLEMENTATION				
CONSTRAINTS	None			
AND				
SPECIFICATIONS:				
ASSUMPTIONS:	None			
OPEN ISSUES:	None			

2.1.8. DELIVERY SUBSYSTEM

NKAP Bolting System

Author (s): Chardé Devine Vries Date: 06/06/2021

Version:

___1.0___

USE CASE NAME:	Search delivery details		USE CASE TYPE			
USE CASE ID:	8.1		Business Requirements:	*		
PRIORITY:	High		System Analysis: *			
Source:	NKAP Bolting system		System Design:			
PRIMARY BUSINESS	Employee					
ACTOR:						
PRIMARY THE	None					
SYSTEM ACTOR:						
OTHER	None					
PARTICIPATING						
ACTORS:						
OTHER INTERESTED	None					
STAKEHOLDERS:						
DESCRIPTION:	The system retrieves informat displays the list of deliveries to	tion from the <u>De</u> o the employee	mployee views a specific delivery elivery entity, <u>Address entity</u> and . The employee selects a delivery n has successfully displayed the D	Courier entity and and the details are		
PRE-CONDITION:	The employee must have	an account and	be logged into the system			
TRIGGER:	A employee wants to view del	liveries				
TYPICAL COURSE			SYSTEM RESPONSE:			
OF EVENTS:	Actor Action:	MANUAL ACTION	AUTOMATED ACTION			
	Step 1: The employee request to view deliveries.					
	Step 2: The employee clicks on the 'Delivery'		Step 3: The system loads t screen with the following of	=		

tab from the menu-	Element	Elemen	Notes:
strip.	Name	t Type	Notes.
	Deliveri	Header	To indicate what the
			screen is
			for
	tblDeliv eries	Table	To show each delivery. Retrieved from the: Delivery entity Courier entity Address entity Province entity City entity Suburb entity
			• OrderStat us
	Back	Button	
	The follow	ving eleme	ents are
	displayed	in <i>tblDeli</i>	veries :
	Delivery	Table Headin g	To display the headings
	Delivery ID	Table column Headin g	Row contains: Delivery_I D
	Trackin	Table	Row
	g Number	column Headin g	contains:Courier_T rackingNu mber
	Delivery Date	Table column	Row contains:

	Headin g	Delivery_Date
	Table column Headin g	Row contains: Courier_ Name Courier_ Number Courier_E mail
	Table column Headin g	Row contains: Address_ Line1 Address_ Line2 Address_ Line3 Address_ PostalCod e Province_ Descripti on City_Desc ription Suburb_D escription
	Table column Headin g	Row contains: OrderStat us_Desc button
Order Status icon button	Button	To show order status and update when clicked.
St ico bu	tatus on utton	rder Button tatus on

ALT Step 3: The system fails to load delivery page:							
Courses:	Element Name	Element Type	Notes				
	Error	Label	This serves as a header for the modal.				
	There was an error loading the Delivery page.	Label	This is to shows that loading the page was unsuccessful.				
	Please try again later.						
	ОК	Button	None				
	Use Case Terminated						
Conclusion:	The use case concludes when the delivery view displays.						
Post-condition:	The customer can see the Delivery screen						
BUSINESS RULES:	None						
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None						
Assumptions:	None	None					
OPEN ISSUES:	None						

NKAP BOLTING SYSTEM				
DATE:	13/06/2021			
VERSION:	1.0			
AUTHOR(S)	Sonali Badrinath			
USE CASE NAME:	Search Courier USE CASE TYPE			
USE CASE ID:	8.4. Business Requirements:			

NKAP BOLTING SYSTEM						
DATE:	13/06/2021					
VERSION:	1.0					
AUTHOR(S)	Sonali Badrina	Sonali Badrinath				
USE CASE NAME:	Add Courier USE CASE TYPE					
USE CASE ID:	8.3.		Business Requirements: □			
PRIORITY:	Medium		System Analysis: □			
SOURCE:	NKAP Bolting System Requirements		System Design: ☑			
PRIMARY BUSINESS ACTOR:	Administrator					
PRIMARY THE SYSTEM ACTOR:	None					
OTHER PARTICIPATIN G ACTORS:	None	None				
OTHER INTERESTED STAKEHOLDE RS:	Courier					
DESCRIPTION:	new courier or admin request courier details	n to the s ts to add and the	es the event where the admin wants to add a system. The use case begins when the I the courier. The admin will enter all the information will be captured in the Courier ands when the courier has been successfully			
PRE- CONDITION:	The admin	should	be logged onto the system.			
TRIGGER:	The admin red	questing	to add a new courier.			
TYPICAL			SYSTEM RESPONSE:			
COURSE OF EVENTS:	ACTOR ACTION: Manu al Automated Action Actio					
	Step 1: The admin requests to	Step 1: The admin				
PRIORITY:	High		System Analysis:			
SOURCE:	NKAP Bo System Requirem	J	System Design: ☑			

	add a new	T					
	courier.						
Step 2 admin click o "Add	Step 2: The admin will click on the "Add	Step 2: The admin will click on the	Step 3: The system responds by loading the "Add Courier" modal with the following controls:				
	Courier" menu item on the		Control Name	Control Type	Notes		
	navigation bar.		Add Courier	Heading	None		
			Courier Name:	Label	This is the label for the name of the courier		
			Name:	Textbox	None		
			Courier Type:	Label	This is the label for the type of the courier		
				Type:	Dropdown	Populated with different titles to select.	
						Courier Number:	Label
			Number:	Textbox	None		
			Courier Email:	Label	This is the label for the email of the courier		
			Email:	Textbox	None		
			Add	Button	None		
			Cancel	Button	This button is used when the admin decides to terminate the process of adding a new courier.		
	Step 4: The admin enters the details in						

	the respective places: Name Type Numb er Email Step 5: The admin clicks on the "Add" button [Alt]		-	plays a modal for addition of the
	button [Ait]	Input Name	Control Type	Notes
		Confirm to Proceed	Label	This serves as a header for the modal.
	Would you like to proceed with the addition of this courier?	Label	This is to allow the admin to finalize the decision to submit the addition of the courier with the details provided.	
		Yes	Button	Proceeds to the next step (Step 7)
Step 7: The admin selects the "Yes" button. [Alt]		No	Button	Allows user to enter courier details again.
	admin selects the "Yes"	Step 8: The sy the information against the Co	n entered l	=
	Dutton. [Ait]	Input Attr Name	ibute in ta	ble Validation requirement s

	Courier Name	Courier_Name	Required
	Type	CourierType_De sc	Required
	Numbe r	Courier_Number	Required, 10 digits
	Email	Courier_Email	Required, maximum of 200 characters
	query to s	ne system uses a ave the captured er table [Alt]:	
	Informati on to	Attribute in table	Details
	be saved:		
	Courier ID		Generated by the system by reading the last Courier_ID from the Courier table and adding it by one.
	Courier Name	Courier_Na me	None
	Courier Type	_ID	Match the CourierType_I D of the selected CourierType_ Desc
	Courier Number	Courier_Nu mber	None
	Courier Email	Courier_Ema il	None

Step 10: The system saves the new courier
details in the relevant entities and displays
a success modal:

Input Name	Control Type	Notes
Success	Modal Heading	None
Courier information was successfully created.	Label	Label to indicate the courier information is saved.
Ok	Button	None

ALTERNATE COURSES:

[Alt Step 5]: The admin clicks on the "Cancel" button which terminates this use case.

Return to step 4

[Alt Step 7]: The admin clicks on the "No" button which makes the admin Return to step 4.

[Alt Step 8]: The details entered were not in the right format, so the validation failed. The system will display a validation unsuccessful modal to the admin.

Input Name	Control Type	Notes
VALIDATION UNSUCCESSFUL	Label	This serves as a header for the modal.
The information provided in the given parameters is not in the correct format.	Label	This is to show that the courier information was incorrectly entered.
BACK	Button	This button allows the admin the return to the 'Add Courier' screen to edit the entered information in the correct format.

After the admin clicks on the "**Back**" button the system will show the admin where the validation errors occurred by showing a red outline where the validation failed.

	The	[Alt Step 10]: The new courier could not be added to the system. The system will display a modal informing the admin that the saving of the courier information was unsuccessful.				
	Inpi	ut Name	Control Type	Notes		
	UN	SUCCESSFUL	Label	This serves as a header for the modal.		
inf		ed to add new courier rmation. Please try iin later.	Label	This is to show that the saving of the courier information was unsuccessful.		
	OK		Button	None		
CONCLUSION:		This use case concludes when the courier has been successfully added on the system.				
POST- CONDITION:	• T	he courier has been ad	ded to the	Courier table.		
BUSINESS RULES:	Only	an admin can add a ne	w courier to	o the system		
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIO NS:	None					
ASSUMPTION S:	None					
OPEN ISSUES:	None					
PRIMARY BUSINESS ACTO	CTOR: Admin					
PRIMARY THE SYSTEM ACTOR	PRIMARY THE SYSTEM ACTOR:					
OTHER PARTICIPATING ACTORS:	None					

OTHER INTERESTED STAKEHOLDERS: DESCRIPTION: PRE-CONDITION:	This use case describes the event where the admin wishes to search for a courier. The admin will enter the search parameters like the suppler name. The system will search for the courier based on the parameters the admin entered. This use case concludes when the courier is shown the results of the search query. • The admin should be logged on to the system.				
			ıld be already exis		stem.
TRIGGER:	The admin	wishes to	o search for a cou		
TYPICAL COURSE OF EVENTS:	ACTOR		SYSTEM RE		:
OF EVENTS:	ACTION:	Manual Action	Automated Actio	n	
	Step 1: The admin would like to search for a courier.				
	Step 2: The admin clicks on the		Step 3: The system the "Search Country following controls:	urier" scree s:	en with the
	"Search Courier "		Control Name	Control Type	Notes
	screen.		Search Courier	Heading	None
			Courier Name:	Label	This is the label to prompt the user to enter a search query.
			Search Name:	Textbox	None

Search icon	Icon button	None
Courier Table	Table	This is to show the Couriers that are saved in the database in a table.
Name	Column	This is the column for the Courier name
Туре	Column	This is the column for the courier type
Number	Column	This is the column for the number of the courier
Email	Column	This is the column for the emthe courier
Options	Icon	This is to show the update and delete button for each row.
Update	Button	Present in Options dropdown. Used to update the information of a courier.
Delete	Button	Present in Options

Step 4: The admin inputs the search	dropdown. Used to delete courier details. Step 5: The system captures and validates the information against the Courier table [Alt]: Input Attribute Validation Name Name Requirments	
	criteria and clicks on the search	Search Courier_Name Maximum of Name: 50 characters
	icon.	
		Step 6: A SQL read query is used to read the following attributes from the Courier table based on the search criteria:
		Courier table:
		 Courier_ID(PK) Courier_Name attribute information displayed in the Name column.
		 Courier_Type attribute information displayed in the Type column.
		 Courier_Number attribute information displayed in the Number column.
		 Courier_Email attribute information displayed in the Email column.
		Step 7: The system will show the search results received from step 6 in a table. [Alt]
ALTERNATE	the validation fa requested infor	etails entered were not in the right format, so called or there are not any existing data on the remation. The system will display a validation nodal to the admin.

	Input Name	Control Type	Notes		
	Error	Label	This serves as a header for the modal.		
	No matches found. Please check the format of the entered information.	Label	This is to show that there were no matches found.		
	OK	Button	None		
CONCLUSION:	The use case concludes when the admin receives the results of the searched criteria.				
POST-CONDITION:	The admin receives the results of the couriers with the same searched criteria				
BUSINESS RULES:	1. None				
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None				
ASSUMPTIONS:	None				
OPEN ISSUES:	None				

NKAP BOLTING SYSTEM			
DATE:	13/06/2021		
VERSION:	1.0		
AUTHOR(S)	Sonali Badrinath		

USE CASE	Update Co	urier	USE CASE TYPE
NAME: USE CASE ID:	0.5		Pusinoss Poquiroments: □
PRIORITY:	8.5.		Business Requirements:
	3	n.a	System Analysis: □
SOURCE:	NKAP Bolti System Requiremen		System Design: ☑
PRIMARY BUSINESS ACTOR:	Admin		
PRIMARY THE SYSTEM ACTOR:	None		
OTHER PARTICIPATIN G ACTORS:	None		
OTHER INTERESTED STAKEHOLDE RS:	None		
DESCRIPTION:	update the previous inf will add the	details of formation new info	ribes the event in which an admin wants to of a courier. The system will display the of that was added before and then the admin ormation. This use case ends when the of updated successfully.
PRE- CONDITION:	The adn	nin shou	ld be logged on to the system.
TRIGGER:	The admin	wishes t	o update the details of the courier.
TYPICAL			Actor Action:
COURSE OF EVENTS:	Actor Action:	Manu al Actio n	Automated Action
	Step 1: The admin requests to update the details of an employee by selecting		Step 2: The system invokes use case 8.4. Search Courier.

the "Update" option in the Option column dropdown of the line of informatio n of which they want to update.			ponds by loading the
	controls:		with the following
	Control Name	Control Type	Notes
	Update Courier	Heading	None
	Courier Name:	Label	This is the label for the name of the courier
	Name:	Textbox	None
	Courier Type:	Label	This is the label for the type of the courier
	Type:	Dropdown	Populated with different types to select.
	Courier Number:	Label	This is the label for the number of the courier
	Number:	Textbox	None
	Courier Email:	Label	This is the label for the email of the courier
	Email:	Textbox	None

	Done Button	This button is used for when the admin is done updating the information and wants to proceed to update the new information on the system.
	Cancel Button	This button is used when the admin decides to terminate the process of adding a new courier.
Step 4: The admin enters the new	the information ente the Courier table [A	
informatio n and clicks on	Input Attribute Name	in table Validation requirement s
the "Done". [Alt]	Courier Courier_ Name	Name Required
	Type CourierT	ype_Des Required
	Numbe Courier_r	Number Required, 10 digits
	Email Courier_	Email Required, maximum of 200 characters
	Step 6: The system to save the captured Courier table [Alt]:	uses a SQL insert query I information in the
	Informati Attribut on to table be saved:	e in Details

Courier Name	Courier_Nam e	None
Courier Type	CourierType_ ID	Match the CourierType_I D of the selected CourierType_D esc
Courier Number	Courier_Num ber	None
Courier Email	Courier_Emai I	None
Step 7: The	system saves t	he new courier

Step 7: The system saves the new courier details in the relevant entities and displays a success modal:

Input Name	Control Type	Notes
Success	Modal Heading	None
Courier information was successfully updated.	Label	Label to indicate the courier information is saved.
Ok	Button	None

ALTERNATE COURSES:

[Alt Step 4]: The admin clicks on the "Cancel" button which terminates this use case.

Return to step 1.

[Alt Step 5]: The details entered were not in the right format, so the validation failed. The system will display a validation unsuccessful modal to the admin.

Input Name	Control Type	Notes
VALIDATION UNSUCCESSFUL	Label	This serves as a header for the modal.
The information provided in the given parameters is not in the correct format.	Label	This is to show that the courier information was incorrectly entered.

	BACK	Button	This button allows the admin the return to the			
			'Update Courier' screen to edit the entered information in the correct format.			
	After the admin clicks on the " Back " button the system will show the admin where the validation errors occurred by showing a red outline where the validation failed.					
	[Alt Step 6]: The updated count to the system. The system wadmin that the saving of the	ill display	a modal informing the			
	Input Name	Control Type	Notes			
	UNSUCCESSFUL	Label	This serves as a header for the modal.			
	Failed to update courier information. Please try again later.	Label	This is to show that the saving of the courier information was unsuccessful.			
	OK	Button	None			
CONCLUSION:	This use case concludes wh successfully updated on the		urier information has been			
POST- CONDITION:	The courier has been up	dated in th	e Courier table.			
BUSINESS RULES:	Only an admin can update the information of the courier in the system.					
IMPLEMENTATI ON CONSTRAINTS						
AND SPECIFICATIO NS:	None					
ASSUMPTIONS:	None					
OPEN ISSUES:	None					

	NKAP BOLTING SYSTEM					
DATE:	27-08-21					
VERSION:	1.0					
AUTHOR(S)	Sonali Badrinath					
USE CASE NAME:	Collect Order	USE CASE TYPE				
USE CASE ID:	8.6.	Business Requirements:				
PRIORITY:	High	System Analysis: □				
Source:	NKAP Bolting System Requirements	System Design: ☑				
PRIMARY BUSINESS ACTOR:	Customer					
PRIMARY SYSTEM ACTOR:	Employee					
OTHER PARTICIPATING ACTORS:	None					
OTHER INTERESTED STAKEHOLDERS :	Customer Owner					
DESCRIPTION:	This use case describes the event where an employee searches for the specific order that the customer has arrived to collect on the system. Once the customer has collected the order, the employee will indicate so on the system and the system will update the order status of the order appropriately in the Sale Entity . The use case concludes when the order status has been updated to collected.					
PRE-CONDITION:	 The Employee must be logged in to the system. The sale must exist on the system and have an order status of "Ready for Collection". 					
TRIGGER:	The customer wants to collect their order.					
TYPICAL		SYSTEM RESPONSE:				

Course		Manual	Automated Actio	n			
OF EVENTS:		Action					
	Step 1: The customer wants to collect their order. Step 3: The customer provides their order details such as their name and sale number.	Step 2: The employee requests the client for their order details Step 4: The employee will select the Admin Menu item on the navigation bar and click the "Online-Sales" dropdown menu item. Step 5: The system					
		invokes Use Case 5.1 Search Online Sales. Step 6: The	Sten 7: The syste	m updates Order Status to	<u> </u>		
		employee	"Collected by making use of a SQL Update query:				
		clicks the menu icon for the specific	Details to be Updated	Attribute in Entity	Notes		
		order. The employee selects the collected option.	Order Sta tus	OrderStatus_ID in the Sale Entity	None		
ALTERNATE	ALT-STEP Step 6: Based on not finding the order that match the details provided by the customer, the employee may Terminate the use case.						
Conclusion:	The use case concludes when the Order has been collected by the customer and the order status is updated to collected.						
Post- condition:	The Order Status has been updated to "Collected".						

BUSINESS RULES:	 Order status can only be updated to "Collected" if it has an order status of "Ready for Collection". Once an order status has been updated to "Collected" it cannot be updated.
IMPLEMENTATIO N CONSTRAINTS AND SPECIFICATIONS :	None
Assumptions:	None
OPEN ISSUES:	None

NKAP BOLTING SYSTEM					
DATE:	27-08-21				
VERSION:	1.0				
AUTHOR(S)	Sonali Badrinath				
USE CASE NAME:	Deliver Order	USE CASE TYPE			
USE CASE ID:	8.7.	Business Requirements:			
PRIORITY:	High	System Analysis: □			
Source:	NKAP Bolting System Requirements	System Design: ☑			
PRIMARY BUSINESS ACTOR:	Employee				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS :	Customer Owner				
DESCRIPTION:	This use case describes the event where an employee searches for the specific order that has been delivered to a customer. Once the order has been delivered to the customer, the employee will indicate so on the system and the system will update the order status of the order appropriately in the Sale Entity. The use case concludes when the order status has been updated to delivered.				
Pre-condition:	 The Employee must be logged in to the system. The sale must exist on the system and have an order status of "Ready for Delivery". 				
TRIGGER:	An employee wants to record on the system that the order has been delivered to the customer.				
TYPICAL		SYSTEM RESPONSE:			

Course		Manual	Automated Actio	n			
		Action	Automateu Actio	"			
OF EVENTS:		Action					
OF EVENTS:	Step 1: The employee to record that he/she has delivered the order to the customer.	Step 2: The employee will select the Admin Menu item on the navigation bar and click the "Online-Sales" dropdown menu item. Step 3: The system invokes Use Case 5.1 Search Online Sales. Step 4: The employee	-	m updates Order Status to			
		clicks the menu icon for the specific order. The employee selects the delivered option.	Details to be Updated Order Status	Attribute in Entity OrderStatus_ID in the Sale Entity	Notes		
ALTERNATE	None						
Conclusion:	The use case concludes when the Order has been delivered by the employee and the order status is updated to delivered.						
Post- condition:	The Order Status has been updated to "Delivered".						
Business Rules:	 Order status can only be updated to "Delivered" if it has an order status of "Ready for Delivery". Once an order status has been updated to "Delivered" it cannot be updated. 						

IMPLEMENTATIO N CONSTRAINTS AND SPECIFICATIONS :	None
Assumptions:	None
OPEN ISSUES:	None

2.1.9. PRODUCTS SUBSYSTEM

NKAP BOLTING SYSTEM					
DATE:	05/06/2021				
VERSION:	1.0				
AUTHOR(S)	Jerome Amenigy				
USE CASE NAME:	Add Product (Category	USE CASE TY	PE	
USE CASE ID:	9.1		Business Requirements:□		
PRIORITY:	Medium		System Analysis:		
SOURCE:	NKAP Boltin Requiremen		System Design: ☑		
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	This use case describes the process whereby an admin adds a new Product Category to the system. The admin will select an option to add a product category to the system and then enter the specified product category's information and upon saving the addition the system will add it to the ProductCategory Entity				
PRE-CONDITION:	The admin is logged in to the system.				
TRIGGER:	The admin wants to add a new product category to the system.				
TYPICAL COURSE	ACTOR SYSTEM RESPONSE:				
OF EVENTS:	ACTION:	Manual Action	Automated Action		
	Step 1: The admin requests to add a new product category on the system				
	Step 2: The admin clicks on the "Add" button in		Step 3: The system then disp Product category modal with controls:		

the Products Screen.		Input Nan		Control Type		Notes	
ocieen.		Add Prod Category		Heading	l	None	
		Product Category Name:		Label		A label for the name of the Product Category	
		Name	-	Textbox		None	
		Add	I	Button		The button is used to save the product category onto the system	
		Cancel	Ī	Button		The button is used to terminate the process of this use case	•
Step 4: The admin enters the specified product category's details.							
Step 5: The admin clicks on the "Add"		Step 6: The system captures the product category's details and validates it against Product Category Table.					>
button		Input Name	Attrib in Tal			ation rements	
		Name:			_		
		Step 7: The system uses an insert query to save the captured information in the ProductCategory Table					
		Information to be save		attribute n table	e De	tails	

			ProductCat egory_ID	ProductC ategory_I D	Generated from the system by reading the last Product Category_I D and adding 1
			Name	ProductC ategory_ Desc	None
ALTERNATE	Step 5b: The	e user clicks	s on "cancel" wh	nich termina	tes the use case
COURSES:	-		product categor outline the place	-	re not in the right validation error
	•	displays a ı	innot save the r modal telling the	•	category; the the addition was
	Input Name	•	Control Type	Notes	
	Error		Label	This serv	res as a or the modal.
	There was while savin product ca Please try	ng the new tegory.	Label	saving of	show that the the product information accessful.
	ОК		Button	None	
CONCLUSION:	This use case concludes once the product category has been successfully added on to the system				
POST-CONDITION:	The product category has been added to the ProductCategory Table				
BUSINESS RULES:	1. Only the a	dmin can a	dd product cate	egories to the	e system
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None				
ASSUMPTIONS:	None				
OPEN ISSUES:	None				

NKAP BOLTING SYSTEM					
DATE:	05/06/2021				
VERSION:	1.0	1.0			
AUTHOR(S)	Jerome Ame	nigy			
USE CASE NAME:	Search Produc	t Category	USE CASE TYPE		
USE CASE ID:	9.2		Business Requirements:□		
PRIORITY:	Medium		System Analysis: □		
SOURCE:	NKAP Bolting Requirements		System Design: ☑		
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	This use case describes the process where the admin searches for a specific Product Category. Once the admin enters the search criteria the system will retrieve the matching results from the ProductCategory entity the use case ends once the retrieved results have been displayed to the admin.				
PRE-CONDITION:	The admir	n is logged	in to the system		
	• The Produ	uctCategor	y exists on the system		
TRIGGER:	The admin wa	ants to sea	arch for a product category.		
TYPICAL COURSE	ACTOR		SYSTEM RESPONSE:		
OF EVENTS:	ACTION:	Manual Action	Automated Action		
	Step 1: The admin requests to search for a product category				
	Step 2: The admin clicks on the "Products" screen.		Step 3: The system responds by loading the "Products" screen with the following controls: Control Control Notes Name Type		

	Dualizat	11"	M
	Products	Heading	None
	Search Bar	Textbox	Search bar used to search for category types
	Product Category	Label	Name of the selected product screen.
	Product Category Table	Table	This is to show the Product Categories that are saved in the database in a table.
	Product Category Name	Column	This is the column for the Product Category names.
	Product Category ID	Column	This is the column for the Product Category IDs.
	Actions	Column	Column to display update and delete buttons
	Product Category / Category Type/ Product Item	Button Group	This is the button group whereby the admin can select which level of products he wants to work with
	Options	Button	This is to show the update and delete button for each row.
	Add	Button	None
	Update	Button	None
	Delete	Button	None
Step 4: The admin selects the "Product Category"			

	radio				
	button.				
	Step 5: The admin enters the product category search criteria.				
	Step 6: The admin		_	tem captures and nest the Product C	
	clicks on the search icon.		Input Name	Attribute Name	Validation Requirement s
			Name:	ProductCateg ory_Desc	Maximum of 50 characters
			ID:	ProductCateg ory_ID	Characters must be integers
			the matching ententity. The system ProductCatego ProductC	ry table: ategory_ID	
-				ategory_Desc tem displays the	retrieved search
			results in a table		
	Step 10: The admin selects the product category they were searching for.				
	Step 7b: The validation fails, and the admin is prompted to enter a new search query.				
COURSES:	-				
	The use case concludes once the system displays the search results to the admin and the admin selects the searched product category.				
POST-CONDITION:	The admin ha	as recei	ved the results of	the search	
BUSINESS RULES:	1. None				

IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

NKAP BOLTING SYSTEM				
DATE:	05/06/2021			
VERSION:	1.0			
AUTHOR(S)	Jerome Am	enigy		
USE CASE NAME:	Update Prod Category	uct	USE CASE TYPE	
USE CASE ID:	9.3		Business Requirements:□	
PRIORITY:	Medium		System Analysis: □	
SOURCE:	NKAP Boltir System Requiremen		System Design: ☑	
PRIMARY BUSINESS ACTOR:	Admin			
PRIMARY THE SYSTEM ACTOR:	None			
OTHER PARTICIPATING ACTORS:	None	None		
OTHER INTERESTED STAKEHOLDERS:	None	None		
DESCRIPTION:	This use case describes the event whereby an already existing Product Category is updated by an admin. Firstly the admin indicates that they want to update a product category, next the system invokes Use Case 9.2 Search category, after the relevant information is displayed the admin chooses to edit the product category. After editing and saving the changes the system updates the ProductCategory entity with the changes made.			
PRE-CONDITION:	 The ProductCategory already exists on the system The admin is logged in to the system. 			
TRIGGER:	The admin v	wants to	update the Product Category on the system.	
TYPICAL COURSE	ACTOR		SYSTEM RESPONSE:	
OF EVENTS:	ACTION:	Manual Action	Automated Action	

	Step 1: The admin requests to update a Product Category.	Product Cate	gory"	es UC 9.2 "Search		
	Step 3: The admin clicks on the	Step 4 : The system responds by loading the "Update Product Category" modal with the following controls:				
	update button	Control Name	Control Type	Notes		
		Update Product Category	Heading	None		
		Product Category Name	Label	Label for the name of the product category		
		Name	Textbox	The textbox through which the name for the product category will be provided		
		Update	Button	None		
		Cancel	Button	None		
	Step 5: The admin edits and makes all the changes they want to make to the product category.					
Step 6: The admin clicks the	The admin	changes mad	le to the pro	ures and validates the duct category entered by duct category table.	y	
	button		Attribute in Table	Validation Requirements		

			Name	Product _Desc	tCategory	Maximu characte	
			Step 8: The table with th	-	•	Product(Category
			Informatio updated	n to be	Attribute	in table	Detail s
			ProductCa _ID	ategory	Product_I	D	None
			Name		ProductCo Desc	ategory_	None
ALTERNATE	Step 6b: Touse case	he admin	clicks on the	"Cancel"	button whi	ch termin	ates the
COURSES:	•		tion of the ent on that does n				
			es made to th ay a modal st				
	Input Nan	ne	Control Ty	ре	Notes		
	Error		Label		This serv	es as a or the mo	dal.
	There was error while the new publicategory informatical Please try later.	e saving product on.	Label		saving of category	show that the prod informati uccessful	uct on
	OK		Button		None		
CONCLUSION:	This use case concludes when the product category information has been successfully updated on the system.						
POST-CONDITION:	The product category is updated in the ProductCategory table						
BUSINESS RULES:	1. Only an	admin car	update Prod	luct Cate	gories in th	e system	
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None						
ASSUMPTIONS:	None						
OPEN ISSUES:	None						

	NKA	P BOLTII	NG SYSTEM		
DATE:	05/06/2021				
VERSION:	1.0				
AUTHOR(S)	JEROME AM	ENIGY			
USE CASE NAME:	Delete Product Category		U	SE CASI	Е ТҮРЕ
USE CASE ID:	9.4		Business Requ	uirements	s:□
PRIORITY:	Medium		System Analys	is:	
SOURCE:	NKAP Bolting Requirements	-	System Design):	Ø
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	This use case describes the process where the admin deletes a Product Category from the system. Upon indicating that the admin wants to delete a product category the system will invoke use case 9.2, the admin will then proceed to delete the selected product category and the system will remove the selected product category from the ProductCategory entity.				ng that the admin I invoke use case 9.2, product category
PRE-CONDITION:			I in to the system ry already exists o		stem
TRIGGER:			ete a product cate	-	
TYPICAL COURSE			· · · · · · · · · · · · · · · · · · ·	I RESPO	NSE:
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Action	n	
	Step 1: The admin requests to delete a product category Step 2: The system invokes use case Search Product Category				
	Step 3: The admin clicks on the		Step 4: The sys following control	•	ays a modal, with the
	delete button			Control Type	Notes

		Confirm deletion	Heading	This serves as a header for the modal.	
			Are you sure you want to delete this product category?	Label	None
			Yes	Button	This is to confirm that the admin wants to delete a Product Category.
			No	Button	This is selected when the admin does not want to delete a product category.
	Step 5: The admin clicks on the "Yes" button to	s	Step 6: The sy product catego entity.		
	confirm the deletion of the product category.		Information to be updated	Attribute in table	Details
			ProductCat egory_ID	Product_ID	None
			Name	ProductCateg _Desc	gory None
			Step 7: The sy product catego screen.		the deleted le in the products
ALTERNATE	[Alt Step 5]: ¹ terminated.	The admin	clicks on the "N	lo" button and t	the use case is
COURSES:	[Alt Step 6]: The product category could not be deleted from the system. The system will display a modal informing the admin that the deletion of the product category was unsuccessful.				

	Input Name	Control Type	Notes		
	Error	Label	This serves as a header for the modal.		
	There was an error while deleting the product category. Please try again later.	Label	This is to show that the deletion of the product category information was unsuccessful.		
	ОК	Button	None		
CONCLUSION:	The use case concludes once the selected product category has been successfully deleted from the system.				
POST-CONDITION:	The product category has been removed from the ProductCategory entity.				
BUSINESS RULES:	1. Only the admin can delete a product category.				
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None				
ASSUMPTIONS:	None				
OPEN ISSUES:	None				

NKAP BOLTING SYSTEM				
DATE:	05/06/2021			
VERSION:	1.0			
AUTHOR(S)	JEROME AMENIGY			
USE CASE NAME:	Add Category Type	USE CASE TYPE		
USE CASE ID:	9.5	Business Requirements:□		
PRIORITY:	Medium	System Analysis: □		
SOURCE:	NKAP Bolting System Requirements	System Design: ☑		
PRIMARY BUSINESS ACTOR:	Admin			
PRIMARY THE SYSTEM ACTOR:	None			
OTHER PARTICIPATING ACTORS:	None			

OTHER WITERSOTER						
OTHER INTERESTED STAKEHOLDERS:	None	None				
DESCRIPTION:	This use case describes the process whereby an admin adds a new Category Type to the system. The admin will select an option to add a category type to the system and then enter the specified category type's information and upon saving the addition the system will add it to the CategoryType Entity.					
PRE-CONDITION:	The related	ProductCa	ategory already	exists		
			to the system			
TRIGGER:	The admin wan	its to add a	new category	type to the syst	em.	
TYPICAL COURSE	ACTOR		SYSTEM	// RESPONSE:		
OF EVENTS:	ACTION:	Manual Action	Automated Act	ion		
	Step 1: The admin requests to add a new category type on the system Step 2: The admin clicks on the "Add" button in the Products Screen.		•	ystem retrieves rently available Attribute in table	•	tem.
	Coroon.		ProductCa tegory_ID	Product_ID	None	
			Name	ProductCateg _Desc	ory None	
			-	ystem then disp ory modal with	-	
			Input Name	Control Type	Notes	
			Add Category Type	Heading	None	
			Product Categories:	Dropdown	A dropdov containing the existing product categories	g all ng

	Category TypeNam	e: Label		A label for the name of the Categor type	
	Name	Textb	ОХ	None	
	Add	Buttor	า	The button used to save the category type onto the system	re y
	Cancel	Buttor	า	The button is used to terminate the process of this use case	ne
Step 5: The admin selects the Product Category that he wants to add the category type to.					
Step 6: The admin enters the category type's details.					
Step 7: The admin clicks on the add button	-	ils and valid	-	the category against the	,
	Input Name	Attribute i Table		Validation requiremen ts	
	Name:	CategoryT _Desc	I	Maximum character length of 50 characters.	
	-	ptured info		insert query to in the	to
	Information to be save			Details	

			CategoryTy pe _ID	CategoryT ype _ID	Generated from the system by reading the last CategoryTyp e _ID and adding 1
			Name	CategoryT ype _Desc	None
ALTERNATE	Step 7b: The u	ser clicks or	n "cancel" whic	h terminates	the use case
COURSES:	Step 8b: The p format; the syst occurred.				_
	Step 9b: The system cannot save the new category type; the system then displays a modal telling the admin that the addition was unsuccessful.				
	Input Name		Control Type	Notes	
	Error			This serves a	
	There was an	error	Label	This is to sho	w that the
	while saving category type try again late	e. Please		saving of the information ty unsuccessful	/pe was
	category type	e. Please		information ty	/pe was
CONCLUSION:	category type try again late	e. Please r. concludes o	Button	information ty unsuccessful None	/pe was
CONCLUSION: POST-CONDITION:	Category type try again late OK This use case cadded on to the	concludes of esystem	Button	information ty unsuccessful None ry type has be	een successfully
	Category type try again late OK This use case cadded on to the	concludes of e system by type has be	Button nce the categor	information ty unsuccessful None ry type has be the Category	een successfully Type entity.
POST-CONDITION:	Category type try again late OK This use case cadded on to the added on to the category.	concludes of e system by type has be	Button nce the categor	information ty unsuccessful None ry type has be the Category	een successfully Type entity.
POST-CONDITION: BUSINESS RULES: IMPLEMENTATION CONSTRAINTS AND	Category type try again late OK This use case of added on to the added on to the category. 1. Only the adm	concludes of e system by type has be	Button nce the categor	information ty unsuccessful None ry type has be the Category	een successfully Type entity.

NKAP BOLTING SYSTEM				
DATE:	05/06/2021			

VERSION:	1.0				
AUTHOR(S)	JEROME AMEN	NIGY			
USE CASE NAME:	Search Category	Гуре		USE CASI	Е ТҮРЕ
USE CASE ID:	9.6		Business Req	uirements	s:□
PRIORITY:	Medium		System Analys	sis:	
SOURCE:	NKAP Bolting Sy Requirements	ystem	System Desig	n:	
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None	None			
DESCRIPTION:	This use case describes the process where the admin searches for a specific Category Type. Once the admin enters the search criteria the system will retrieve the matching results from the <u>CategoryType</u> entity the use case ends once the retrieved results have been displayed to the admin.				search criteria the ategoryType entity the
PRE-CONDITION:			eady exist on the	•	
TRIGGER:	The admin want	s to searc	ch for a category	type	
TYPICAL COURSE	ACTOR		SYSTE	M RESPO	NSE:
OF EVENTS:	ACTION:	Manual Action	Automated Action	on	
	Step 1: The admin requests to search for a category type				
	Step 2: The admin clicks on the "Products" screen.				onds by loading the e following controls:
	3016611.		Name Products	Type Heading	None

			Search Bar	Textbox	Search bar used to search for product categories
			Category Type	Label	Name of the selected product screen.
			Category Type Table	Table	This is to show the Category Types that are saved in the database in a table.
			Category Type Name	Column	This is the column for the Category type names.
			Category Type ID	Column	This is the column for the Category type IDs.
			Actions Column	Column	None
	Catego Catego Type/ Produc Item	Product	Buton Group	This is the button group whereby the admin can select which level of products he wants to work with	
		Options	Button	This is to show the update and delete button for each row.	
			Add	Button	None
			Update	Button	None
			Delete	Button	None
	Step 4: The admin selects the "Category Type" radio button.				
	Step 5: The admin enters the category type search criteria.				

	Step 6: The admin clicks on the search icon.		Step 7: The system captures and validates the information against the Category Type table .				
	icon.		Input Name	Attribute Name	Validation Requirements		
			Name:	CategoryType _Desc	Maximum of 50 characters		
			ID:	CategoryType_ID	Characters must be integers		
			the matchi	he system uses a reing entries from the essistem retrieves:			
			Category				
				egoryType_ID ductCategory_ID			
				egoryType_Desc			
				ne system displays to	he retrieved		
	Step 10: The admin selects the category type they were searching for.						
ALTERNATE COURSES:	Step 7b: The vasearch query.	alidation fa	ails, and the	admin is prompted	to enter a new		
CONCLUSION:				stem displays the se searched category t			
POST-CONDITION:	The admin re	eceives th	e results of	the search query			
BUSINESS RULES:	1. None						
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None						
ASSUMPTIONS:	None						
OPEN ISSUES:	None						

NKAP BOLTING SYSTEM				
DATE:	05/06/2021			
VERSION:	1.0			

AUTHOR(S)	JEROME AN	MENIGY			
USE CASE NAME:	Update Categ	ory Type	USE CASE TYPE		
USE CASE ID:	9.7		Business Requirements:□		
PRIORITY:	Medium		System Analysis: □		
SOURCE:	NKAP Boltin System Requirement		System Design: ☑		
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	This use case describes the event whereby an already existing Category Type is updated by an admin. Firstly the admin indicates that they want to update a category type, next the system invokes Use Case 9.6 Search category type, after the relevant information is displayed the admin chooses to edit the category type. After editing and saving the changes the system updates the CategoryType entity with the changes made.				
PRE-CONDITION:	Ŭ		already exist on the system gged in to the system		
TRIGGER:	The admin w	ants to up	odate the Category Type on the system.		
TYPICAL COURSE			SYSTEM RESPONSE:		
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Action		
	Step 1: The admin requests to update a Category Type.		Step 2 : The system invokes UC 9.6 "Search Category Type"		
	Step 3: The admin clicks on the update button		Step 4 : The system responds by loading the "Update Category Type" modal with the following controls:		
	Duttori		Control Notes Name Type		

			Update Categor Type	y Head	ding I	None		
		Categor Type Name	y Labe	· -	Label for th the categor			
		Name	Text	\ t	The textbook which the r the categor be provided	name for ry type will		
			Update	Butto	n n	None		
			Cancel	Butto	n l	None		
	Step 5: The admin edits the category type accordingly							
Step 6: The admin clicks the "Save"	The admin clicks the "Save"		changes	made to th	ne categ	es and vali gory type ei y type table	ntered by t	he
	button		Input Name	Attribute Table	e in	Validation Require		
			Name	Category Desc	Type_	Maximur characte		
			Step 8: T table with	•	•	es the Cate de.	goryType	
			Informa be upda		Attrib table	oute in	Details	
			Catego _ID	ryType	Cateo	goryType	None	
			Name		Cateo	goryType c	None	
ALTERNATE COURSES:	Step 6b: The a	admin cli	icks on the	e "Cancel"	button	which term	inates the	
	Step 7b: The voutline the information						•	
	Step 8b: The c system will disp	_		_				he

	Input Name	Control Type	Notes		
	Error	Label	This serves as a header for the modal.		
	There was an error while saving the new category type information. Please try again later.	Label	This is to show that the saving of the category type information was unsuccessful.		
	ОК	Button	None		
CONCLUSION:	This use case conclude successfully updated	• • •	type information has beer	า	
POST-CONDITION:	The category type	is updated in the Cate	goryType table		
BUSINESS RULES:	1. Only an admin can	update Category Type	es in the system		
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None				
ASSUMPTIONS:	None				
OPEN ISSUES:	None				

	NKAP BOLTING SYSTEM				
DATE:	05/06/2021	05/06/2021			
VERSION:	1.0				
AUTHOR(S)	JEROME AMENIGY				
USE CASE NAME:	Delete Category Type	USE CASE TYPE			
USE CASE ID:	9.8	Business Requirements:□			
PRIORITY:	Medium	System Analysis: □			
SOURCE:	NKAP Bolting System Requirements	System Design: ☑			
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				

DESCRIPTION:	This use case describes the process where the admin deletes a Category Type from the system. Upon indicating that the admin wants to delete a category type the system will invoke use case 9.6, the admin will then proceed to delete the selected category type and the system will remove the selected category type from the CategoryType entity.						
PRE-CONDITION:	Ŭ		already exist on	•			
TRIGGER:			ged in to the sys				
TYPICAL COURSE	rne admin wa	ants to dei	ete a category t	ype EM RESPOI	NCE.		
OF EVENTS:	ACTOR ACTION:	Manual Action	Automated Acti		NGE.		
	Step 1: The admin requests to delete a category type		Step 2: The sy Category Type	es UC 9.6 "Search			
	Step 3: The admin clicks on the		Step 4 : The system displays a modal, with the following controls:				
	delete button		Control Name	Control Type	Notes		
				Confirm deletion	Heading	This serves as a header for the modal.	
			Are you sure you want to delete this category type?	Label	None		
			Yes	Button	This is to confirm that the admin wants to delete a Category Type.		
			No	Button	This is selected when the admin does not want to delete a category type.		
	Step 5: The admin clicks on the "Yes"		•		ves the selected egory type entity.		

	button to confirm the		Information to be updated	Attribute in table	Details		
	deletion of the category type.		CategoryType _ID	CategoryType _ID	None		
	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,		ProductCatego ry_ID	Product_ID	None		
			Name	CategoryType _Desc	None		
			Step 7: The system from the table in the				
ALTERNATE	[Alt Step 5]: 1 terminated.	Alt Step 5]: The admin clicks on the "No" button and the use case is erminated.					
COURSES:	[Alt Step 6]: The category type could not be deleted from the system. The system will display a modal informing the admin that the deletion of the category type was unsuccessful.						
	Input Name	(Control Type	Notes			
	Error		Label	This serves as a header for the modal.			
	There was a error while deleting the category type Please try against the later.	oe.	Label	This is to show the deletion of category type information wa unsuccessful.	the		
	ОК		Button	None			
CONCLUSION:	The use case concludes once the selected category type has been successfully deleted from the system.						
POST-CONDITION:	The category	ory type h	as been removed fro	om the Category	Type entity		
BUSINESS RULES:	1. Only the ad	lmin can c	delete a category typ	e.			
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None						
ASSUMPTIONS:	None						
OPEN ISSUES:	None						

NKAP BOLTING SYSTEM				
DATE:	05/06/2021			

VERSION:	1.0					
AUTHOR(S)	JEROME AI	MENIGY				
USE CASE NAME:	Add Product	ltem	USE CASE	TYPE		
USE CASE ID:	9.9		Business Requirement	s:□		
PRIORITY:	Medium		System Analysis:			
SOURCE:	NKAP Boltin Requiremen		System Design:	Ø		
PRIMARY BUSINESS ACTOR:	Admin					
PRIMARY THE SYSTEM ACTOR:	None					
OTHER PARTICIPATING ACTORS:	None					
OTHER INTERESTED STAKEHOLDERS:	None					
DESCRIPTION:	This use case describes the process whereby an admin adds a new Product Item to the system. The admin will select an option to add a product item to the system and then enter the specified product item's information and upon saving the addition the system will add it to the ProductItem Entity.					
PRE-CONDITION:	The adm	in is logged	in to the system.			
	 The relat 	ed Categor	y Type already exists			
TRIGGER:	The admin w	vants to add	d a new product category t	to the system.		
TYPICAL COURSE	ACTOR		SYSTEM RESPON	SE:		
OF EVENTS:	ACTION:	Manual Action	Automated Action			
	Step 1: The admin requests to add a new product item on the system					
	Step 2: The admin selects Product Item radio button					
	Step 3: The admin clicks on		Step 4: The system retri types currently available	0 0		

the "Add" button in	Information retrieved	Attribute in table	Details
the Products Screen.	CategoryTy pe ID	CategoryType _ID	None
	ProductCat egory ID	ProductCateg ory ID	None
	Name	CategoryType _Desc	None
	Step 5: The system of the syst		
	Input Name	Control Type	Notes
	Add Product Item	Heading	None
	Category Type:	Dropdown	A dropdown containing all the existing category types
	Product Item Name:	Label	Message telling the admin to enter the product item name
	Name	Textbox	Field to enter the product item name
	Description	Textbox	Field to enter the product item description
	Quantity	Numeric up down	Numeric up down to select the product item quantity
	Image	File input	File input used to select the product item image

		Upload Button	Button	Button used to upload the image
		Cost	Textbox	Field to enter the product item cost
		Add	Button	The button is used to save the category type onto the system
		Cancel	Button	The button is used to terminate the process of this use case
Step 6: The admin selects the Category Type that he wants to add the product item to.				
Step 7: The admin enters the product item's details.				
Step 8: The admin clicks on the add		-	e system capture Is and validates i n Table.	-
button		Input Name	Attribute in Table	Validation requiremen ts
		Name:	ProductItem_ Name	Maximum character length of 50 characters.
		Descrip tion	ProductItem_ Desc	Maximum character length of

						150 characters.
			Quantit y	Qua Han	intity_On_ d	Entered values must be integers
			Image	Prod mag	ductItem_I je	File Uploaded must be an image
			Cost	Proc Cos	ductItem_ t	Entered values must be decimals
			-	capt	ured informa	an insert query ation in the
			Informati to be sav		Attribute in table	Details
			ProductIt m _ID		ProductIte m_ID	Generated from the system by reading the last ProductItem _ID and adding 1
			Name		ProductIte m _Name	None
			Descripti		ProductIte m_Desc	None
			Image		ProductIte m_Image	None
			Cost		ProductIte m_Cost	None
			Quantity		Quantity_ on_Hand	None
ALTERNATE	Step 8b: The	e user clicks	on "cancel	" whic	ch terminate	s the use case
COURSES:	-					ot in the right didation error/s
	Step 10b: T then displays unsuccessfu	s a modal te			•	item; the system ion was

	Input Name	Control Type	Notes				
	Error	Label	This serves as a header for the modal.				
	There was an error while saving the product item. Please try again later.	Label	This is to show that the saving of the product item information type was unsuccessful.				
	ОК	Button	None				
CONCLUSION:	This use case concludes of successfully added on to the	•	ct item has been				
POST-CONDITION:	The product item has be	The product item has been added to the ProductItem entity.					
BUSINESS RULES:	1. Only the admin can add	product items	to the system				
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None						
ASSUMPTIONS:	None						
OPEN ISSUES:	None						

NKAP BOLTING SYSTEM					
DATE:	05/06/2021				
VERSION:	1.0				
AUTHOR(S)	JEROME AMENIGY				
USE CASE NAME:	Search Product Item	USE CASE TYPE			
USE CASE ID:	9.10	Business Requirements:□			
PRIORITY:	Medium	System Analysis: □			
SOURCE:	NKAP Bolting System Requirements	System Design: ☑			
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				

OTHER INTERESTED STAKEHOLDERS:	None	None					
DESCRIPTION:	specific Proc system will r	This use case describes the event where the admin searches for a specific Product Item. Once the admin enters the search criteria the system will retrieve the matching results from the <u>ProductItem</u> entity the use case ends once the retrieved results have been displayed to the admin.					
PRE-CONDITION:	The admi	n is logged	in to the system				
	The Prod	ductItem e	xists on the system	1			
TRIGGER:	The admin w	vants to se	earch for a Product	Item			
TYPICAL COURSE	ACTOR		SYSTEM	RESPONSE:	:		
OF EVENTS:	ACTION:	Manual Action	Automated Action				
	Step 1: The admin requests to search for a product item Step 2:		Ston 2: The syste	m rosponds h	ny loading tho		
	The admin clicks on the		Step 3: The syste "Products" screen				
	"Products" screen.		Control Name	Control Type	Notes		
			Products	Heading	None		
			Search Bar	Textbox	Search bar used to search for product categories		
			Product Item	Label	Name of the selected product screen.		
		Product Item Table	Table	This is to show the Product Items that are saved in the database in a table.			
			Product Item Name	Column	This is the column for the Product Item names.		

			l <u>.</u> .	
		Product Item ID	Column	This is the column for the Product Item IDs.
		Product Item Description	Column	This is the column for the Product Item Descriptions
		Product Item Image	Column	This is the column for the Product Item images
		Product Item Cost	Column	This is the column for the Product Item cost prices
		Product Item Quantity	Column	This is the column for the Product Item quantities available
		Actions Column	Column	None
		Product Category/ Category Type/ Product Item	Button group	This is the button group whereby the admin can select which level of products he wants to work with
		Options	Icon button	This is to show the update and delete button for each row.
		Add	Button	None
		Write-off	Button	None
		Stock Take	Button	None
		Update	Button	None
		Delete	Button	None
Step 4 The a select	dmin			

"Product Item" radio button. Step 5: The admin enters the product item search				
Step 6: The admin clicks on		ystem captures an ainst the Product		
the search icon.	Input Name	Attribute Name	Validation Requirements	
	Name:	ProductItem _Name	Maximum of 50 characters	
	ID:	ProductItem_ID	Characters must be integers	
	Step 8: The system uses a read query to read the matching entries from the Product Item entity. The system retrieves: ProductItem: ProductItem_ID CategoryType_ID ProductItem_Name ProductItem_Desc ProductItem_Image ProductItem_Cost Quantity_on_Hand CategoryType_Desc Step 9: The system displays the retrieved sea			
Step 10: The admin selects the product item they were searching for.	results in the t	aule.		

ALTERNATE	Step 7b: The validation fails, and the admin is prompted to enter a new search query.
CONCLUSION:	The use case concludes once the system displays the search results to the admin and the admin selects the searched product item
POST-CONDITION:	The admin receives the results of the search query
BUSINESS RULES:	1. None
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

NKAP BOLTING SYSTEM					
DATE:	05/06/2021				
VERSION:	1.0				
AUTHOR(S)	JEROME AMENIGY	,			
USE CASE NAME:	Update Product Item	USE (CASE TYPE		
USE CASE ID:	9.11	Business Requirement	s:□		
PRIORITY:	Medium	System Analysis:			
SOURCE:	NKAP Bolting System Requirements	System Design:	☑		
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	This use case describes the event whereby an already existing Product Item is updated by an admin. Firstly the admin indicates that they want to update a Product Item, next the system invokes Use Case 9.10 Search Product Item, after the relevant information is displayed the admin chooses to edit the Product Item. After editing and saving the changes the system updates the ProductItem entity with the changes made.				
PRE-CONDITION:	The ProductItem a	lready exists on the system			

	The admin is logged in to the system																		
TRIGGER:	The admin	The admin wants to update the Product Item on the system.																	
TYPICAL COURSE	ACTOR	SYSTEM RESPONSE:																	
OF EVENTS:	ACTION:	Manual Action	Automated Action																
	Step 1: The admin requests to update a Product Item.		Step 2: The sy Item"	es UC 9.10 "Search Product															
	Step 3: The admin clicks on			•	nds by loading the "Upo the following controls:	date													
	the update		Control Name	Control Type	Notes														
	button		•	•					Update Product Item	Heading	None								
				Product Item Name	Label	Label for the name of the product item													
					Name	Textbox	The textbox through which the name for the product item will be provided												
			Description	Textbox	Textbox for which the changes to the description of a product item will be provided														
																Quantity	Textbox	Textbox for which the changes to the quantity of a product item will be provided	
			Cost	Textbox	Textbox for which the changes to the cost of a product item will be provided														
		Image	File upload	Control used to upload the product item images															

Upload Button Button used to upload the image								
Cancel Button None				Button			ge	
Step 5: The admin edits the product item Step 6: The admin clicks the "Update" button Step 7: The system captures and validates the changes made to the product item entered by the admin against the ProductItem table. Input Name Attribute in Table Validation Requirements Name ProductItem_Name Maximum of 50 characters Description ProductItem_Desc Maximum of 150 characters Image ProductItem_Image File Uploaded must be an image Cost ProductItem_Cost Entered values must be integers Quantity Quantity_on_Hand Entered values must be decimals Step 8: The system updates the CategoryType table with the changes made. Information to Attribute in table Details be updated ProductItem_ID ProductItem_ID None Name ProductItem_Desc None Image ProductItem_Desc None Image ProductItem_Desc None Image ProductItem_Image None Cost ProductItem_Cost None			Update	Button	None			
The admin edits the product item Step 6: The admin clicks the "Update" button Step 7: The system captures and validates the changes made to the product item entered by the admin against the ProductItem table. Input Name Attribute in Table Validation Requirements Name ProductItem_Name Maximum of 50 characters Description ProductItem_Desc Maximum of 150 characters Image ProductItem_Image File Uploaded must be an image Cost ProductItem_Cost Entered values must be integers Quantity Quantity_on_Hand Entered values must be decimals Step 8: The system updates the CategoryType table with the changes made. Information to Attribute in table Details be updated ProductItem_ID ProductItem_ID None Name ProductItem_Desc None Image ProductItem_Desc None Image ProductItem_Image None Cost ProductItem_Image None			Cancel	Button	None			
changes made to the product item entered by the admin clicks the "Update" button Input Name		The admin edits the product						
Name ProductItem_Name Attribute in Table ProductItem_Name Assimum of 50 characters Description ProductItem_Desc Maximum of 150 characters Image ProductItem_Image File Uploaded must be an image Cost ProductItem_Cost Entered values must be integers Quantity Quantity_on_Hand Entered values must be decimals Step 8: The system updates the CategoryType table with the changes made. Information to be updated ProductItem_ID None Name ProductItem_ID None ProductItem_Name None Description ProductItem_Desc None Image ProductItem_Image None Cost ProductItem_Cost None		The admin clicks the	changes made	to the pr	oduct item e	ntered		:
Description ProductItem_Desc Maximum of 150 characters Image ProductItem_Image File Uploaded must be an image Cost ProductItem_Cost Entered values must be integers Quantity Quantity_on_Hand Entered values must be decimals Step 8: The system updates the CategoryType table with the changes made. Information to Attribute in table Details be updated ProductItem_ID ProductItem_ID None Name ProductItem_Name None Description ProductItem_Desc None Image ProductItem_Image None Cost ProductItem_Cost None			Input Name	Attribut	e in Table			nts
Image ProductItem_Image File Uploaded must be an image Cost ProductItem_Cost Entered values must be integers Quantity Quantity_on_Hand Entered values must be decimals Step 8: The system updates the CategoryType table with the changes made. Information to be updated ProductItem_ID ProductItem_ID None Name ProductItem_Name None Description ProductItem_Desc None Image ProductItem_Image None Cost ProductItem_Cost None			Name	Product	Item_Name			
Cost ProductItem_Cost Entered values must be integers Quantity Quantity_on_Hand Entered values must be decimals Step 8: The system updates the CategoryType table with the changes made. Information to Attribute in table Details be updated ProductItem_ID ProductItem_ID None Name ProductItem_Name None Description ProductItem_Desc None Image ProductItem_Image None Cost ProductItem_Cost None			Description	Product	Item_Desc			
Values must be integers			Image	Product	Item_Image	must	be an	ed
Values must be decimals			Cost	Product	oductItem_Cost		values must	
Information to be updated ProductItem_ID ProductItem_ID None Name ProductItem_Name None Description ProductItem_Desc None Image ProductItem_Image None Cost ProductItem_Cost None			Quantity	Quantity	_on_Hand	value	es must	
be updated ProductItem_ID ProductItem_ID None Name ProductItem_Name None Description ProductItem_Desc None Image ProductItem_Image None Cost ProductItem_Cost None				•		egory	Type ta	able
Name ProductItem_Name None Description ProductItem_Desc None Image ProductItem_Image None Cost ProductItem_Cost None				to At	tribute in ta	ble	Detai	ls
Description ProductItem_Desc None Image ProductItem_Image None Cost ProductItem_Cost None			ProductItem	_ ID Pr	oductItem _I	D	None	
Image ProductItem_Image None Cost ProductItem_Cost None			Name	Pr	oductItem_N	lame	None	
Cost ProductItem_Cost None			Description	Pr	oductItem_D	esc	None	
			Image	Pr	oductItem_Ir	nage	None	
Quantity Quantity_on_Hand None								
			Quantity	Qı	uantity_on_H	land	None	

ALTERNATE	Step 6b: The admin clicks on the "Cancel" button which terminates the use case					
COURSES:	Step 7b: The validation of the entered information fails; the system will outline the information that does not meet the validation requirements.					
	Step 8b: The changes made to the product item could not be saved. The system will display a modal stating the update was unsuccessful					
	Input Name	Control Type	Notes			
	Error	Label	This serves as a header for the modal.			
	There was an error while saving the new product item information. Please try again later.	Label	This is to show that the saving of the product item information was unsuccessful.			
	ОК	Button	None			
CONCLUSION:	This use case concludes when the category type information has been successfully updated on the system.					
POST-CONDITION:	The product item is upd	ated in the Produ	ıctltem table			
BUSINESS RULES:	Only an admin can update Product Items in the system					
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None					
ASSUMPTIONS:	None					
OPEN ISSUES:	None					

	NKAP BOLTING SYSTEM				
DATE:	05/06/2021				
VERSION:	1.0				
AUTHOR(S)	JEROME AMENIGY	JEROME AMENIGY			
USE CASE NAME:	Delete Product Item	USE	CASE TYPE		
USE CASE ID:	9.12	Business Requiremen	nts:□		
PRIORITY:	Medium	System Analysis:			
SOURCE:	NKAP Bolting System Requirements	System Design:			
PRIMARY BUSINESS ACTOR:	Admin				

PRIMARY THE						1
SYSTEM ACTOR:	None					
OTHER PARTICIPATING ACTORS:	None	None				
OTHER INTERESTED STAKEHOLDERS:	None					
DESCRIPTION:	Item from t product ite proceed to	the system m the syst delete the	n. Upon indicatir tem will invoke u	ng that the ause case 9.1 ause case 9.1 act item and	admin deletes a Pradmin wants to dele 10, the admin will that the system will renty.	ete a nen
PRE-CONDITION:		min is logo oduct item	ged in already exists			
TRIGGER:	The admin	wants to	delete a produc	t		
TYPICAL COURSE	ACTOR		1	STEM RES	PONSE:	
OF EVENTS:	ACTION:	Manual Action	Automated Action			
				system invokes use case 9.10 "Search n"		
	Step 3: The admin clicks on		Step 4: The sy following contr	•	ays a modal, with th	е
	the delete		Control Name	Control Type	Notes	
	button		Confirm deletion	Heading	This serves as a header for the modal.	
			Are you sure you want to delete this product item	Label	None	
			Yes	Button	This is to confirm that the admin wants to	

				delete a product item.			
		No	Butto	on This is selected when the admin does not want to delete a product item.			
	Step 5: The admin clicks on	•	Step 6: The system removes the selected categor type from the category type entity.				
	the " Yes "	Information be removed		Attribute in table	Details		
	confirm	Productiter	n_ID	ProductItem_ID	None		
	the deletion of the	CategoryTy _ID	уре	CategoryType_ID	None		
	product	Name		ProductItem _Name	None		
	item.	Description	1	ProductItem_Desc	None		
		Image		ProductItem_Image	None		
		Cost		ProductItem_Cost	None		
		Quantity		Quantity_on_Hand	None		
		Supplier O	rder	SupplierOrderLine_ID	none		
		Step 7: The table in the p	-	removes the product ite screen.	m from the		
	[Alt Step 5]: The admin clicks on the "No" button and the use case is terminated.				e is		
COURSES:	[Alt Step 6]: product item could not be deleted from the system. The syswill display a modal informing the admin that the deletion of the product is was unsuccessful.						
	Input Name	Control Ty	pe	Notes			
	Error	Label		This serves as a header for the modal.			
	There was an error while deleting the product item. Please try againster.	Label		This is to show that the deletion of the product item information was unsuccessful.			

Button

None

OK

CONCLUSION:	The use case concludes once the selected product item has been successfully deleted from the system.
POST-CONDITION:	The category type has been removed from the ProductItem entity.
BUSINESS RULES:	1. Only the admin can delete a product item.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

2.1.10. SPECIALS SUBSYSTEM

NKAP BOLTING SYSTEM						
DATE:	04-06-2021					
VERSION:	1.0					
AUTHOR(S)	S'nethemba Xulu					
USE CASE NAME:	Add Special			USE CASI	Е ТҮРЕ	
USE CASE ID:	10.1		Business Requi	rements:		
PRIORITY:	High		System Analysi	s:		
SOURCE:	NKAP Bolting System Requirements	n	System Design:		Ø	
PRIMARY BUSINESS ACTOR:	None					
PRIMARY THE SYSTEM ACTOR:	None					
OTHER PARTICIPATING ACTORS:	None					
OTHER INTERESTED STAKEHOLDERS:	None					
DESCRIPTION:	This use case describe system. The use case enter all the special do use case ends when the	begins whe etails and t	en the admin reque he information wi	ests to add the II be captured	e special. The adm	in will
PRE-CONDITION:	The admin shoult	ld be on th	ne Specials scree	n		
TRIGGER:	The admin requesting	to add a n	ew special.			
TYPICAL COURSE			SYS	TEM RESPO	NSE:	
OF EVENTS:	ACTOR ACTION:	Manual	Automated Acti	ion		
	Step 1: The admin requests to add a new special.	Action				
	Step 2: The admin will click on the "Add Special" button in the		Step 3: The sys Special" modal		ds by loading the owing controls:	"Add
	Special Screen.		Control Name	Control Type	Notes	
			Add Special	Heading	None	

	Special Image	Label	None
	Special Image	Image	None
	Special Description:	Label	This is the label for the description of the special
	Description	Textbox	None
	Special Price:	Label	This is the label for the price of the special
	Price	Textbox	None
	Start Date:	Label	This is the label for the start date of the special
	Date	Calendar	None
	End Date:	Label	This is the label for the end date of the special
	Date	Calendar	None
	Add	Button	None
	Cancel	Button	This button is used when the admin decides to terminate the process of adding a new special.
Step 4: The admin enters the details in the respective places: • Description • Special price • Start Date • End Date • Image			

Step 5: The admin clicks on the "Add" button [Alt]	ir	Step 6: The synformation entable [Alt]:				
		Input Name	Attribut	e in table		idation uirements
		Image	Special_	_lmage	Red	quired
		Price	Special _.	_Price	max	quired, ximum of characters
		Description	Special	_Description	max 200	quired, ximum of racters.
		Start Date	Special_	_StartDate	Red	quired
		End Date	Special	_EndDate	Rec	quired
		Step 7: The sy he captured in				
		Information t be saved:	O	Attribute in table		Details
		Special_ID		Special_ID	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Generated by the system by reading the last Special_ID from the Special table and adding it by one.
		Special_ Price		Special_Price	e I	None
		Special_ Image		Special_Imag	ge I	None
		Special_ Description		Special_ Description		None
		Special_ StartDate		Special_ StartDate		None
		Special_ EndDate		Special_ EndDate		None

ALTERNATE COURSES:	[Alt Step 5]: The admin clicks on the "Cancel" button which terminates the this use case. Return to step 4 [Alt Step 6]: The details entered were not in the right format, so the validation failed. The system will display to the admin where the validation errors occurred by showing a red outline where the validation failed. [Alt Step 7]: The new special could not be added to the system. The system will display a modal informing the admin that the saving of the special information was unsuccessful.				
	Input Name	Cont	rol Type	Notes	
	Error	Label		This serves as a header for the modal.	
	There was an error while saving the new special. Please try again later.	Label		This is to show that the saving of the special information was unsuccessful.	
	ОК	Butto	n	None	
CONCLUSION:	This use case conclud system.	des when	the special has b	een successfully added	on the
POST-CONDITION:	The special has beer	n added to	the Special table.		
BUSINESS RULES:	Only an admin can add a	a new spe	cial to the system		
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None				
ASSUMPTIONS:	None				
OPEN ISSUES:	None				

	NKAP BOLTIN	IG SYSTEM			
DATE:	04-06-2021				
VERSION:	1.0				
AUTHOR(S)	S'nethemba Xulu				
USE CASE NAME:	Search Special			USE CASE TY	PE
USE CASE ID:	10.2		Business Rec	quirements:	
PRIORITY:	High		System Anal	ysis:	
SOURCE:	NKAP Bolting System Requi	rements	System Design	gn:	$\overline{\mathbf{V}}$
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None				
DESCRIPTION:	This use case describes the event where the admin wishes to search for a special. The admin will enter the search parameters like the suppler name. The system will search for the special based on the parameters the admin entered. This use case concludes when the special is shown the results of the search query.				
PRE-CONDITION:	 The admin should be logged on to the system. The special should be already exist in the system. 				
TRIGGER:	The admin wishes to search fo		the system.		
TYPICAL COURSE	The damin wishes to search to	a special.	SYSTEM	RESPONSE:	
OF EVENTS:	ACTOR ACTION:	Manual	Automated A		
	Step 1: The admin would like to search for a special.	Action			
	Step 2 : The admin clicks on the " Special " screen.	Step 3: The system responds by loading the "Special" screen with the following controls:			
			Control Name	Control Type	Notes
			Special	Heading	None
			Search:	Label	This is the label to prompt the

			user to enter a search query.
	Search	Textbox	None
	Search icon	lcon button	None
	Special Table	Table	This is to show the Specials that are saved in the database in a table.
	Special Image	Column	This is the column for the Special image
	Special Price	Column	This is the column for the special price
	Special description	Column	This is the column for the description of the special
	Options	Icon	This is to show the update and delete button for each row.
	Update	Button	None
	Delete	Button	None
Step 4: The admin inputs the search criteria and clicks on the search icon.	Step 5: The sy validates the ir Special table	nformation	
	Step 6: A SQL read the follow		

		Special table based on the search criteria:
		Special table: Special_ID(PK) Special_Image Special_Price Special_Description Special_EndDate Special_StartDate Step 7: The system will show the search
		results received from step 6 in a table. [Alt]
ALTERNATE COURSE:	[Alt Step 5]: The validation failed. The acquery.	lmin is prompted to enter a new search
CONCLUSION:	The use case concludes when the admin	receives the results of the searched criteria.
POST-CONDITION:	The admin receives the results of the	e specials with the same searched criteria
BUSINESS RULES:	1. None	
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None	
ASSUMPTIONS:	None	
OPEN ISSUES:	None	

	ſ	NKAP BOL	TING SYSTEM			
DATE:	04-06-2021					
VERSION:	1.0					
AUTHOR(S)	S'nethemba Xulu					
USE CASE NAME:	Update Special			USE CAS	SE TYPE	
USE CASE ID:	10		Business Requ	irements:		
PRIORITY:	3		System Analys	is:		
SOURCE:	NKAP Bolting Syste Requirements	em	System Design	:		
PRIMARY BUSINESS ACTOR:	Admin					
PRIMARY THE SYSTEM ACTOR:	None					
OTHER PARTICIPATING ACTORS:	None					
OTHER INTERESTED STAKEHOLDERS:	None					
DESCRIPTION:	This use case describes special. The system with admin will add the been updated successibles.	will display ne new info	the previous info	rmation that v	was added before a	and then
PRE-CONDITION:	The admin shoul	d be logge	d on to the systen	n.		
TRIGGER:	The admin wishes up	odate the d	etails of the speci	al.		
TYPICAL COURSE				Actor Action	:	
OF EVENTS:	Actor Action:	Manual Action	Automated Act	tion		
	Step 1: The admin requests to update the details of a special.		Step 2: The system invokes use case 10.2 Search Special.			
	Step 3: The admin clicks on the "Update" button.		Step 4: The system responds by loading the "Update Special" modal with the following controls:			"Update
			Control Name	Control Type	Notes	
			Update Special	Heading	None	

Special Image	Label	None
Special Image	Image	None
Special Description:	Label	This is the label for the description of the special
Description	Textbox	None
Special Price:	Label	This is the label for the price of the special
Price	Textbox	None
Start Date:	Label	This is the label for the start date of the special
Date	Calendar	None
End Date:	Label	This is the label for the end date of the special
Date	Calendar	None
Update	Button	None
Cancel	Button	This button is used when the admin decides to terminate the process of adding a new special.

Step 5: The admin enters the new information and clicks on the "Update" button. [Alt] **Step 6**: The system captures and validates the information entered by the admin against the Special table [Alt]:

Input Name	Attribute in table	Validation requirements
Image	Special_Image	Required
Price	Special_Price	Required, maximum of 10 characters

Des	scription	Special_Description	Required, maximum of 200 characters.
Sta	rt Date	Special_StartDate	Required
End	d Date	Special_EndDate	Required

Step 7: The system uses a SQL insert query to save the captured information in the **Special** table [Alt]:

Information to be updated:	Attribute in table	Details
Special_ Price	Special_Price	None
Special_ Image	Special_Image	None
Special_ Description	Special_ Description	None
Special_ StartDate	Special_ StartDate	None
Special_ EndDate	Special_ EndDate	None

ALTERNATE COURSES:

[Alt Step 5]: The admin clicks on the "Cancel" button which terminates the this use case.

Return to step 1.

[Alt Step 6]: The details entered were not in the right format, so the validation failed. The system will display to the admin where the validation errors occurred by showing a red outline where the validation failed.

[Alt Step 7]: The updated special information could not be added to the system. The system will display a modal informing the admin that the saving of the special information was unsuccessful.

Input Name	Control Type	Notes
Error	Label	This serves as a header for the modal.
There was an error while saving the new special information. Please try again later.	Label	This is to show that the saving of the special information was unsuccessful.

	ОК	Button	None	
CONCLUSION:	This use case concludes when the system.	hen the special information	n has been successfully upo	dated on
POST-CONDITION:	The special has been updated in the Special table.			
BUSINESS RULES:	1. Only an admin can update the information of the special in the system.			
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None			
ASSUMPTIONS:	None			
OPEN ISSUES:	None			

	NKAP BOLTING SYSTEM				
DATE:	04-06-2021				
VERSION:	1.0				
AUTHOR(S)	S'nethemba Xulu				
USE CASE NAME:	Delete Special	USE	CASE TYPE		
USE CASE ID:	10.4	10.4 Business Requirements:□			
PRIORITY:	High	System Analysis:			
SOURCE:	NKAP Bolting System Requirements	System Design:	abla		
PRIMARY BUSINESS ACTOR:	Admin				
PRIMARY THE SYSTEM ACTOR:	None				
OTHER PARTICIPATING ACTORS:	None				

	1							
OTHER INTERESTED STAKEHOLDERS:	None							
DESCRIPTION:	from the sys	stem. This s s use case	use case begin	ns when the a	dmin wants to del Administrator requal n gets notified that	uests to delete a		
PRE-CONDITION:	The adn	nin should	be logged in to	the system.				
			l already exist o	-				
TRIGGER:	The admin i	requesting	to delete a spe	cial.				
TYPICAL COURSE	ACTOR		5	SYSTEM R	ESPONSE:			
OF EVENTS:	ACTION:	Manual Action	Automated A	ction				
	Step 1: The admin request to delete a special from the system.		Step 2: The system invokes Use case 10.		es Use case 10.	2 Search Special.		
	Step 3: The admin clicks on		Step 4: The scontrols:	system displa	ays a modal, with	the following		
	the "Delete"		Control	Control	Notes			
	button on the Special screen.	the Special	the Special		Name Confirm deletion	Type Heading	This serves as a header for the modal.	
			Are you sure you want to delete this special?	Label	None			
			Yes	Button	This is to confirm that the admin wants to delete a special.			
			No	Button	This is selected when the admin does not want to delete a special.			

	Step 5: The admin clicks on the "Yes"	Step 6: The system special information for				
	button to confirm the	Information to be deleted:	Attribute in table	Details		
	deletion of the special.	Special_ Price	Special_Price	None		
	[Alt]	Special_ Image	Special_Image	None		
		Special_ Description	Special_ Description	None		
		Special_ StartDate	Special_ StartDate	None		
		Special_ EndDate	Special_ EndDate	None		
		Step 7: The system table in the Special s		d special from the		
ALTERNATE		[Alt Step 5]: The admin clicks on the "No" button. Terminate this use case.				
COURSES:		cial could not be delete ning the admin that the	_	•		
	Input Name	Control Type	Notes			
	Error	Label	This serves as header for the			
	There was an error while deleting the special. Please try again later.	Label	This is to show the saving of t special informations was unsucces	he ation		
	ОК	Button	None			
CONCLUSION:	This use case ends w system.	hen the special has be	en successfully dele	eted from the		
POST-CONDITION:	 The details of the 	special have been rem	noved from the Spec	cial table.		

BUSINESS RULES:	1. Only the admin can remove a special.
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None
ASSUMPTIONS:	None
OPEN ISSUES:	None

	NKAP BOLTING SYSTEM					
DATE:	04-06-2021					
VERSION:	1.0					
AUTHOR(S)	S'nethemba Xulu					
USE CASE NAME:	Send Promotional Emails	USE CASE TYPE				
USE CASE ID:	10.5	Business Requirements:□				
PRIORITY:	High	System Analysis: □				
SOURCE:	NKAP Bolting System Requirements	System Design: ☑				
PRIMARY BUSINESS ACTOR:	Admin					
PRIMARY THE SYSTEM ACTOR:	None					

OTHER PARTICIPATING ACTORS:	None				
OTHER INTERESTED STAKEHOLDERS:	None	None			
DESCRIPTION:	This use case describes the e Promotional emails. The use case starts with the admin re then prompts the admin to s from the existing specials. The emails.	case start equesting t elect the s	s with the admi to post an anno specials that the	n requestir uncement, ey want to	ng to The use the system promote
PRE-CONDITION:	The admin is already on the	Supplier so	creen		
TRIGGER:	The admin wants to send pro	motional	email.		
TYPICAL COURSE			Response:		
OF EVENTS:	Actor Action:	Manual	Automated Ac	tion	
	Cton 1. The admin requests	Action			
	Step 1 : The admin requests to send promotional email.				
	Step 2: The admin clicks on the "Send Promotions" button.		Step 3: The sys Promotions" s following cont	creen with	
			Control Name	Control Type	Notes
			Special	Heading	None
			Send Promotion	Label	None
			Special Table	Table	This is to show the Specials that are saved in the database in a table.
			Select	Colum	This is to tick whether you want to include the special in the email.
			Special Image	Column	This is the column for the

					Special image	
			Special Price	Column	This is the column for the special price	
			Special description	Column	This is the column for the description of the special	
			Send Email	Button	None	
			Cancel	Button	None	
	Step 4: The admin selects the specials that they want in their email.		 Step 5: The system of the specials in the Special_ID Special_Print Special_Determine Special_State Special_En 	special to ice escription artDate dDate	able.	
	Step 6: The admin clicks on		Step 7 : The system compiles the email and sends it to the customers			
	the send email. [Alt]		Step 8: The system of the CUST of Custon	stem gets t	the details	
ALTERNATE	[Alt-Step 6]: The admin cancels. The use case terminates.					
COURSES:						
CONCLUSION:	The promotional emails have been sent to the customers.					
POST-CONDITION:	The emails are sent to the customers.					
BUSINESS RULES:	None					
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	None					
ASSUMPTIONS:	None					
OPEN ISSUES:	None					

CONCLUSION

This section concludes. It contained the use cases in the form of a full set of use case diagrams as well as Design Use Case Documentation of each Use Case.

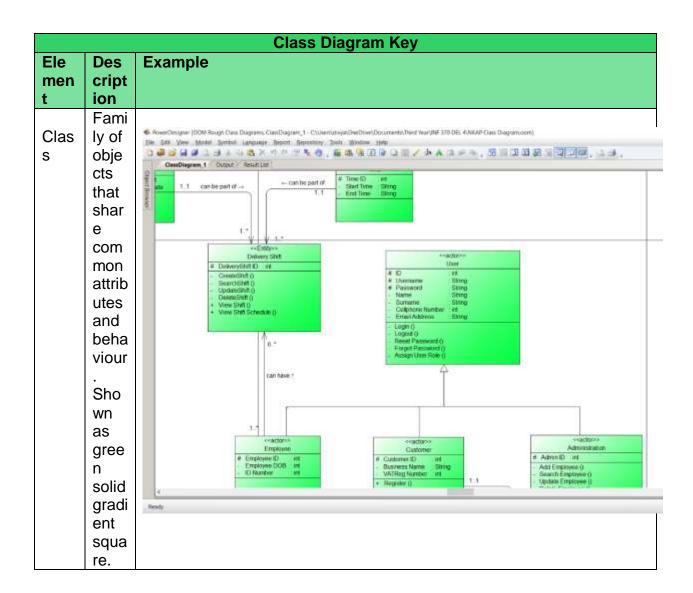
2. Process/Object-Oriented Design

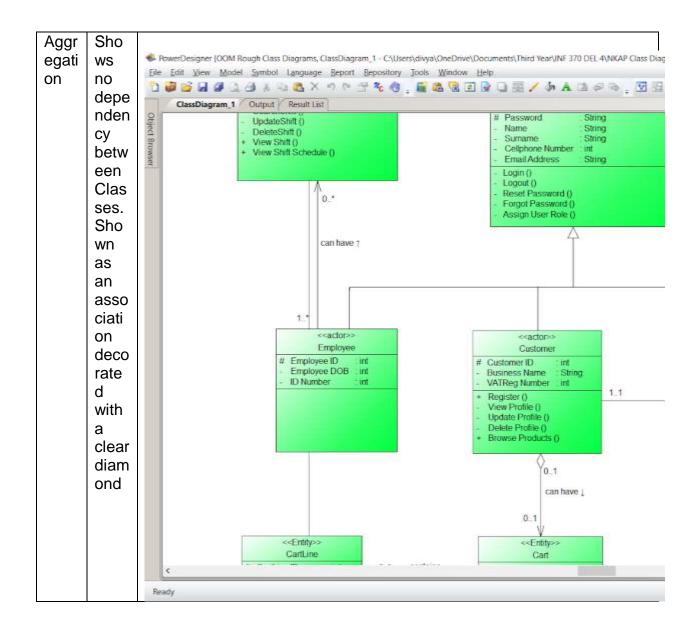
INTRODUCTION

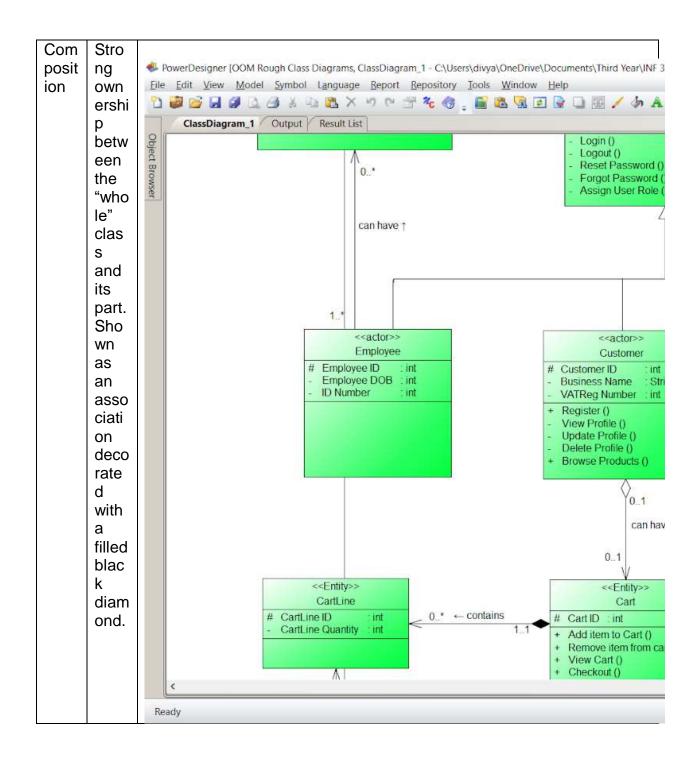
This section contains the Object-Oriented Design for the NKAP Bolting System. This includes a class diagram for the entire system, state diagrams for each object in the class diagram as well as sequence diagrams for each Use Case and Pseudo Code.

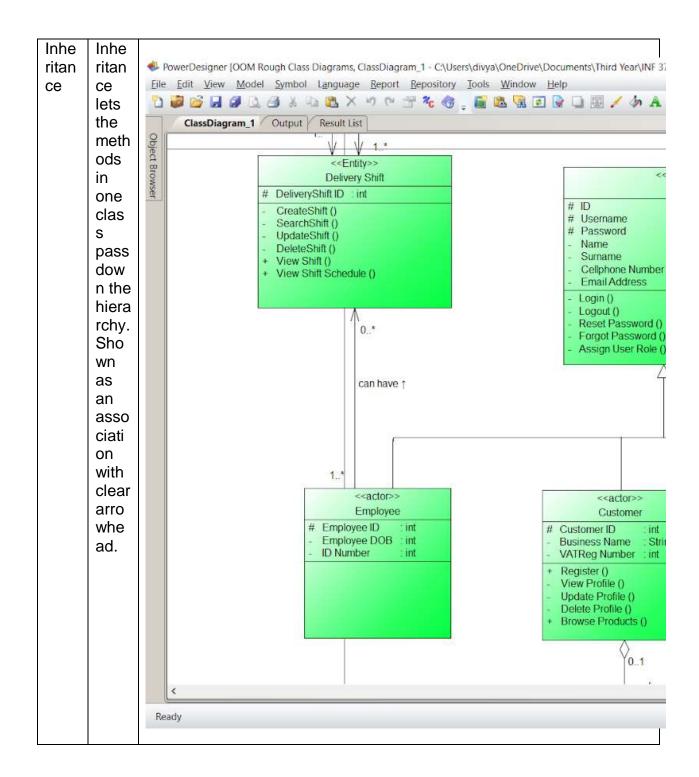
	Use Case Diagram Key	
Element	Description	Example
System	Outer Rectangle Element with the system name, 0. NKAP Bolting system and black outline.	
Sub-System	Inner rectangle element with black outline and Sub-System name.	
Use Case	Use Case element with black outline and green fill.	3.1 Register New Customer
Actor	Use Case actor element with green fill and name of actor and actor type below.	< <pba>> Customer</pba>

Association	Use Case relationship line with 'initiates' tag.	
Uses Association	Use case dependency relationship with 'uses' tag.	

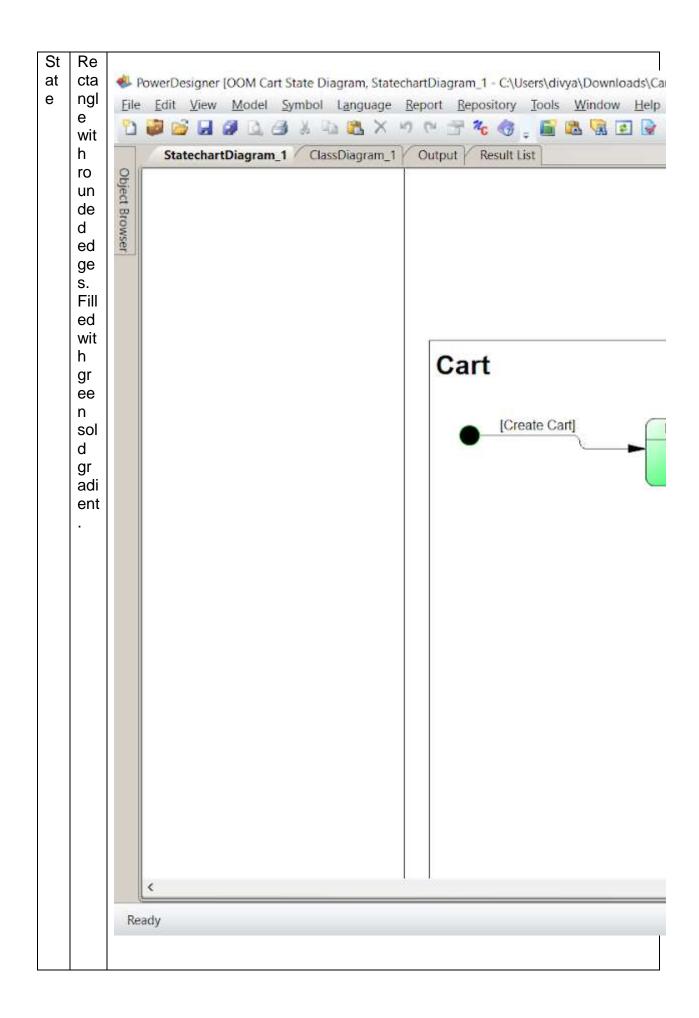




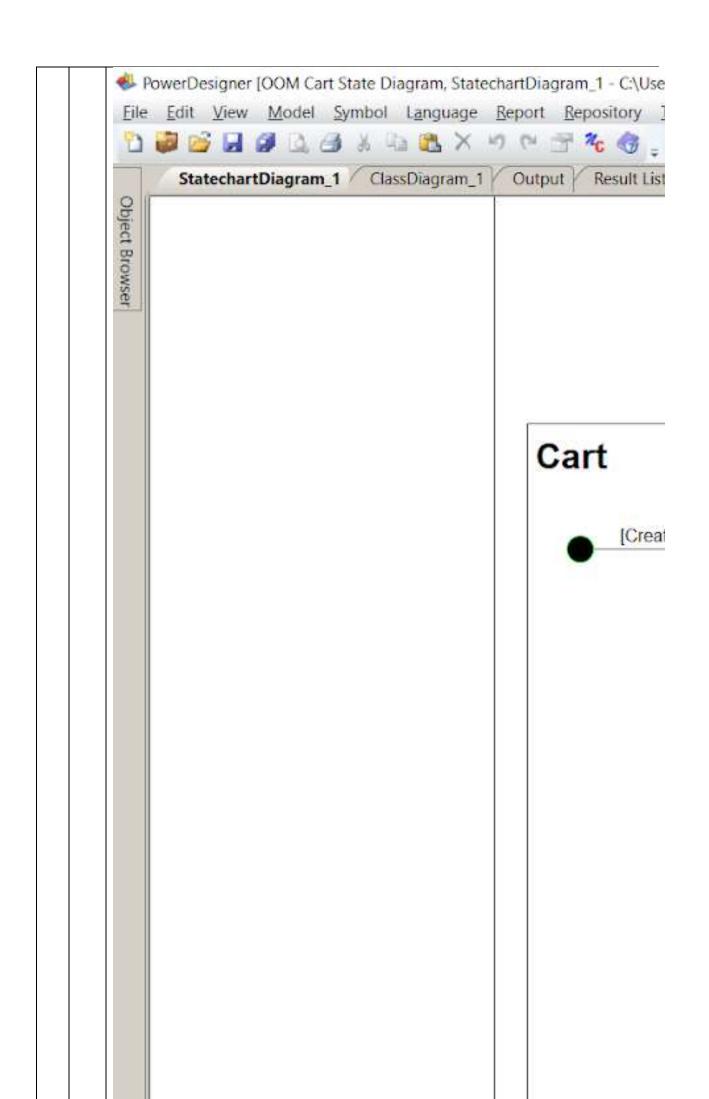




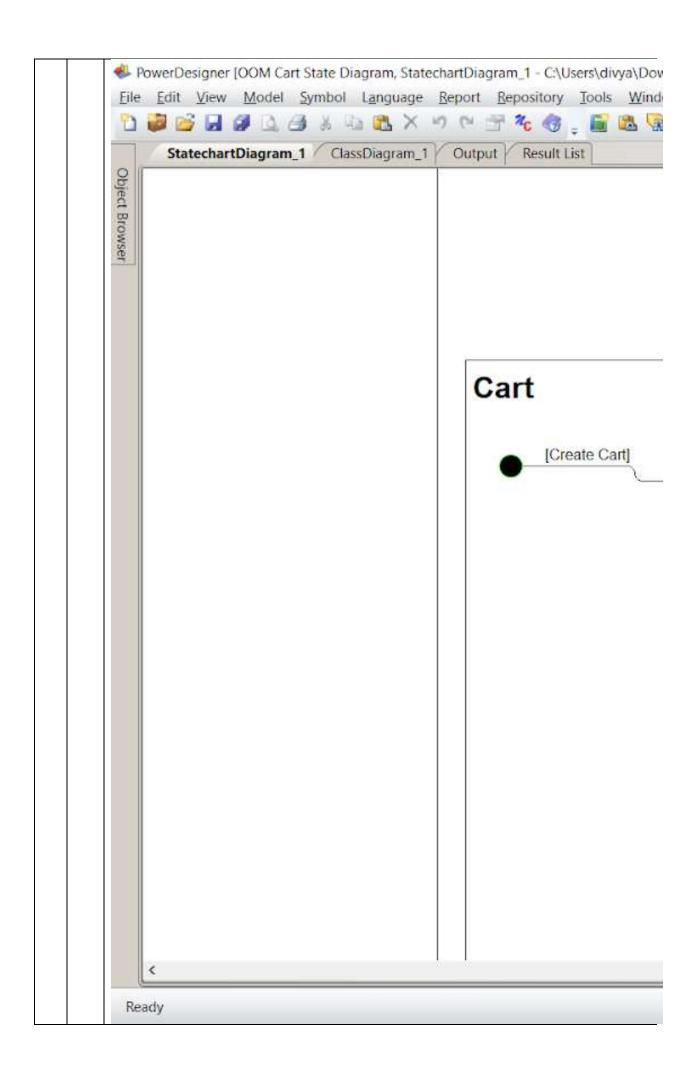
		State Diagram Key
EI	De	Key
е	SC	
m	rip	
е	tio	
nt	n	



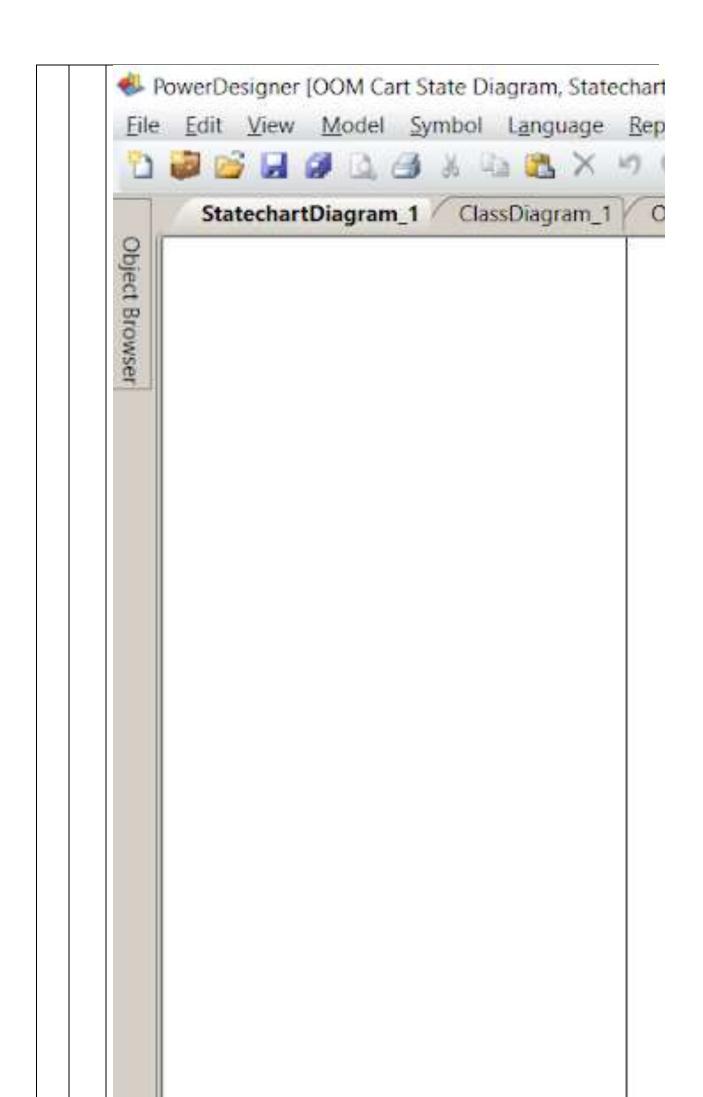
Tr	BI
an	ac
sit	k
io	Lin
n	e
''	wit
	h
	sol
	id
	bla
	ck
	arr
	OW
	he
	ad
	an
	d
	bla
	ck
	tex
	t.



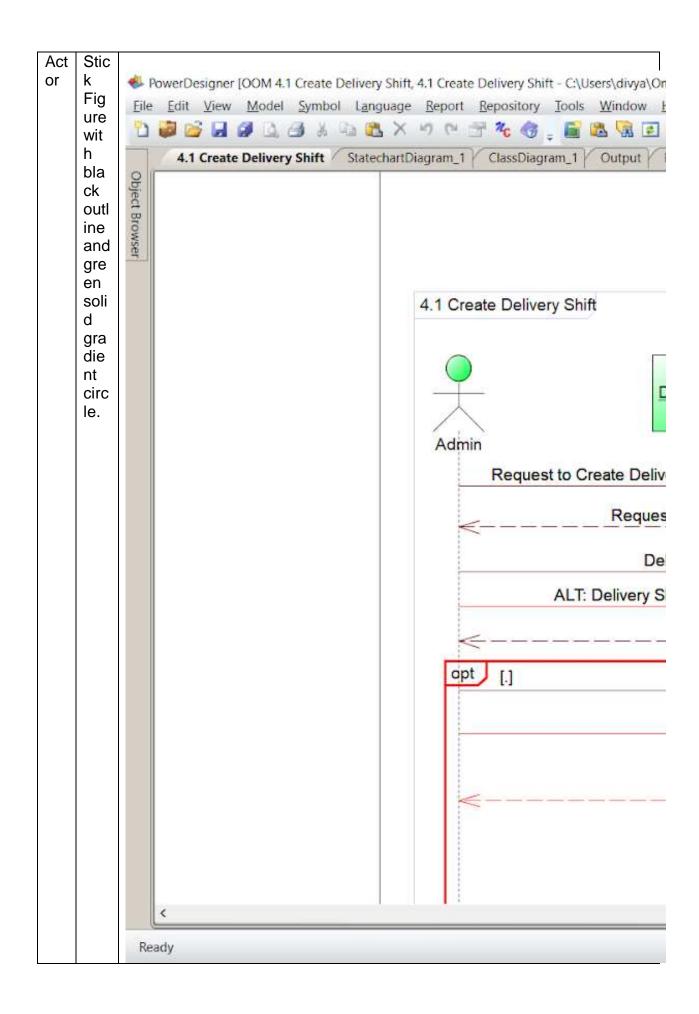
St	Cir
ar	cle
t	fill
Ν	ed
od	bla
е	ck
	wit
	h
	gr
	ee
	n
	out
	lin
	e.

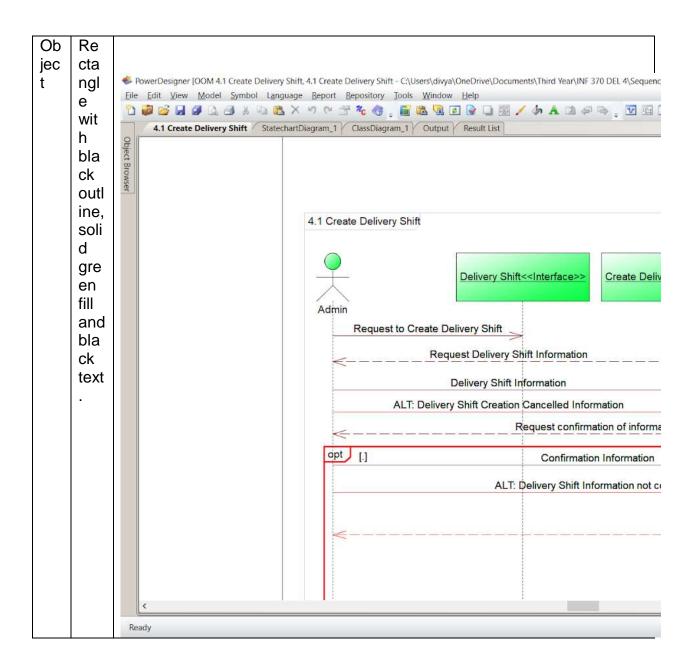


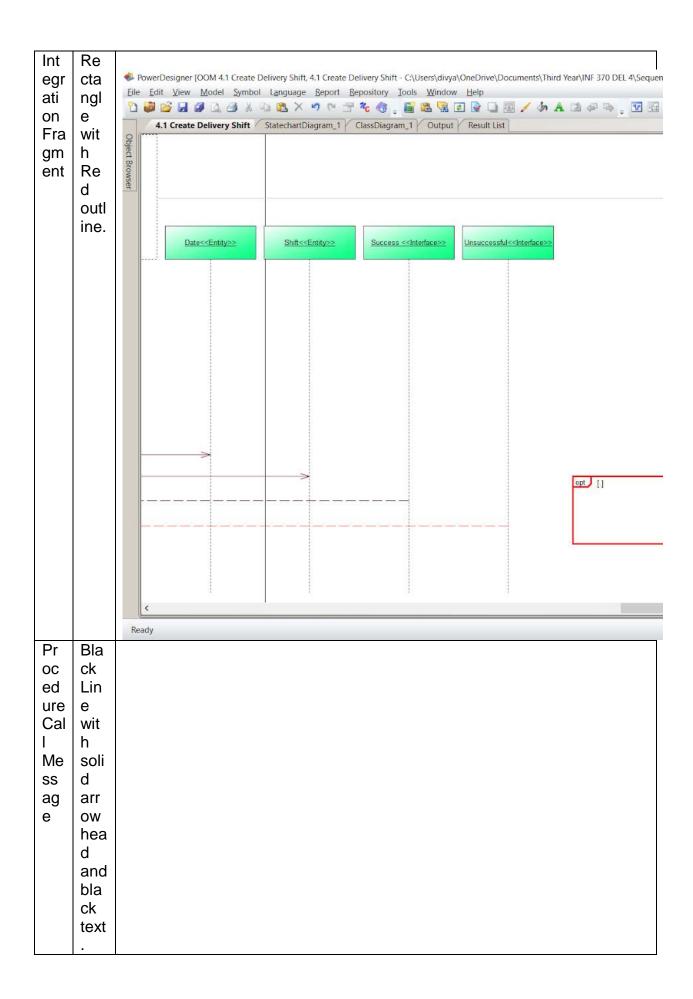
Ε	So
nd	lid
N	BI
od	ac
e	k
	Cir
	cle
	wit
	h
	Gr
	ee
	n
	out
	er
	cir
	cle

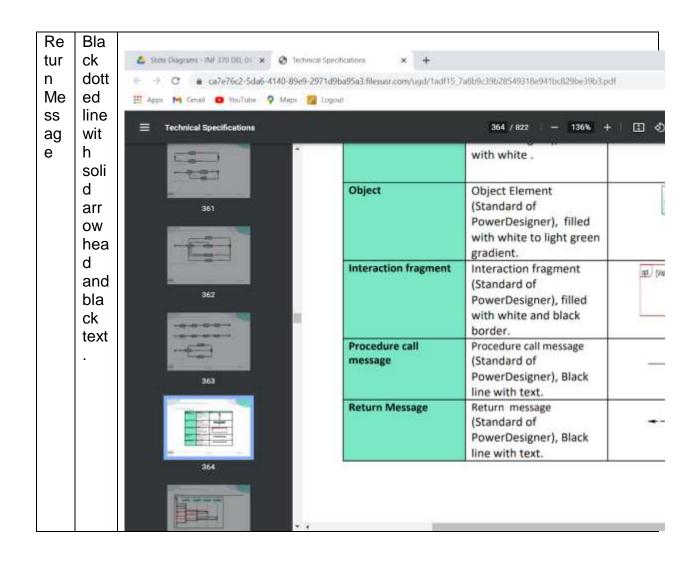


			Sequence Diagram Key	
Ele	De	Key		
me	scr			
nt	ipti			
	on			

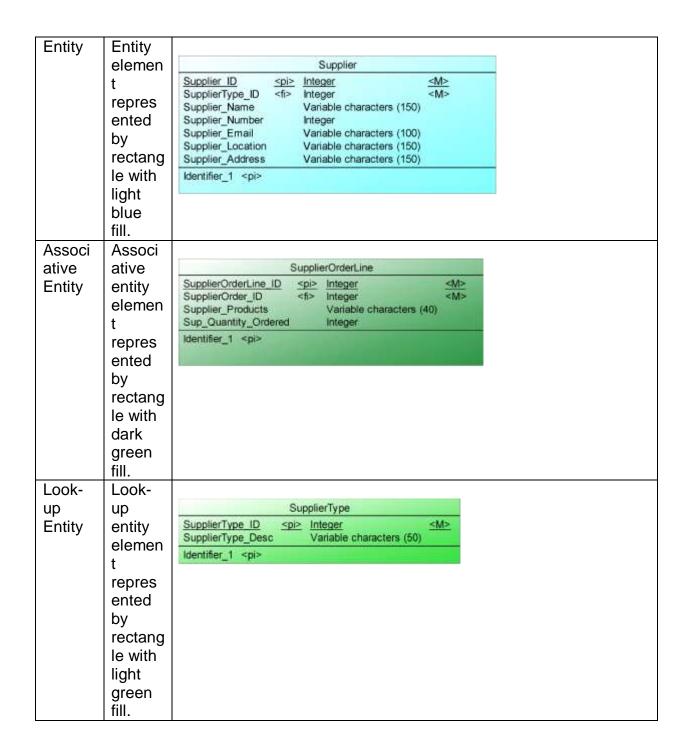


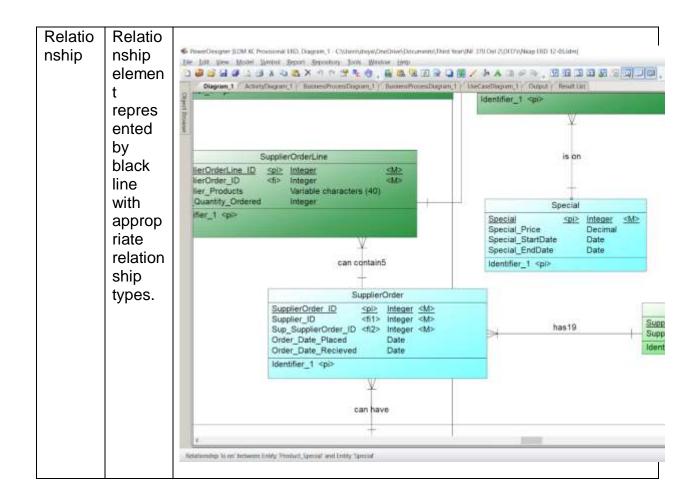






	Data Model Key						
Eleme	Descri	Example					
nt	ption						





2.2. OBJECT CLASS DIAGRAM

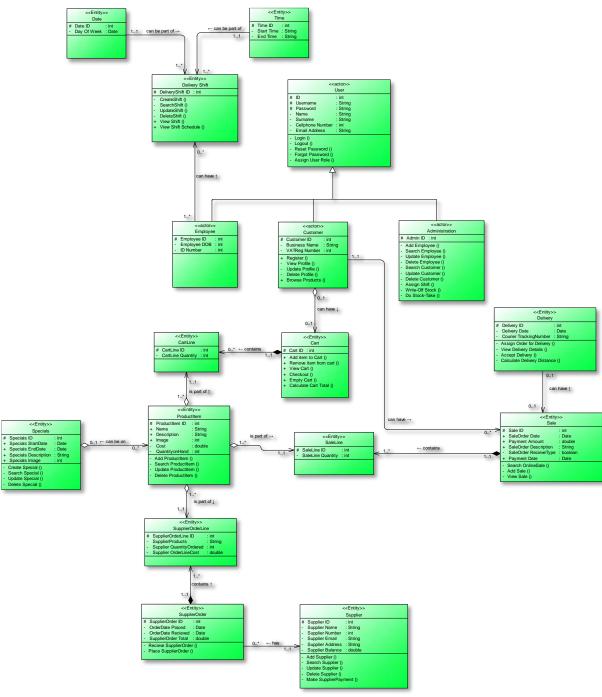


Figure 11 Object Class Diagram

2.3. STATE DIAGRAM

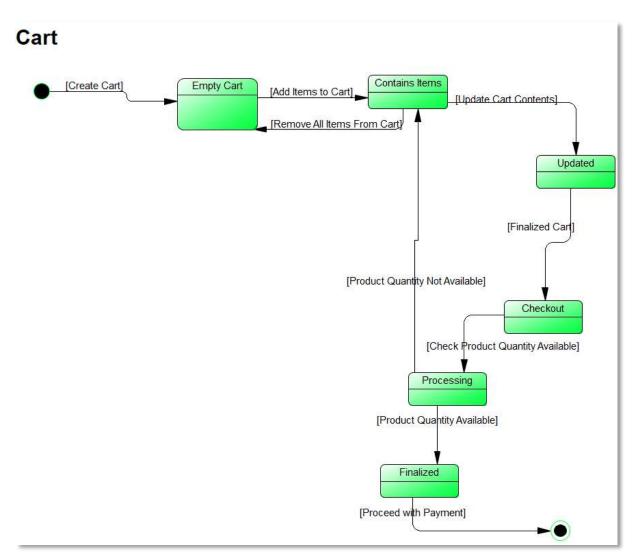


Figure 12 Cart State Diagram

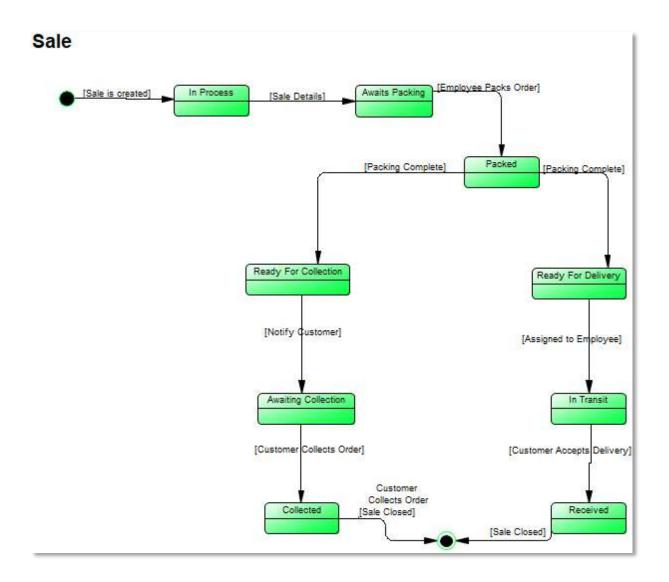


Figure 13 Sale State Diagram

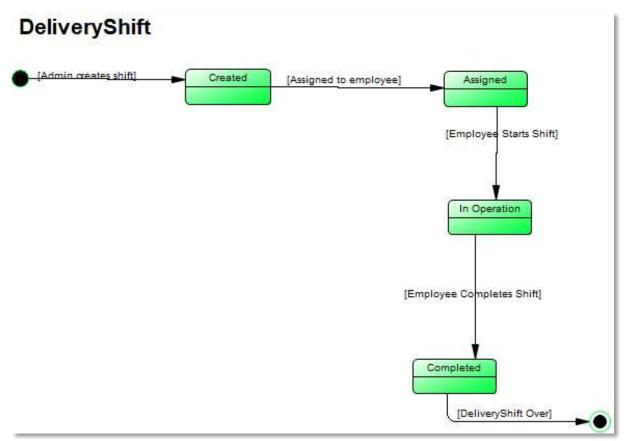


Figure 14 Delivery Shift State Diagram

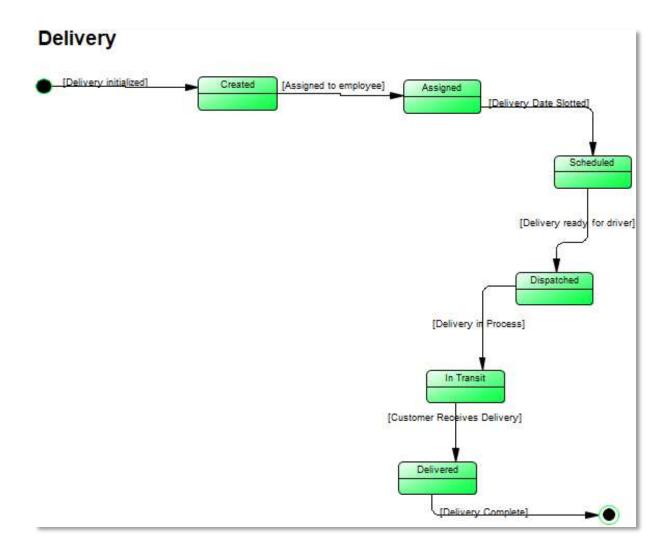


Figure 15 Delivery State Diagram

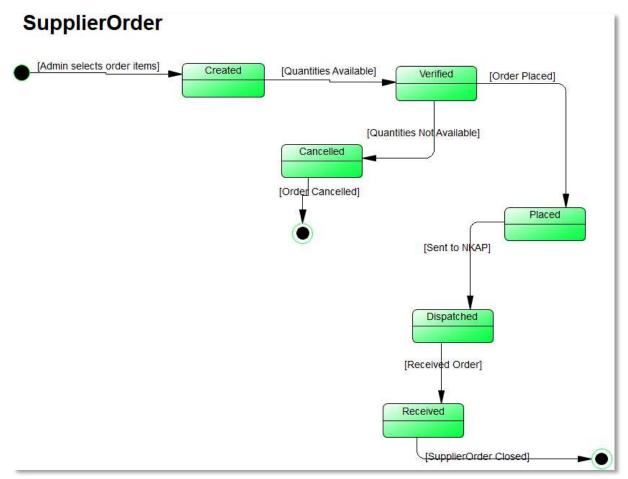


Figure 16 Supplier Order State Diagram

2.4. SEQUENCE DIAGRAM

2.4.1. Login Subsystem

Login

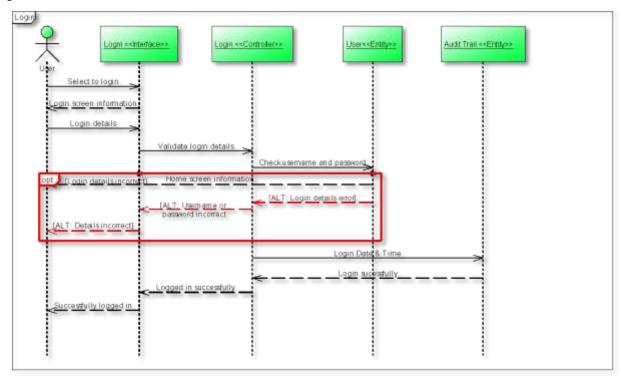


Figure 17 1.1. Login Sequence Diagram

Forgot Password

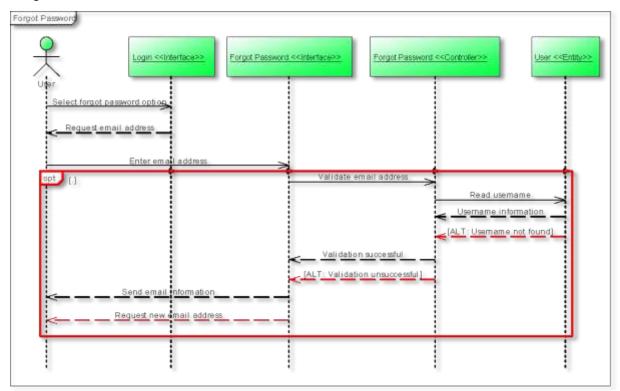


Figure 18 1.2. Forgot Password Sequence Diagram

Reset Password

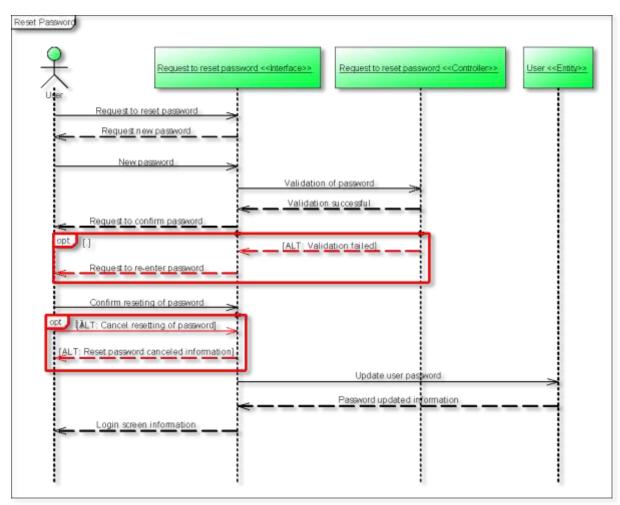


Figure 19 1.3. Reset Password Sequence Diagram

Logout

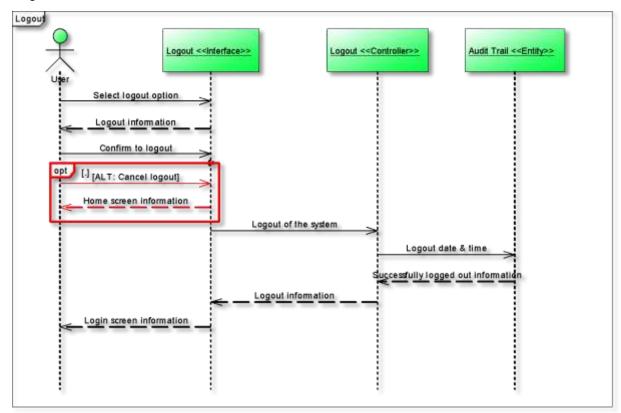


Figure 20 1.4. Logout Sequence Diagram

2.4.2. Supplier Subsystem

PASTE SEQUENCE DIAGRAMS ABOVE THE FOLLOWING CAPTIONS. DO NOT ERASE THE CAPTIONS!!

Add Supplier
< <insert diagram="" here="">></insert>
Figure 21 2.1. Add Supplier Sequence Diagram
Search Supplier
< <insert diagram="" here="">></insert>
Figure 22 2.2. Search Supplier Sequence Diagram
Update Supplier
< <insert diagram="" here="">></insert>
Figure 23 2.3. Update Supplier Sequence Diagram
Delete Supplier
< <insert diagram="" here="">></insert>
Figure 24 2.4. Delete Supplier Sequence Diagram
Place Supplier Order
< <insert diagram="" here="">></insert>
Figure 25 2.5. Place Supplier Order Sequence Diagram

Receive Supplier Order

<<insert Diagram here>>

Figure 26 2.6. Receive Supplier Order Sequence Diagram

Capture Supplier Order <<insert Diagram here>>

Figure 27 2.7. Capture Supplier Payment Sequence Diagram

2.4.3. User Subsystem

Register New Customer

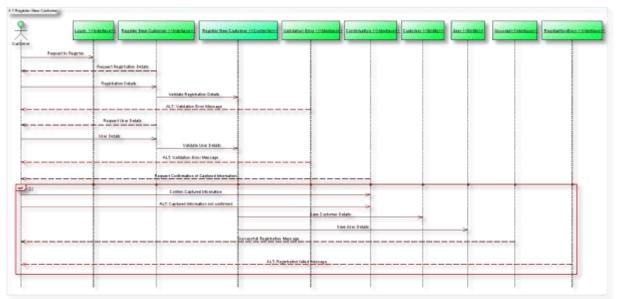


Figure 28 3.1. Register New Customer Sequence Diagram

Search Customer

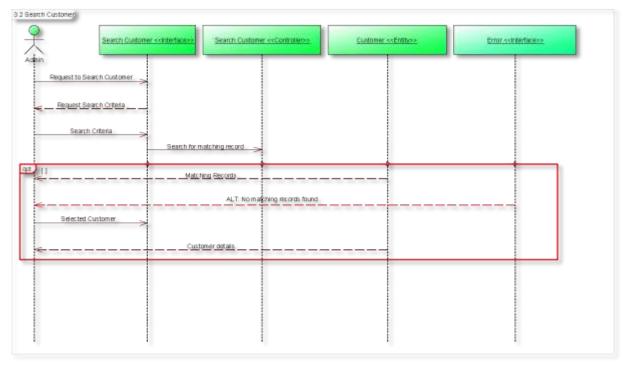


Figure 29 3.2. Search Customer Sequence Diagram

Update Customer

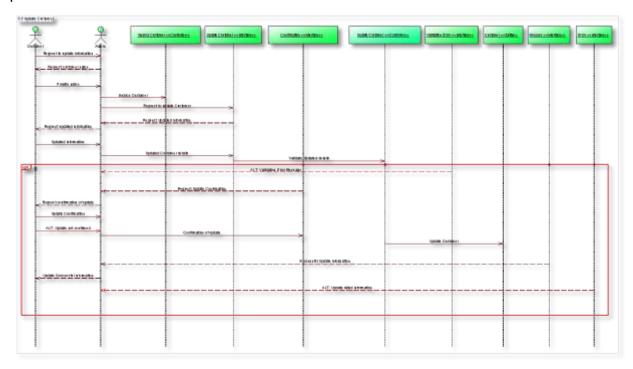


Figure 30 3.3. Update Customer Sequence Diagram

Delete Customer

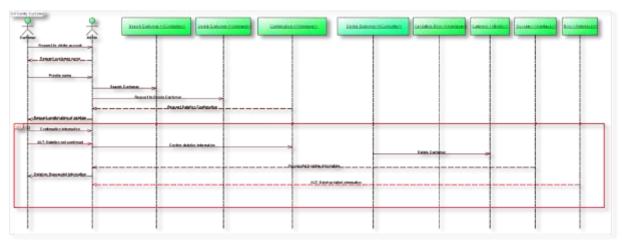


Figure 31 3.4. Delete Customer Sequence Diagram

View Profile

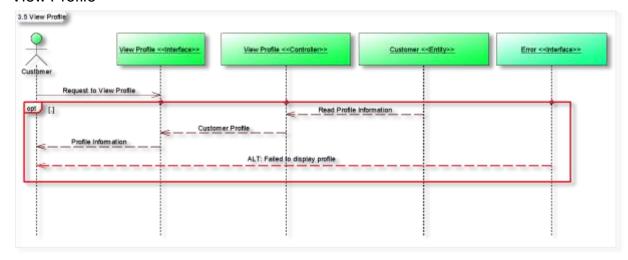


Figure 32 3.5. View Profile Sequence Diagram

Update Profile

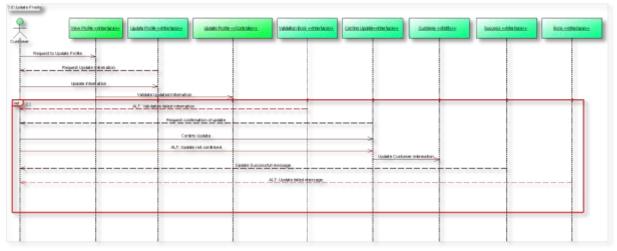


Figure 33 3.6. Update Profile Sequence Diagram

Delete Profile

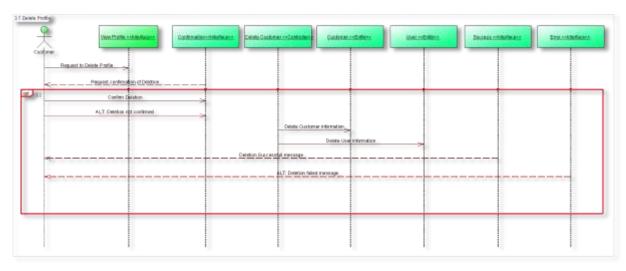


Figure 34 3.7. Delete Profile Sequence Diagram

Add User Role

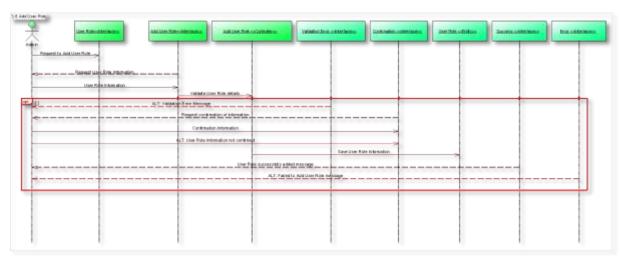


Figure 35 3.8. Add User Role Sequence Diagram

Search User Role

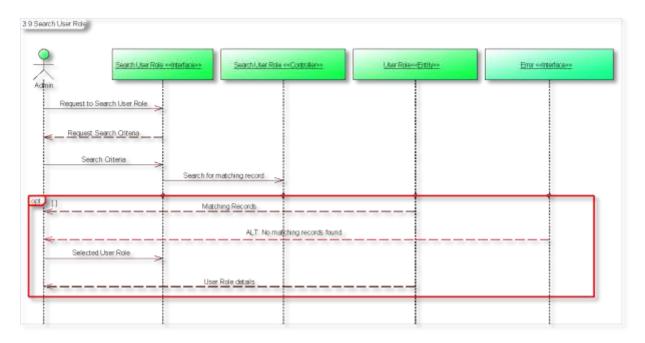


Figure 36 3.9. Search User Role Sequence Diagram

Update User Role

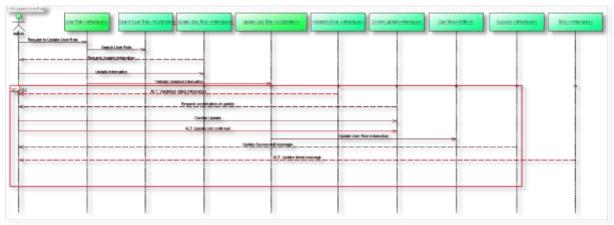


Figure 37 3.10. Update User Role Sequence Diagram

Delete User Role

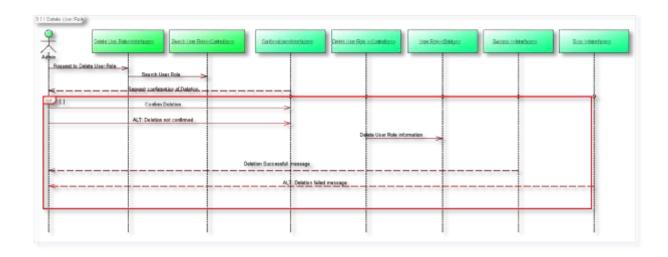


Figure 38 3.11. Delete User Role Sequence Diagram

2.4.4. Employee Delivery Shifts Subsystem

Create Delivery Shift



Figure 39 4.1. Create Delivery Shift Sequence Diagram

Search Delivery Shift

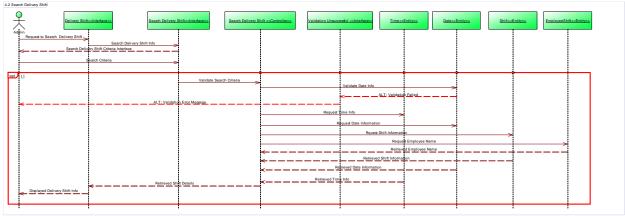
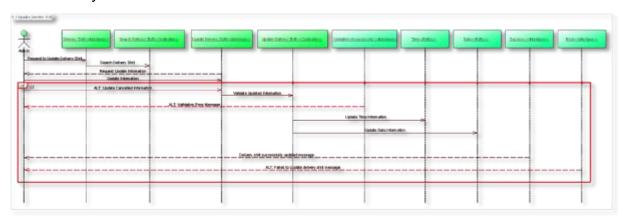


Figure 40 4.2. Search Delivery Shift Sequence Diagram

Update Delivery Shift



Delete Delivery Shift

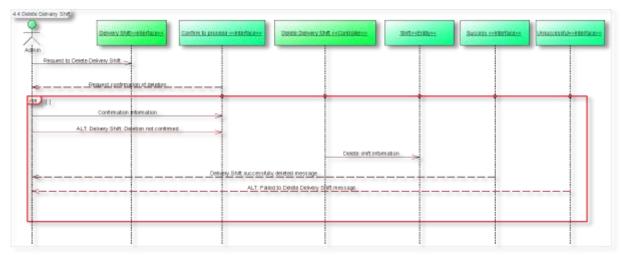


Figure 42 4.4. Delete Delivery Shift Sequence Diagram

Add Employee

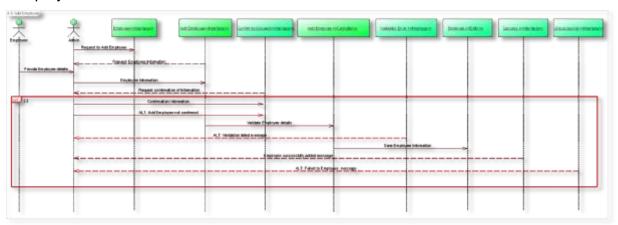


Figure 43 4.5. Add Employee Sequence Diagram

Search Employee

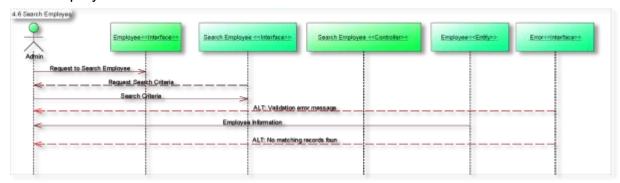


Figure 44 4.6. Search Employee Sequence Diagram

Update Employee

<<insert Diagram here>>

Figure 45 4.7. Update Employee Sequence Diagram

Delete Employee

<<insert Diagram here>>

Figure 46 4.8. Delete Employee Sequence Diagram

Assign Order for Delivery/Courier

<<insert Diagram here>>

Figure 47 4.9. Assign Order for Delivery/Courier Sequence Diagram

Assign Delivery Shift <<insert Diagram here>>

Figure 48 4.10. Assign Delivery Shift Sequence Diagram

Cancel Delivery Shift

<<insert Diagram here>>

Figure 49 4.11. Cancel Delivery Shift Sequence Diagram

2.4.5. Admin Subsystem

Search Online Sales

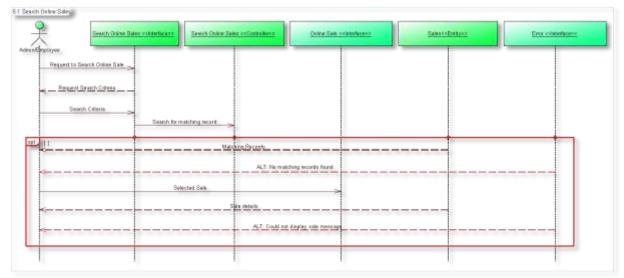


Figure 50 5.1. Search Online Sales Sequence Diagram

Pack Order



Figure 51 5.2. Pack Order Sequence Diagram

Do Stock-take

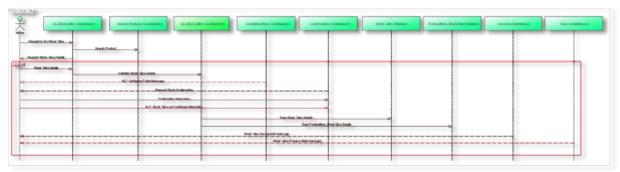


Figure 52 5.3. Do Stock-take Sequence Diagram

View Delivery Shift Schedule

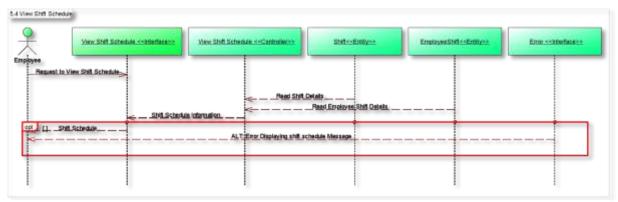


Figure 53 5.4. View Delivery Shift Schedule Sequence Diagram

Write-Off Stock



Figure 54 5.5. Write-off Stock Sequence Diagram

2.4.6. Customer Subsystem

View Products

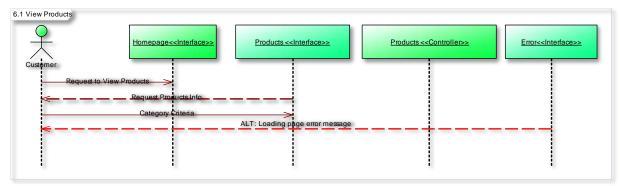


Figure 55 6.1. View Products Sequence Diagram

View Product Details

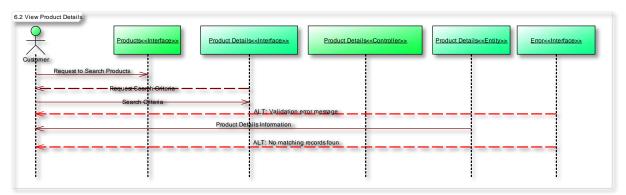


Figure 56 6.2. View Product Details Sequence Diagram

Search Specials and Promotions

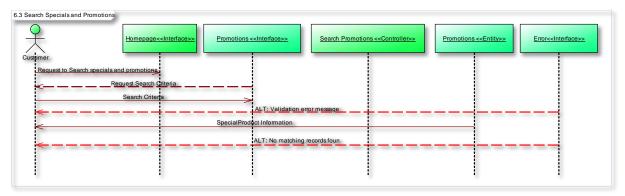


Figure 57 6.3. Search Specials and Promotions Sequence Diagram

Search Cart

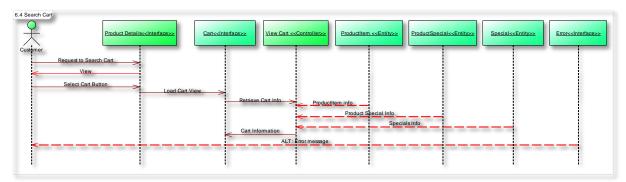


Figure 58 6.4. Search Cart Sequence Diagram

Add Item to Cart

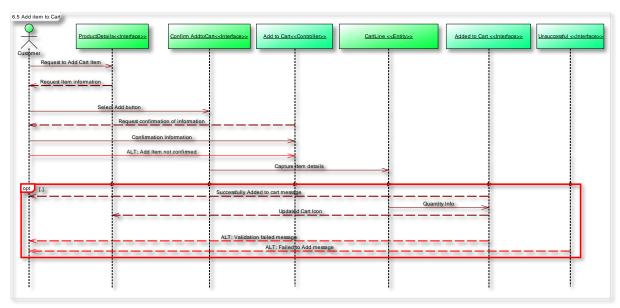


Figure 59 6.5. Add Item to Cart Sequence Diagram

Remove Item from Cart

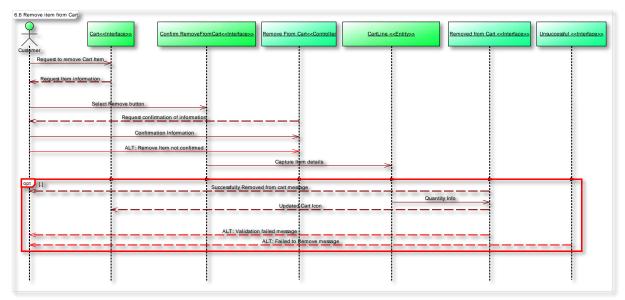


Figure 60 6.6. Remove Item from Cart Sequence Diagram

Checkout Order

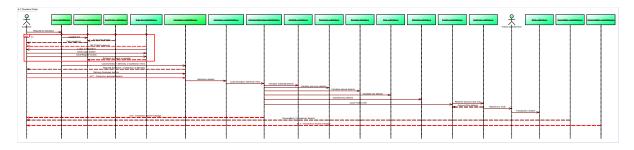


Figure 61 6.7. Checkout Order Sequence Diagram

2.4.7. Reports Subsystem

Generate Fast Selling Product List

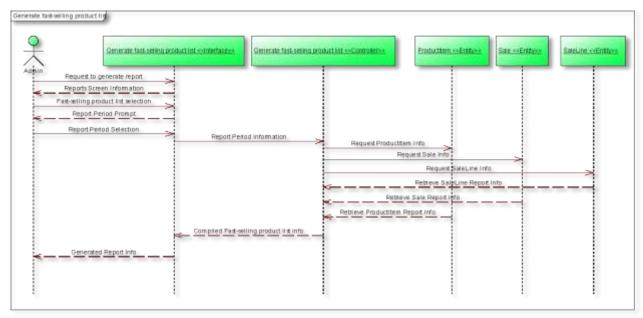


Figure 62 7.1. Generate Fast Selling Product List Sequence Diagram

Generate Slow Selling Product List

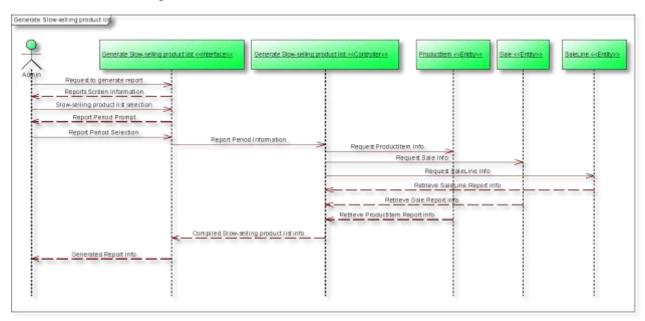


Figure 63 7.2. Generate Slow Selling Product List Sequence Diagram

Generate Most Frequent Buyers List

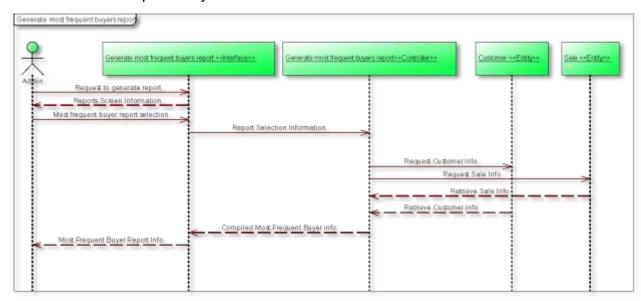


Figure 64 7.3. Generate Most Frequent Buyers List Sequence Diagram

Generate Most Popular Location Graph

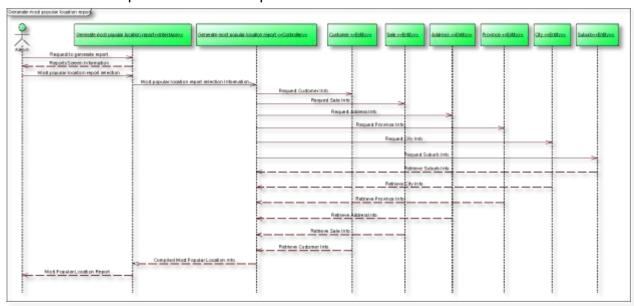


Figure 65 7.4. Generate Most Popular Location Graph Sequence Diagram

Generate Weekly Sale Orders List

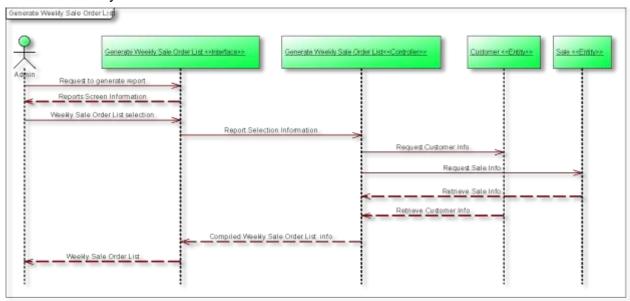


Figure 66 7.5. Generate Weekly Sale Order List Sequence Diagram

Generate Monthly Sale Orders List

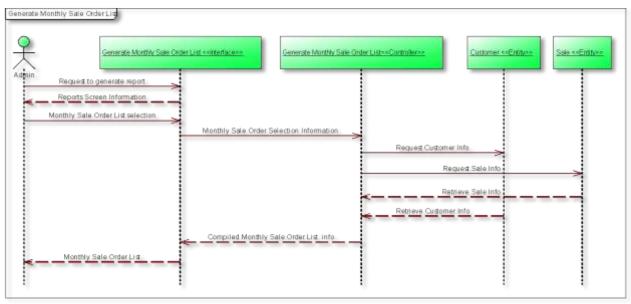


Figure 67 7.6. Generate Monthly Sale Orders List Sequence Diagram

Generate Stock-Level Report

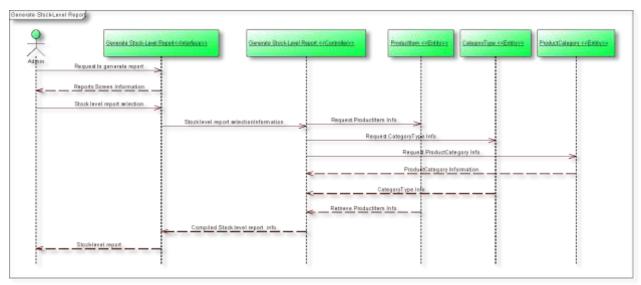


Figure 68 7.7. Generate Stock-level Report Sequence Diagram

Packing Report

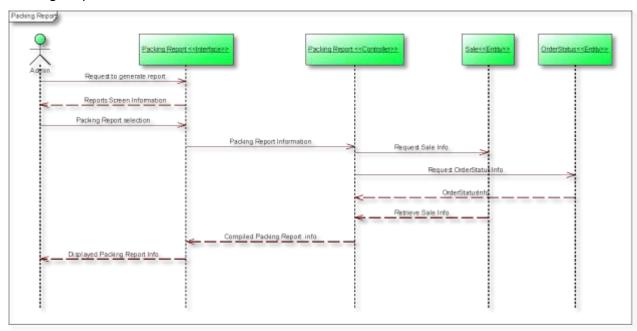


Figure 69 7.8. Packing Report Sequence Diagram

Delivery Report

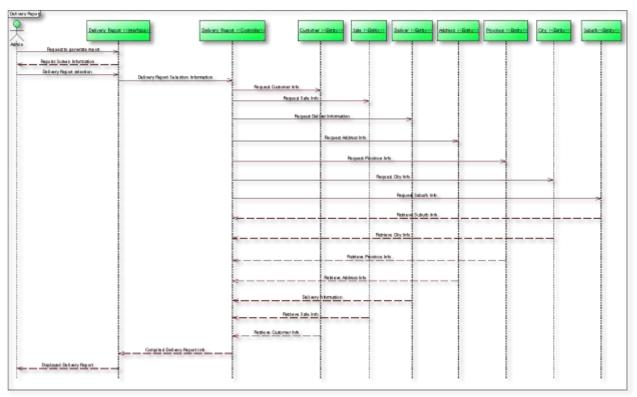


Figure 70 7.9. Delivery Report Sequence Diagram

2.4.8. Delivery Subsystem

8.1. Search Delivery Details

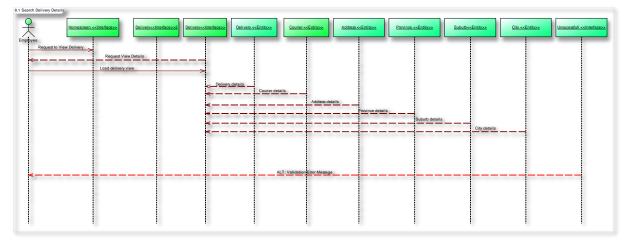


Figure 71 8.1. Search Delivery Details Sequence Diagram

8.2. Accept Delivery

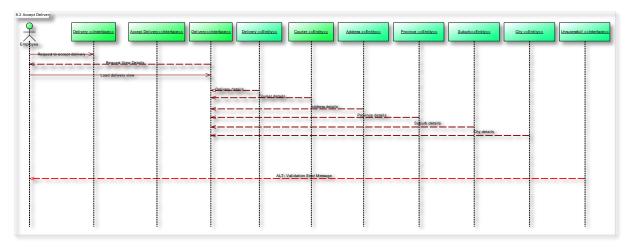


Figure 72 8.2. Accept Delivery Sequence Diagram

Add Courier

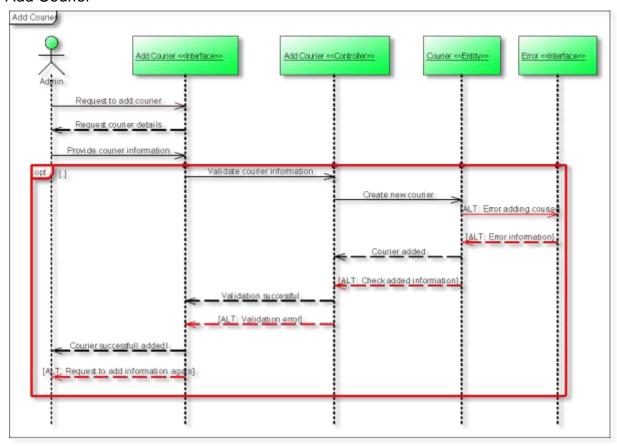


Figure 73 8.3. Add Courier Sequence Diagram

Search Courier

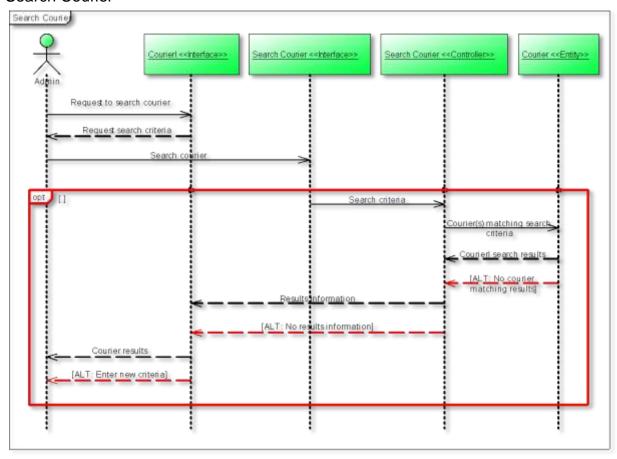


Figure 74 8.4. Search Courier Sequence Diagram

Update courier

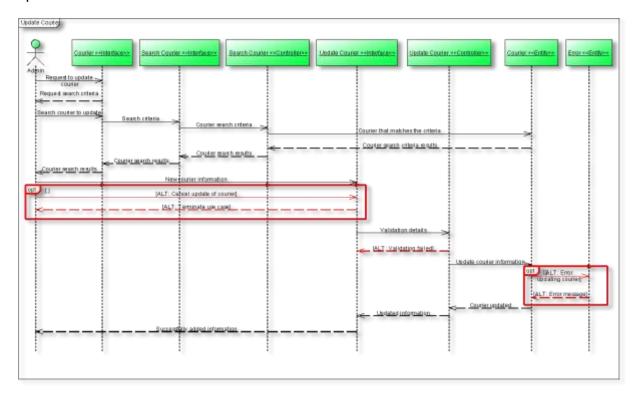


Figure 75 8.5. Update Courier Sequence Diagram

Delete Courier

<<insert diagram here>>

Figure 76 8.6. Delete Courier Sequence Diagram

2.4.9. Products Subsystem

Add Product Category

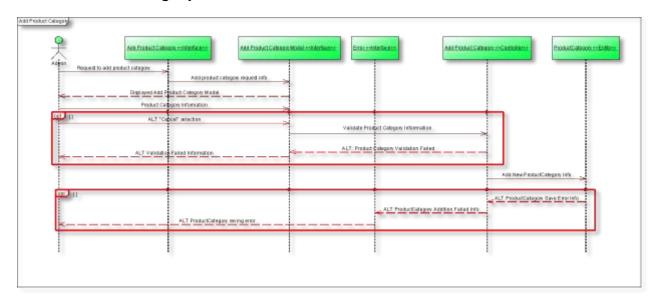


Figure 77 9.1. Add Product Category Sequence Diagram

Search Product Category

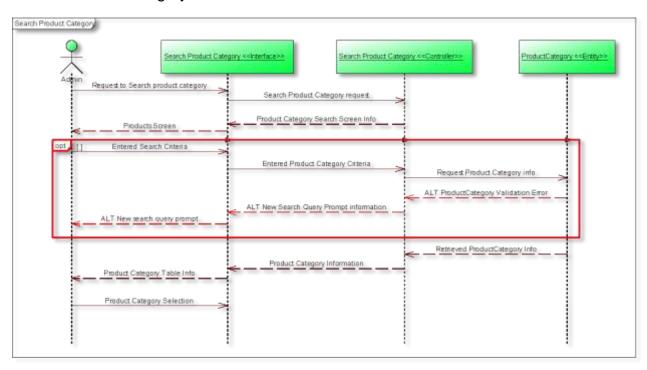


Figure 78 9.2. Search Product Category Sequence Diagram

Update Product Category

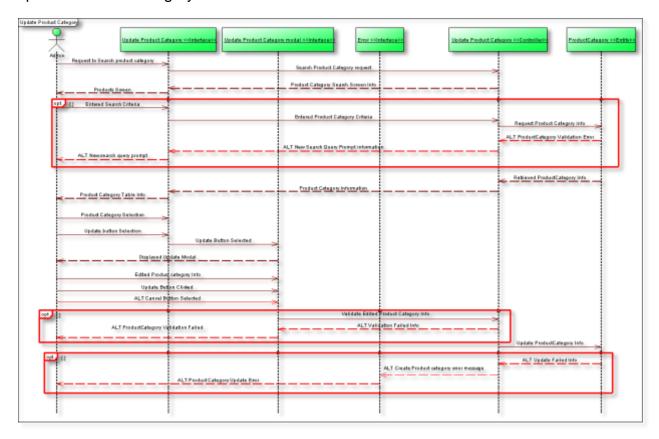


Figure 79 9.3. Update Product Category Sequence Diagram

Delete Product Category

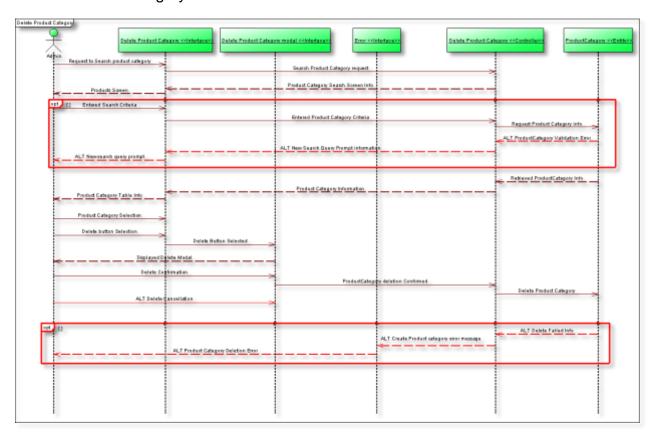


Figure 80 9.4. Delete Product Category Sequence Diagram

Add Category Type

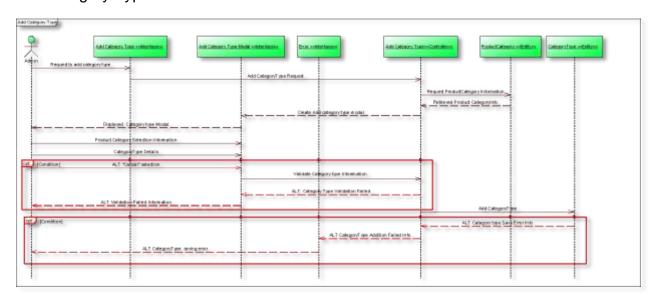


Figure 81 9.5. Add Category Type Sequence Diagram

Search Category Type

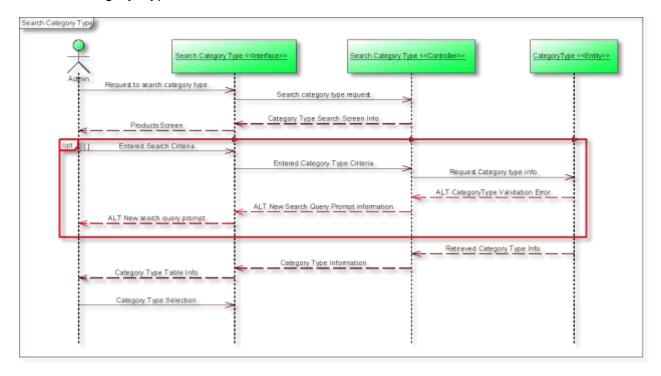


Figure 82 9.6. Search Category Type Sequence Diagram

Update Category Type

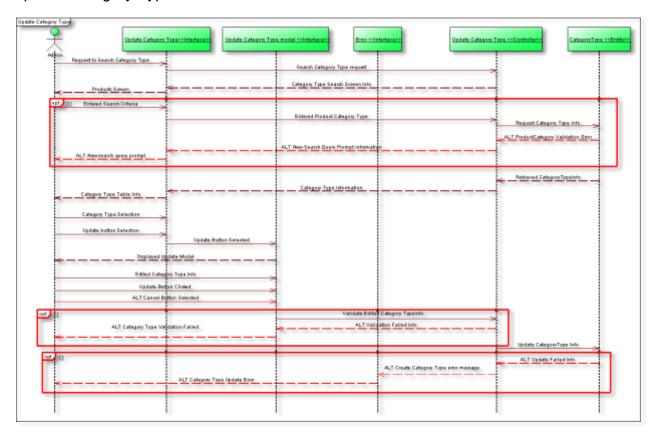


Figure 83 9.7. Update Category Type Sequence Diagram

Delete Category Type

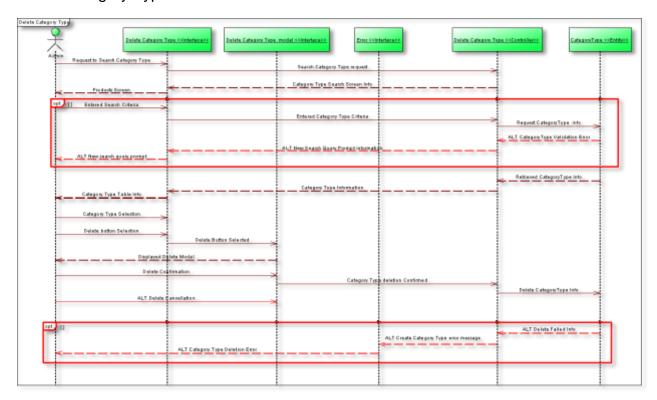


Figure 84 9.8. Delete Category Type Sequence Diagram

Add Product Item

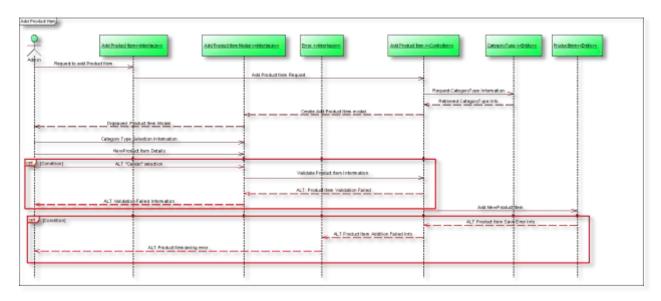


Figure 85 9.9. Add Product Item Sequence Diagram

Search Product Item

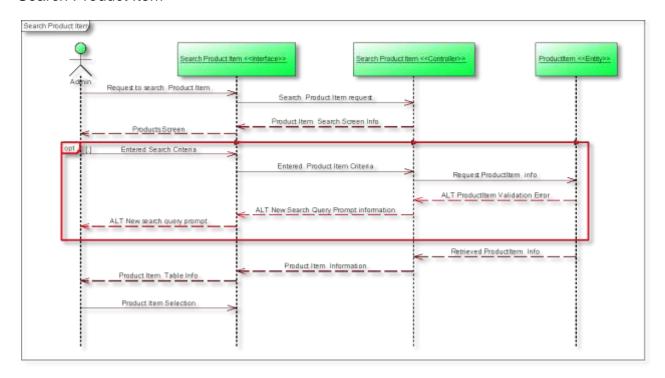


Figure 86 9.10. Search Product Item Sequence Diagram

Update Product Item

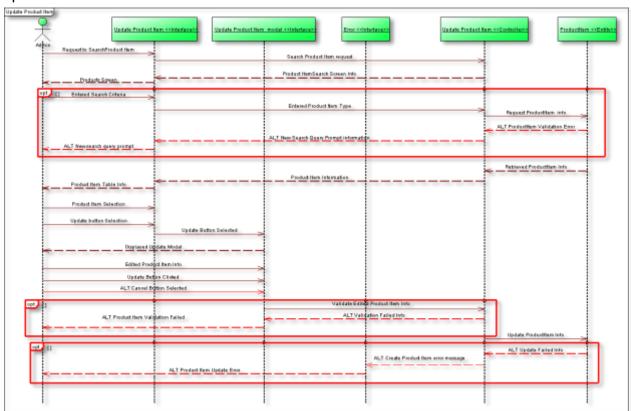


Figure 87 9.11. Update Product Item Sequence Diagram

Delete Product Item

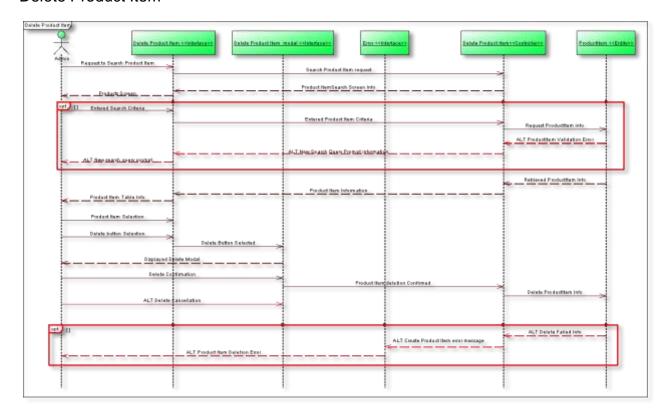


Figure 88 9.12. Delete Product Item Sequence Diagram

2.4.10. Specials Subsystem

Add Special

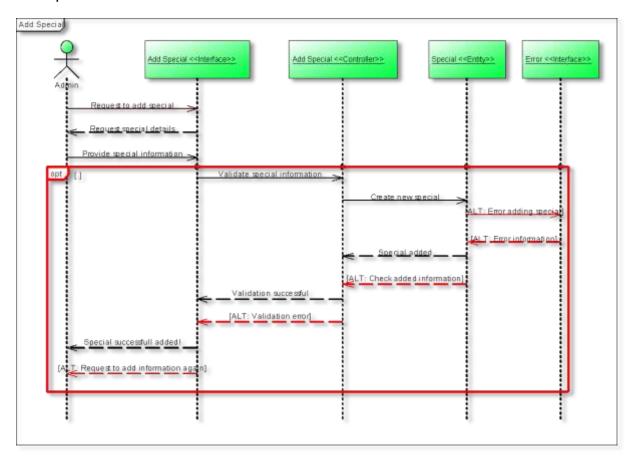


Figure 89 10.1. Add Special Sequence Diagram

Search Special

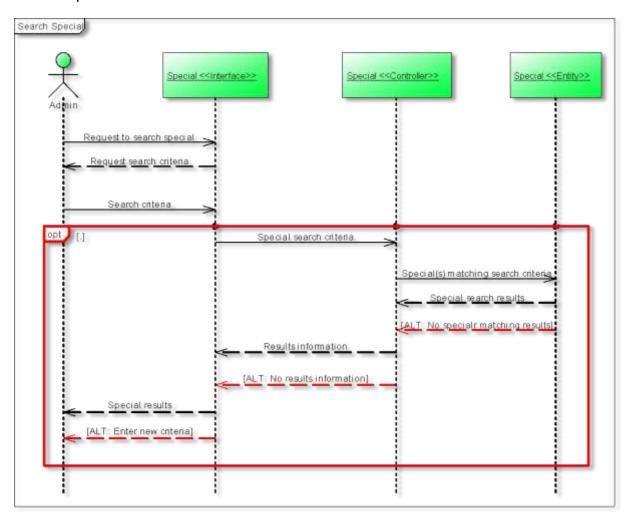


Figure 90 10.2. Search Special Sequence Diagram

Update Special

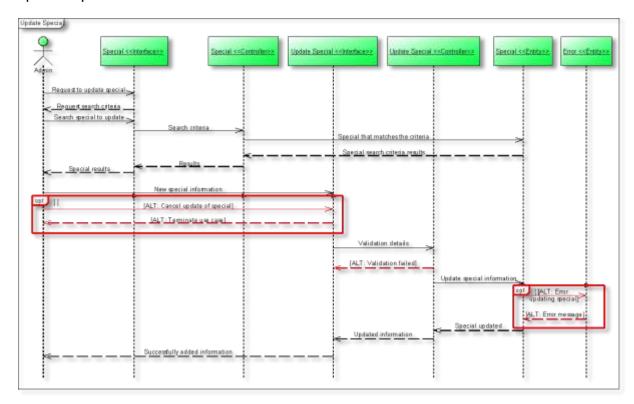


Figure 91 10.3. Update Special Sequence Diagram

Delete Special

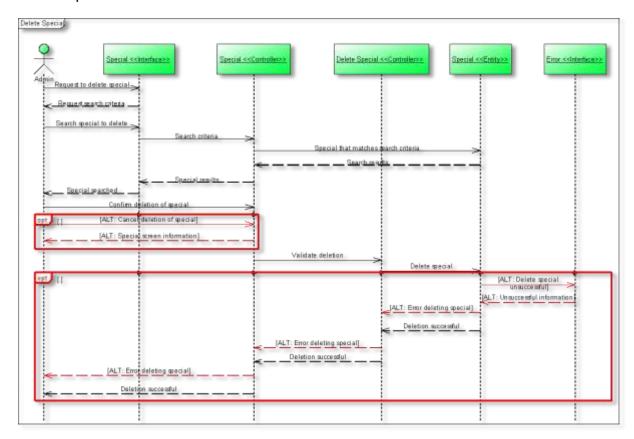


Figure 92 10.4. Delete Special Sequence Diagram

Send Promotional Emails

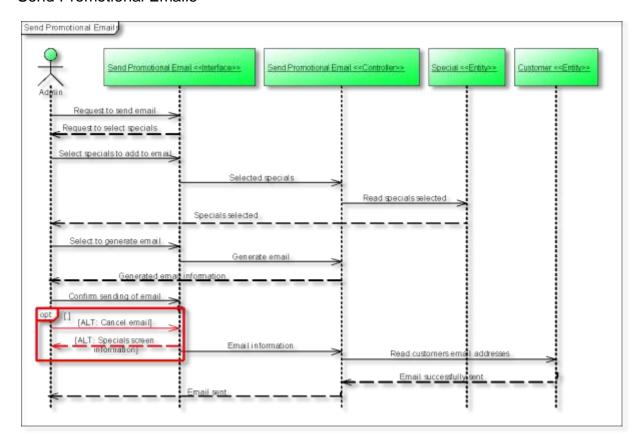


Figure 93 10.5. Send Promotional Emails Sequence Diagram

2.5. PSEUDO CODE

2.5.1. LOGIN SUBSYSTEM

3. Subsystem:	1. Login
Use Case:	1.1 Login
Language:	C#
Pseudo Code:	Step 1: Request to login [Angular]
	ON CLICK Login button
	THEN
	Step 2: Display Login screen [Angular]
	Display Login screen.
	Step 3: Login [Angular]
	ON CLICK Login button.
	THEN
	Step 4: Validate login details
	SQL SELECT
	Username and Password
	From User entity
	WHERE Username == User input && Password == User Input
	If (User input == Valid)
	Generate AuditTrail_ID
	THEN
	SQL ADD
	AuditTrail_ID
	INTO User entity

[ALT Step 4] :Display validation error message [Angular]

Else (User input == NotValid)

Display error message "The entered login credentials are incorrect. Please try again."

THEN

Step 5: Add the login to the AuditTrail entity

SQL ADD

Logged in users' Username, Date and Time In the AuditTrail table.

THEN

System logs the user in to the system.

End use case.

Subsystem:	1. Login
Use Case:	1.2 Forgot Password
Language:	C#
Pseudo Code:	Step 1: User has forgotten their password.
	THEN
	Step 2: Display Login Screen [Angular]
	Display Login screen.
	Step 3: User clicks on the "Forgot Password" link
	ON CLICK "Forgot Password" link
	THEN
	Stop 4: Diaplay Forget Bassword sarson [Angular]
	Step 4: Display Forgot Password screen [Angular]
	Display Forgot Password screen THEN
	ITEN
	Step 5: User adds their email information
	Input email address
	ON CLICK "Submit" button.
	THEN
	Step 6: Validate reset details
	SQL SELECT Email Address
	FROM Customer/Admin Entity
	WHERE Email Address == User Input
	If(User Input == Valid)
	ALT Step 6]: Display invalid email message [Angular]

Else if(User Input == NotValid)

Display "Invalid email address. Please try again."

THEN

Step 7: Send an email to the email the user entered.

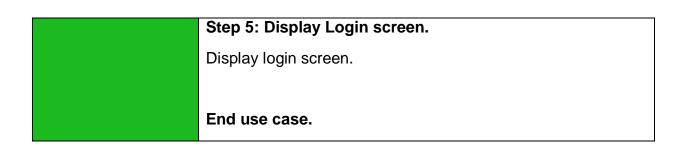
THEN

Step 8: Notify the user that an email has been sent to them.

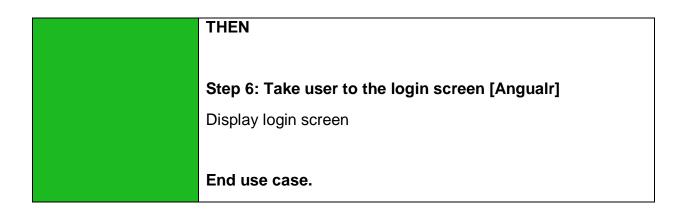
Display message "A password reset link has been sent to your email."

End use case.

Subsystem:	1. Login
Use Case:	1.3 Reset Password
Language:	C#
Pseudo Code:	Step 1: User clicks on the link sent in the email
	ON CLICK "Reset Password" link
	THEN
	Step 2: Display Reset Password screen
	Display Reset Password screen
	THEN
	Step 3: User enters new password and confirmation
	Input new password and confirmations
	ONCLICK "Submit" button
	THEN
	Step 4: Validate password
	SQL SELECT
	Password
	FROM User entity
	WHERE Password == User
	If(User Input == Valid)
	[ALT Step 4]: Display validation error message [Angular]
	Else if (User Input) == NotValid)
	Display error message "Password does not match with confirm password"
	THEN



Subsystem:	1. Login
Use Case:	1.4 Logout
Language:	C#
Pseudo Code:	Step 1:
	Step 1: The user requests to Logout of the system [Angular]
	ON Click "Logout" option
	THEN
	Step 2: Display logout confirmation pop-up [Angular]
	Display pop-up with the following message:
	"Are you sure you want to logout?"
	THEN
	Step 3: User confirms log out [Angular]
	ON CLICK "Yes" button.
	THEN
	[ALT Step 3]: User cancels logout [Angular]
	ON CLICK "No" button.
	THEN Take away the confirmation pop-up.
	Step 4: User gets logged out of the system.
	User is logged out.
	THEN
	Step 5: Information is added to the Audit Trail
	SQL ADD
	Logged out users' Username, Date and Time
	In the AuditTrail table.



3.1.1. SUPPLIER SUBSYTEM

3.1.2. USER SUBSYSTEM

Subsystem:	3. User
Use Case:	3.1 Register New Customer
Language:	C#
Pseudo Code:	Step 1: Customer wants to Register.
	THEN
	Step 2: Customer clicks User Icon (Angular)
	ONCLICK User icon on navigation bar
	THEN
	Step 3: Display Login Dropdown menu item (Angular)
	Display Login dropdown menu item
	Step 4: Customer clicks Login dropdown menu item (Angular)
	ONCLICK Login dropdown menu item
	THEN ONCLICK Register Button
	THEN
	Step 5: Display Register New Customer Screen (Angular)
	Display Register New Customer Screen
	THEN
	Step 6: Provide Customer Details (Angular)
	Provide Customer Details
	ONCLICK Next Button
	THEN
	Step 7: Validate Details (C#)
	If (Input fields == Valid)
	Continue to step 8
	Else If (Input fields == false)
	Display Validation Error Modal "Entered details are in the incorrect format"

THEN

Step 8: Display Register New Customer Screen (Angular)

Display Register New Customer Screen

THEN

Step 9: Provide User Details (Angular)

Provide User Details

ONCLICK Register Button

THEN

Step 10: Validate Details (C#)

If (Input fields Valid == True)

Continue to step 11

Else If (Input fields Valid == false)

Display Validation Error Modal "Entered details are in the incorrect format"

THEN

Step 11: Display Confirm Details Modal (Angular)

Display Confirm Details Modal " Is your captured information correct?"

THEN

Step 12: Confirm Details (Angular)

Confirm Details

ONCLICK Confirm Button

[ALT] ONCLICK Cancel Button → Terminate Use Case

THEN

Step 13: Save Customer Details and Generate CustomerID and UserID (C#)

SQL INSERT

Title, Name, Surname, Cell Phone Number,

Email Address, VAT Registration Number,

Business Name

IN CUSTOMER ENTITY

SQL INSERT

Username, Password IN USER ENTITY

THEN

Step 14: Display Success Modal (Angular)

IF (Registration Successful == True)

Display Success Modal "You have successfully registered."

ELSE IF (Registration Successful == False)

Display Error Modal "An unexpected error occurred when saving the customer."

Use Case:	3.2 Search Customer
Language:	C#
Pseudo Code:	Step 1: Admin wants to Search for a Customer
	THEN
	Step 2: Admin clicks User Menu Item (Angular)
	ONCLICK User Menu Item
	THEN ONCLICK Search Customer Dropdown menu Item
	THEN
	Step 3: Display Search Customer Screen (Angular)
	Display Search Customer Screen
	THEN
	Step 4: Admin Provides Search Criteria (Angular)
	Provide Search Criteria
	ONCLICK Search Button
	THEN
	Step 5: Search for matching records (C#)
	SQL READ
	Matching Records
	FROM CUSTOMER ENTITY
	Step 6: Display matching records (Angular)
	IF (Matching Records Found == True)
	Display Search Customer Table
	ELSE IF (Matching Records Found == False)
	Display Error Modal" No customers matching the search criteria was found."
	THEN
	Step 7: Admin Selects Customer to View (Angular)
	Select Customer to View
	ONCLICK View Button
	THEN

Step 8: Display View Customer Screen (Angular) SQL READ

Title, Name, Surname, Cell Phone Number,

Email Address, VAT Registration Number,

Business Name

FROM CUSTOMER ENTITY

Use Case:	3.3 Update Customer
Language:	C#
Pseudo Code:	Step 1: The admin searches for a customer using the Customer Name Provided (C#)
	THEN
	Step 2: Invoke Use Case 3.2 Search Customer (C#)
	THEN
	Step 3: Admin requests to update Customer (Angular)
	Admin requests to update Customer
	ONCLICK Update Button
	THEN
	Step 4: Display Update Customer Screen (Angular)
	Display Update Customer Screen
	THEN
	Step 5: Admin provides update details (Angular)
	Admin provides update details.
	ONCLICK Update Button
	THEN
	Step 6: Validate Update Details (C#)
	If (Input fields == Valid)
	Continue to step 7
	Else If (Input fields == false)
	Display Validation Error Modal "Entered details are in the incorrect format."
	THEN
	Step 7: Display Confirm Update Modal (Angular)
	Display Confirm Update Modal "Are you sure you want to update the customer information?"
	THEN
	Step 8: Confirm Update (Angular)
	Admin Confirms Update

ONCLICK Confirm Button

THEN

Step 9: Update Customer Details (C#)

SQL UPDATE

Title, Name, Surname, Cell Phone Number,

Email Address, VAT Registration Number,

Business Name

IN CUSTOMER ENTITY

THEN

Step 10: Display Success Modal (Angular)

IF (Update Successful == True)

Display Success Modal "Customer information was successfully updated."

ELSE IF (Update Successful == False)

Display Error Modal "There was an unexpected error when trying to update the customer information."

Use Case:	3.4 Delete Customer
Language:	C#
Pseudo Code:	Step 1: The admin searches for a customer using the Customer Name Provided (C#)
	THEN
	Step 2: Invoke Use Case 3.2 Search Customer (C#)
	THEN
	Step 3: Admin requests to Delete Customer (Angular)
	Admin requests to Delete Customer
	ONCLICK Delete Button
	THEN
	Step 4: Display Confirm Update Modal (Angular)
	Display Confirm Deletion Modal "Are you sure you want to delete this customer?"
	THEN
	Step 5: Confirm Deletion (Angular)
	Admin Confirms Deletion
	ONCLICK Confirm Button
	Step 6: Delete Customer Details (C#)
	SQL DELETE
	Title, Name, Surname, Cell Phone Number,
	Email Address, VAT Registration Number,
	Business Name
	FROM CUSTOMER ENTITY
	SQL DELETE
	Username, Password
	IN USER ENTITY
	THEN
	Step 7: Display Success Modal (Angular)
	IF (Deletion Successful == True)

Display Success Modal "The customer was successfully deleted."

ELSE IF (Deletion Successful == False)

Display Error Modal "There was an unexpected error when trying to delete the customer."

Subsystem:	3. User
Use Case:	3.5 View Profile
Language:	C#
Pseudo Code:	Step 1: A customer requests to View Profile
	THEN
	Step 2: Customer Clicks User Icon (Angular)
	ONCLICK User Icon
	THEN ONCLICK View Profile DropDown Menu Item
	THEN
	Step 3: Read Customer Details (C#)
	SQL Read
	Title, Name, Surname, Cell Phone Number,
	Email Address, VAT Registration Number,
	Business Name
	FROM CUSTOMER ENTITY
	THEN
	Step 4: Display View Profile Screen (Angular)
	IF (Read Customer Details == True)
	Display View Profile Screen
	ELSE IF (Read Customer Details == False)
	Display Error Modal "There was an unexpected error when trying to display Customer Profile."

Subsystem:	3. User
Use Case:	3.6 Update Profile
Language:	C#
Pseudo Code:	Step 1: A customer wants to Update Profile.
	THEN
	Step 2: Customer Clicks User Icon (Angular)
	ONCLICK User Icon
	THEN ONCLICK View Profile DropDown Menu Item
	THEN
	Step 3: Invoke Use Case 3.5 View Profile (C#)
	Invoke Use Case 3.5 View Profile
	THEN
	Step 4: Customer requests to update profile (Angular)
	Customer requests to update profile.
	ONCLICK Update Button
	THEN
	Step 5: Display Update Profile Screen (Angular)
	Display Update Profile Screen
	THEN
	Step 6: Customer provides updated details. (Angular)
	Customer updates details.
	ONCLICK Update Button
	THEN
	Step 7: Validate Update Details (C#)
	If (Input fields Valid == True)
	Continue to step 8
	Else If (Input fields Valid == false)
	Display Validation Error Modal "The information entered is in the incorrect format. Please try again.
	THEN

Step 8: Display Confirm Update Modal (Angular)

Display Confirm Update Modal "Are you sure you want to update Profile?"

THEN

Step 9: Confirm Update (Angular)

Customer Confirms Update

ONCLICK Confirm Button

[ALT] ONCLICK Cancel Button → Terminate Use Case

THEN

Step 10: Update Customer Details (C#)

SQL UPDATE

Title, Name, Surname, Cell Phone Number,

Email Address, VAT Registration Number,

Business Name

IN CUSTOMER ENTITY

THEN

Step 11: Display Success Modal (Angular)

IF (Update Successful == True)

Display Success Modal "Customer Profile was successfully updated."

ELSE IF (Update Successful == False)

Display Error Modal "There was an unexpected error when trying to update the customer Profile."

Subsystem:	3. User
Use Case:	3.7 Delete Profile
Language:	C#
Pseudo Code:	Step 1: A customer wants to Delete Profile.
	THEN
	Step 2: Customer Clicks User Icon (Angular)
	ONCLICK User Icon
	THEN ONCLICK View Profile DropDown Menu Item
	THEN
	Step 3: Invoke Use Case 3.5 View Profile (C#)
	Invoke Use Case 3.5 View Profile
	THEN
	Step 4: Customer requests to delete profile (Angular)
	Customer requests to delete profile.
	ONCLICK Delete Button
	THEN
	Step 5: Display Confirm Update Modal (Angular)
	Display Confirm Deletion Modal "Are you sure you want to delete your account?
	THEN
	Step 6: Confirm Deletion (Angular)
	Customer Confirms Update
	ONCLICK Confirm Button
	[ALT] ONCLICK Cancel Button → Terminate Use Case
	THEN
	Step 7: Update Customer Details (C#)
	SQL DELETE
	Title, Name, Surname, Cell Phone Number,
	Email Address, VAT Registration Number,
	Business Name

FROM CUSTOMER ENTITY

SQL DELETE

Username, Password

FROM USER ENTITY

THEN

Step 8: Display Success Modal (Angular)

IF (Deletion Successful == True)

Display Success Modal "Customer Profile was successfully deleted.."

ELSE IF (Deletion Successful == False)

Display Error Modal "There was an unexpected error when trying to delete the customer account."

Subsystem:

3. User

Use Case:	3.8 Add User Role
Language:	C#
Pseudo Code:	Step 1: An Admin wants to add a User Role
	THEN
	Step 2: Admin Clicks User Icon (Angular)
	ONCLICK User Icon
	THEN ONCLICK Add User Role DropDown Menu Item
	THEN
	Step 3: Display Add User Role Screen (Angular)
	Display Add User Role Screen
	THEN
	Step 4: Provide User Role Details (Angular)
	Admin provides use role details.
	ONCLICK Add Button
	THEN
	Step 5: Validate User Role Details (C#)
	If (Input fields Valid == True)
	Continue to step 8
	Else If (Input fields Valid == false)
	Display Validation Error Modal "The information entered is in the incorrect format. Please try again.
	THEN
	Step 6: Display Confirmation Modal (Angular)
	Display Confirmation Modal "Are the User Role details correct?"
	THEN
	Step 7: Confirm User Role Details (Angular)
	Admin Confirms Details
	ONCLICK Confirm Button
	[ALT] ONCLICK Cancel Button → Terminate Use Case
	THEN

Step 8: Save User Role Details and generate UserRoleID (C#)

SQL INSERT

User Role Name, User Role Description

IN USERROLE ENTITY

THEN

Step 9: Display Success Modal (Angular)

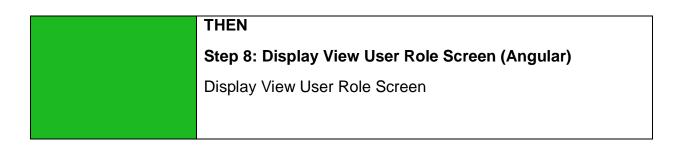
IF (Add User Role Successful == True)

Display Success Modal "The User Role was successfully added.."

ELSE IF (Add User Role Successful == False)

Display Error Modal "There was an unexpected error when trying to add the User Role.

Use Case:	3.9 Search User Role
Language:	C#
Pseudo Code:	Step 1: An Admin wants to Search for a User Role
	THEN
	Step 2: Admin Clicks User Icon (Angular)
	ONCLICK User Icon
	THEN ONCLICK Search User Role DropDown Menu Item
	THEN
	Step 3: Display Search User Role Screen (Angular)
	Display Search User Role Screen
	THEN
	Step 4: Provide Search criteria (Angular)
	Admin Provides Search Criteria
	ONCLICK Search Button
	THEN
	Step 5: Read Matching Records (C#)
	SQL READ
	Matching Records
	FROM CUSTOMER ENTITY
	THEN
	Step 6: Display matching records (Angular)
	IF (Matching Records Found == True)
	Display Search User Role Table
	ELSE IF (Matching Records Found == False)
	Display Error Modal" No User Role's matching the search criteria was found."
	THEN
	Step 7: Admin Selects User Role to View (Angular)
	Select User Role to View
	ONCLICK View Button



Subsystem:

3. User

Use Case:	3.10 Update User Role
Language:	C#
Pseudo Code:	Step 1: An Admin wants to Update a User Role
	THEN
	Step 2: Admin Clicks User Icon (Angular)
	ONCLICK User Icon
	THEN ONCLICK Update User Role DropDown Menu Item
	THEN
	Step 3: Invoke Use Case 3.9 Search User Role (C#)
	Invoke Use Case 3.9 Search User Role
	THEN
	Step 4: Admin requests to update User Role
	Admin requests to update the User Role
	ONCLICK Update Button
	THEN
	Step 5: Display Update User Role Screen (Angular)
	Display Update User Role Screen
	THEN
	Step 6: Admin provides updated details. (Angular)
	Admin updates details.
	ONCLICK Update Button
	THEN
	Step 7: Validate Update Details (C#)
	If (Input fields Valid == True)
	Continue to step 8
	Else If (Input fields Valid == false)
	Display Validation Error Modal "The information entered is in the incorrect format. Please try again.
	THEN
	Step 8: Display Confirm Update Modal (Angular)

Display Confirm Update Modal "Are you sure you want to update the User Role details.?"

THEN

Step 9: Confirm Update (Angular)

Admin Confirms Update

ONCLICK Confirm Button

[ALT] ONCLICK Cancel Button → Terminate Use Case

THEN

Step 10: Update User Role Details (C#)

SQL UPDATE

User Role Name, User Role Description

IN USERROLE ENTITY

THEN

Step 11: Display Success Modal (Angular)

IF (Update Successful == True)

Display Success Modal "The User Role was successfully Updated."

ELSE IF (Update Successful == False)

Display Error Modal "There was an unexpected error when trying to update the User Role."

Use Case:	3.11 Delete User Role
Language:	C#
Pseudo Code:	Step 1: An Admin wants to Delete a User Role
	THEN
	Step 2: Admin Clicks User Icon (Angular)
	ONCLICK User Icon
	THEN ONCLICK Delete User Role DropDown Menu Item
	THEN
	Step 3: Invoke Use Case 3.9 Search User Role (C#)
	Invoke Use Case 3.9 Search User Role
	THEN
	Step 4: Admin Requests to Delete User Role (Angular)
	Admin requests to delete the user role.
	ONCLICK Delete Button
	THEN
	Step 5: Display Confirmation Modal (Angular)
	Display Confirmation Modal "Are you sure you want to delete the User Role?"
	THEN
	Step 6: Confirm Deletion (Angular)
	Admin Confirms Deletion
	ONCLICK Confirm Button
	[ALT] ONCLICK Cancel Button → Terminate Use Case
	THEN
	Step 7: Delete User Role Details (C#)
	SQL DELETE
	User Role Name, User Role Description
	FROM USERROLE ENTITY
	THEN
	Step 8: Display Success Modal (Angular)

IF (Update Successful == True)

Display Success Modal "The User Role was successfully deleted."

ELSE IF (Update Successful == False)

Display Error Modal "There was an unexpected error when trying to delete the User Role."

3.1.3. EMPLOYEE SUBSYSTEM

4. Subsystem:	4. Employee Delivery Shifts
Use Case:	4.1. Create Delivery Shift
Language:	C#
Pseudo Code:	Step 1: An Admin wants to add a new delivery shift
	THEN
	Step 2: Admin Clicks Create Delivery Shift menu item (Angular)
	ONCLICK Delivery Shift
	THEN ONCLICK Create Delivery Shift Drop Down Menu Item
	THEN
	Step 3: Display Create Delivery Shift Screen (Angular)
	Display Create Delivery Shift Screen
	THEN
	Step 4: Provide details for new Delivery Shift (Angular)
	Admin provides new delivery shift details.
	THEN
	Step 5: Admin clicks Add button (Angular)
	ONCLICK Add Button
	[ALT] ONCLICK Cancel Button → terminate use case (Angular)
	THEN
	Step 6: Display Confirmation Modal (Angular)
	Display Confirmation Modal "Would you like to proceed with the creation of a new delivery shift?"
	THEN

Step 7: Admin clicks Yes button (Angular)

ONCLICK Yes Button

[ALT] ONCLICK No Button → Display Add Delivery Shift Screen (Angular)

THEN

Step 8: Validate Delivery Shift Details (C#)

If (Input fields Valid == True)

Continue to step 9

Else If (Input fields Valid == false)

[ALT] Display Validation Unsuccessful Modal "The information provided in the date and time already exists in a delivery shift."

THEN

Step 9: Save delivery shift details and generates a Time_ID, Date_ID and Shift_ID (#C)

SQL INSERT

Start_Time, End_Time

IN TIME ENTITY

SQL INSERT

Date

IN THE DATE ENTITY

SQL INSERT

Time_ID, Date_ID

IN SHIFT ENTITY

THEN

Step 10: Display Success Modal (Angular)

IF (Create Delivery Shift Successful == True)

Display Success Modal "Delivery Shift was successfully created."

ELSE IF (Create Delivery Shift Successful == False)

[ALT] Display Unsuccessful Modal "Failed to create delivery shift. Please try again later."

Subsystem:	4. Employee Delivery Shifts
Use Case:	4.2. Search Delivery Shift
Language:	C#
Pseudo Code:	Step 1: An Admin wants to search for a delivery shift
	THEN
	Step 2: Admin Clicks Search Delivery Shift
	ONCLICK Delivery Shift
	THEN ONCLICK Search Delivery Shift Drop Down Menu Item
	THEN
	Step 3: Display Search Delivery Shift Screen (Angular)
	Display Search Delivery Shift Screen
	THEN
	Display Search Delivery Shift Screen

Step 4: Input Search criteria (Angular)

Admin inputs Search Criteria

ONCLICK Search Icon

THEN

Step 5: Validate Delivery Shift Details (C#)

If (Input fields Valid == True)

Continue to step 6

Else If (Input fields Valid == false)

[ALT] Display Validation Unsuccessful Modal "No matches found. Please check the format of the entered information."

THEN

Step 6: Read Matching Records (C#)

SQL READ

Matching Records

FROM SHIFT ENTITY

SQL READ

Matching Records

FROM DATE ENTITY

SQL READ

Matching Records

FROM TIME ENTITY

SQL READ

Matching Records

FROM EMPLOYEESHIFT ENTITY
SQL READ
Matching Records
FROM EMPLOYEE ENTITY
THEN
Step 7: Display matching records (Angular)
IF (Matching Records Found == True)
Display Delivery Shift Table on Search Delivery Shift Screen
ELSE IF (Matching Records Found == False)
→ terminate use case (Angular)

Subsystem:	4. Employee Delivery Shifts
Use Case:	4.3. Update Delivery Shift
Language:	C#
Pseudo Code:	Step 1: An Admin wants to update delivery shift information
	ONCLICK Delivery Shift
	THEN ONCLICK Search Delivery Shift Dropdown Menu Item
	THEN
	Step 2: Invoke Use Case 4.2. Search Delivery Shift (C#)

Invoke Use Case 4.2. Search Delivery Shift

Then ONCLICK Update Button in Option Column

THEN

Step 3: Display Update Delivery Shift Screen (Angular)

Display Update Delivery Shift Screen

THEN

Step 4: Admin provides new information. (Angular)

Admin edits textboxes to update information.

ONCLICK Done Button

[ALT] ONCLICK Cancel Button → terminate use case (Angular)

THEN

Step 5: Validate Update Details (C#)

If (Input fields Valid == True)

Continue to step 6

Else If (Input fields Valid == false)

Display Validation Unsuccessful Modal "The information provided in the date and time already exists in a delivery shift."

THEN

Step 6: Update Delivery Shift Details (C#)

SQL UPDATE

Start_Time, End_Time, Date

IN SHIFT ENTITY

THEN

Step 7: Display Success Modal (Angular)

IF (Update Successful == True)

Display Success Modal "Delivery Shift information was successfully updated."

ELSE IF (Update Successful == False)

Display Unsuccessful Modal "Failed to update delivery shift information. Please try again later."

Use Case:	4.4. Delete Delivery Shift
Language:	C#
Pseudo Code:	Step 1: An Admin wants to delete delivery shift information
	ONCLICK Delivery Shift
	THEN ONCLICK Search Delivery Shift Dropdown Menu Item
	THEN
	Step 2: Invoke Use Case 4.2. Search Delivery Shift (C#)
	Invoke Use Case 4.2. Search Delivery Shift
	THEN
	Step 3: Admin Clicks Delete Button
	Then ONCLICK Delete Button in Option Column
	THEN
	Charle Carffords Daniel Andria (Ann. In.)
	Step 4: Display Confirm to Proceed Modal (Angular)
	Display Confirm Proceed Modal "Would you like to proceed with the deletion of the selected delivery shift?"
	THEN
	Step 5: Admin clicks Yes button (Angular)
	ONCLICK Yes Button
	[ALT] ONCLICK No Button → Display Search Delivery Shift Screen (Angular)
	THEN
	Step 6: Delete Delivery Shift
	SQL DELETE
	Shift_ID
	IN SHIFT ENTITY

THEN

Step 7: Remove Delivery Shift Information line on screen

Remove Delivery Shift Table on Search Delivery Shift Screen

THEN

Step 8: Display Success Modal (Angular)

IF (Delete Successful == True)

Display Success Modal "Delivery Shift information was successfully deleted."

ELSE IF (Delete Successful == False)

Display Unsuccessful Modal "Failed to delete delivery shift. Please try again later."

Use Case: 4.5. Add Employee Language: C# Pseudo Code: Step 1: The admin wants to add the employee. THEN	
Pseudo Step 1: The admin wants to add the employee. Code:	
Code:	
Code:	
IIILIN	
Step 2: Admin clicks on Employee menu item(Angular)	
ONCLICK Employee Menu item on navigation bar	
THEN	
Step 3: Display Employee Dropdown menu item (Angular)	
Display Employee dropdown menu item	
THEN	
Step 4: Admin clicks Employee dropdown menu item (Angular)	
ONCLICK Add Employee dropdown menu item	
THEN	
Step 5: Display Add Employee Screen (Angular)	
Display Add Employee Screen	
THEN	
//Step 6: Employee Communicates details for the admin to enter into the system.	
by brenn.	
Step 7: Provide Employee Details (Angular)	
Provide Employee Details	
ONCLICK Next Button	
THEN	
Step 8: Validate Details (C#)	

```
If (Input fields == Valid)
       Continue to step 9
Else If (Input fields == false)
       Display Validation Unsuccessful Modal "The information provided in
       the given parameters is not in the correct format."
THEN
Step 9: Enable and disable controls on Add Employee Screen (Angular)
       Title= combobox. (enable==false)
       Name= textbox. (enable==false)
       Surname= textbox. (enable==false)
       CellNumber= textbox. (enable==false)
       Address= textbox. (enable==false)
       DOB= calendar. (enable==false)
       IDNumber= textbox. (enable==false)
       Next=button. (enable==false)
       UserRole = combobox. (enable==true)
       Username = textbox. (enable==true)
       Password = textbox. (enable==true)
       Confirm Password = textbox. (enable==true)
       Add Employee = button. (enable==true)
THEN
Step 10: Provide User Details (Angular)
Provide User Details
ONCLICK Add Employee Button
THEN
Step 11: Validate Details (C#)
```

If (Input fields Valid == True)

Continue to step 12

Else If (Input fields Valid == false)

Display Validation Unsuccessful Modal "The information provided in the given parameters is not in the correct format."

THEN

Step 12: Display Confirm to Proceed Modal (Angular)

Display Confirm Proceed Modal "Would you like to proceed with the addition of a new employee?"

THEN

Step 13: Admin clicks Yes button (Angular)

ONCLICK Yes Button

[ALT] ONCLICK No Button → Display Add Employee Screen (Angular)

THEN

Step 14: Save Employee Details and Generate EmployeeID and UserID (C#)

SQL INSERT

Title, Employee_Name, Employee_Surname, Employee_Number, Employee_Address, Employee_DOB, Employee_IDNumber

IN EMPLOYEE ENTITY

SQL INSERT

User_Username, User_Password, UserRole_ID

IN USER ENTITY

THEN

Step 15: Display Success Modal (Angular)

IF (Add Employee Successful == True)

Display Success Modal "Employee information was successfully created."

ELSE IF (Add Employee Successful == False)

Display Unsuccessful Modal "Failed to create employee information. Please try again later."

Subsystem:	4. Employee Delivery Shifts
Use Case:	4.6. Search Employee
Language:	C#
Pseudo Code:	Step 1: An Admin wants to search for an employee
	THEN
	Step 2: Admin Clicks Search Employee
	ONCLICK Employee
	THEN ONCLICK Search Employee Drop Down Menu Item
	THEN
	Step 3: Display Search Employee Screen (Angular)
	Display Search Employee Screen
	THEN
	Step 4: Input Search criteria (Angular)
	Admin inputs Search Criteria
	ONCLICK Search Button
	THEN
	Step 5: Validate Delivery Shift Details (C#)
	If (Input fields Valid == True)
	Continue to step 6
	Else If (Input fields Valid == false)
	[ALT] Display Validation Unsuccessful Modal "No matches found. Please check the format of the entered information."
	THEN

Step 6: Read Matching Records (C#)

SQL READ

Matching Records

FROM EMPLOYEE ENTITY

THEN

Step 7: Display matching records (Angular)

IF (Matching Records Found == True)

Display Employee Table on Search Employee Screen

ELSE IF (Matching Records Found == False)

→ terminate use case (Angular)

Subsystem:	4. Employee Delivery Shifts
Use Case:	4.7. Update Employee
Language:	C#
Pseudo Code:	Step 1: An Admin wants to update employee information
	ONCLICK Employee
	THEN ONCLICK Search Employee Dropdown Menu Item
	THEN
	Step 2: Invoke Use Case 4.6. Search Employee (C#)
	Invoke Use Case 4.2. Search Employee
	Then ONCLICK Update Button in Option Column
	THEN
	Step 3: Display Update Employee Screen (Angular)
	Display Update Employee Screen

THEN

Step 4: Admin provides new information. (Angular)

Admin edits textboxes to update information.

ONCLICK Done Button

[ALT] ONCLICK Cancel Button → terminate use case (Angular)

THEN

Step 5: Validate Update Details (C#)

If (Input fields Valid == True)

Continue to step 6

Else If (Input fields Valid == false)

Display Validation Unsuccessful Modal "The information provided in the given parameters is not in the correct format."

THEN

Step 6: Update Employee Details (C#)

SQL UPDATE

Title, Employee_Name, Employee_Surname, Employee_Number, Employee_Address, Employee_DOB, Employee_IDNumber

IN EMPLOYEE ENTITY

THEN

Step 7: Display Success Modal (Angular)

IF (Update Successful == True)

Display Success Modal "Employee information was successfully updated."

ELSE IF (Update Successful == False)

Display Unsuccessful Modal "Failed to update employee information. Please try again later."

Subsystem:	4. Employee Delivery Shifts
Use Case:	4.8. Delete Employee
Language:	C#
Pseudo Code:	Step 1: An Admin wants to delete employee information
	ONCLICK Employee
	THEN ONCLICK Search Employee Dropdown Menu Item
	THEN
	Step 2: Invoke Use Case 4.6. Search Employee (C#)
	Invoke Use Case 4.6. Search Employee
	THEN
	Step 3: Admin Clicks Delete Button
	Then ONCLICK Delete Button in Option Column
	THEN
	Ston 4. Display Confirm to Drosped Model (Angular)
	Step 4: Display Confirm to Proceed Modal (Angular) Display Confirm Proceed Modal "Would you like to proceed with
	the deletion of the selected employee?"
	THEN
	Step 5: Admin clicks Yes button (Angular)
	ONCLICK Yes Button
	[ALT] ONCLICK No Button → Display Search Employee Screen (Angular)
	THEN
	Step 6: Delete Employee
	SQL DELETE
	Employee_ID

IN EMPLOYEE ENTITY
THEN
Step 7: Remove Employee Information line on screen
Remove Employee Table on Search Employee Screen
THEN
Step 8: Display Success Modal (Angular)
IF (Delete Successful == True)
Display Success Modal "Employee information was successfully deleted."
ELSE IF (Delete Successful == False)
Display Unsuccessful Modal "Failed to delete employee. Please try again later."

Subsystem:	4. Employee Delivery Shifts
Use Case:	4.10. Assign Delivery Shift
Language:	C#
Pseudo Code:	Step 1: An Admin wants to assign a delivery shift to an employee.
	ONCLICK Delivery Shift
	THEN ONCLICK Search Delivery Shift Dropdown Menu Item
	THEN
	Step 2: Invoke Use Case 4.2. Search Delivery Shift (C#)
	Invoke Use Case 4.2. Search Delivery Shift
	THEN
	Step 3: Admin selects Assign button
	ONCLICK Assign button in Assign To Column

THEN

Step 4: Display Assign Delivery Shift Modal (Angular)

Display Assign Delivery Shift Modal

THEN

Step 5: Admin selects the employee name. (Angular)

Admin selects the employee name in dropdown.

THEN

Step 6: Admin clicks on the Proceed button

ONCLICK Proceed Button

[ALT] ONCLICK Cancel Button → Display Search Delivery Screen (Angular)

THEN

Step 7: Display Confirm to Proceed Modal (Angular)

Display Confirm Proceed Modal "Would you like to proceed with assigning this employee to the selected delivery shift?"

THEN

Step 8: Admin clicks Yes button (Angular)

ONCLICK Yes Button

[ALT] ONCLICK No Button → Display Search Delivery Shift Screen (Angular)

THEN

Step 9: Remove Delivery Shift Details (C#)

SQL DELETE

EmployeeShift_ID, Employee_ID

IN EMPLOYEESHIFT ENTITY
THEN
Step 10: Display Success Modal (Angular)
IF (Removal Successful == True)
Display Success Modal "Delivery shift was successfully assigned."
ELSE IF (Remove Successful == False)
Display Unsuccessful Modal "Failed to assign delivery shift to employee. Please try again later"

Subsystem:	4. Employee Delivery Shifts
Use Case:	4.11. Cancel Delivery Shift
Language:	C#
Pseudo Code:	Step 1: An Admin wants to cancel an assigned delivery shift
	ONCLICK Delivery Shift
	THEN ONCLICK Search Delivery Shift Dropdown Menu Item
	THEN
	Step 2: Invoke Use Case 4.2. Search Delivery Shift (C#)
	Invoke Use Case 4.2. Search Delivery Shift
	THEN
	Step 3: Admin Clicks Cancel Button
	Then ONCLICK Cancel Button in Option Column
	THEN
	Step 4: Display Confirm to Proceed Modal (Angular)
	Display Confirm Proceed Modal "Would you like to proceed with the cancellation of the selected assigned delivery shift?"

THEN

Step 5: Admin clicks Yes button (Angular)

ONCLICK Yes Button

[ALT] ONCLICK No Button → Display Search Delivery Shift Screen (Angular)

THEN

Step 6: Delete Employee shift information

SQL DELETE

Employee_ID

IN EMPLOYEESHIFT ENTITY

THEN

Step 7: Display Success Modal (Angular)

IF (Delete Successful == True)

Display Success Modal "Cancellation of the assigned shift was successful."

ELSE IF (Delete Successful == False)

Display Cannot Be Cancelled Modal "This delivery shift cannot be cancelled as it is not yet assigned to an employee. Please try again later."

THEN

Step 8: Remove Employee Information line on screen

Remove Employee Name from Delivery Shift Table on Search Delivery Shift Screen

THEN

4.1.1. ADMIN SUBSYSTEM

5. Subsystem:	5. Administration Back Office
Use Case:	5.1 Search Online Sales
Language:	C#
Pseudo Code:	Step 1: Employee/Admin wants to Search an Online Sale
	THEN
	Step 2: Employee/Admin selects Administration Back Office Menu Item (Angular)
	ONCLICK Administration Back Office Menu Item
	THEN ONCLICK Search Online Sales dropdown menu item
	THEN
	Step 3: Display Search Online Sales Screen (Angular)
	Display Search Online Sales Screen
	THEN
	Step 4: Admin provides search criteria (Angular)
	Admin Selects filter option.
	ONCLICK Search Button
	THEN
	Step 5: Read Matching Record (C#)
	SQL READ
	Matching Records
	FROM SALE ENTITY
	THEN
	Step 6:
	IF (Matching Records Found == True)
	Display Search Online Sales Table
	ELSE IF (Matching Records Found == False)
	Display Error Modal "No sales matching the search criteria was found."
	THEN
	Step 7: Admin selects Sale to View

Admin selects sale.

ONCLICK View Button

THEN

Step 8: Display View Sale Screen

IF (Read Sale Details Successfully == True)

Display View Sale Screen

ELSE IF (Read Sale Details Successfully == False)

Display Error Modal "There was an unexpected error when trying to display the sales."

Use Case:	5.2 Pack Order
Language:	C#
Pseudo Code:	Step 1: Employee wants to Pack an Order
	THEN
	Step 2: Employee selects Administration Back Office Menu Item (Angular)
	ONCLICK Administration Back Office Menu Item
	THEN ONCLICK Pack Order dropdown menu item
	THEN
	Step 3: Invoke Use Case 5.1 Search Sale Order (C#)
	Invoke Use Case 5.1 Search Sale Order
	Step 4: Employee requests to pack Order (Angular)
	Employee requests to pack an order
	ONCLICK Pack Order Button
	Step 5: Display Pack Order Screen (Angular)
	Display Search Online Sales Screen
	THEN
	Step 6: Employee Provides Pack Order Details (Angular)
	Employee Provides Pack Order Details
	ONCLICK Confirm Button
	THEN
	Step 7: Update Order Status(C#)
	IF (Order For Collection Checkbox == Checked)
	SQL UPDATE
	Order Status == "Ready for Collection"
	IN SALE ENTITY
	[ALT]
	ELSE IF (Order For Delivery Checkbox == Checked)
	SQL UPDATE

Order Status == "Ready for Delivery"

IN SALE ENTITY

Skip to Step 9.

THEN

Step 8: Notify Customer that Order is Ready for Collection (C#)

Notify customer via SMS that Order is Ready for Delivery. SQL READ

Customer CellPhone Number

From Customer Entity

THEN

Step 9: Display Success Modal

IF (Pack Order Successful == True)

Display Success Modal "Order has been

Packed."

ELSE IF (Pack Order Successful == False)

Display Error Modal "There was an unexpected error when trying to complete the pack order process.

Subsystem:	5. Administration Back Office
Use Case:	5.3 Do Stock Take
Language:	C#
Pseudo Code:	Step 1: Admin wants to do Stock Take
	THEN
	Step 2: Admin selects Administration Back Office Menu Item (Angular)
	ONCLICK Administration Back Office Menu Item
	THEN ONCLICK Do Stock Take dropdown menu item
	THEN
	Step 3: Invoke Use Case 9.10 Search Product (C#)
	Invoke Use Case 9.10 Search Product
	Step 4: Admin requests to do stock take (Angular)
	Employee requests to do stock take
	ONCLICK Do Stock-Take Button
	Step 5: Display Do Stock-Take Screen (Angular)
	Display Do Stock-Take Screen
	THEN
	Step 6: Admin Provides Stock Take Details (Angular)
	Admin Provides Stock Take Details
	ONCLICK Save Button
	THEN
	Step 7: Validate Stock Take Details (C#)
	Validate Details (C#)
	If (Input fields Valid == True)
	Continue to step 8
	Else If (Input fields Valid == false)
	Display Validation Error Modal "The information entered is in the incorrect format. Please try again."
	THEN
	Step 8: Display Confirmation Modal (Angular)

Display Confirmation Modal "Do you want to proceed with the stock take?"

THEN

Step 9: Confirm Stock Take (Angular)

Admin Confirms to proceed

ONCLICK Confirm Button

[ALT] ONCLICK Cancel Button → Terminate Use Case

THEN

Step 10: Save Stock Take Details and Generate Stock Take ID (C#)

SQL INSERT

Stock Take Date

IN STOCKTAKE ENTITY

SQL INSERT

Stock Take Quantity

IN PRODUCTITEM_STOCKTAKE ENTITY

THEN

Step 11: Update Quantity on Hand (C#)

SQL UPDATE

Quantity on Hand

IN PRODUCTITEM ENTITY

THEN

Step 12: Display Success Modal (Angular)

IF (StockTake Successful == True)

Display Success Modal "Stock Take was completed successfully."

ELSE IF (StockTake Successful == False)

Display Error Modal "There was an unexpected error when trying to do the stock-take."

Subsystem:	5. Administration Back Office
Use Case:	5.4 View Delivery Shift Schedule
Language:	C#
Pseudo Code:	Step 1: Employee wants to View Delivery Shift Schedule
	THEN
	Step 2: Employee selects Administration Back Office Menu Item (Angular)
	ONCLICK Administration Back Office Menu Item
	THEN ONCLICK View Delivery Shift dropdown menu item
	THEN
	Step 3: Read Delivery Shift Schedule Details (C#)
	SQL Read
	Shift Date, Shift Start Time, Shift End Time
	FROM SHIFT ENTITY
	SQL READ
	Employee Name
	FROM EMPLOYEE_SHIFT ENTITY
	THEN
	Step 4: Display Delivery Shift Schedule (Angular)
	IF (Read Delivery Shift Details Successfully == True)
	Display Delivery Shift Schedule Screen
	ELSE IF (Read Delivery Shift Details Successfully == False)
	Display Error Modal "There was an unexpected error when trying to display the Delivery Shift Schedule."

Subsystem:	5. Administration Back Office
Use Case:	5.5 Write-Off Stock
Language:	C#
Pseudo Code:	Step 1: Admin wants to Write-Off Stock
	THEN
	Step 2: Admin selects Administration Back Office Menu Item (Angular)
	ONCLICK Administration Back Office Menu Item
	THEN ONCLICK Write-Off Stock dropdown menu item
	THEN
	Step 3: Invoke Use Case 9.10 Search Product (C#)
	Invoke Use Case 9.10 Search Product
	Step 4: Admin requests to do stock take (Angular)
	Admin requests to Write-Off Stock
	ONCLICK Write-Off Stock Button
	Step 5: Display Do Write-Off Stock Screen (Angular)
	Display Do Stock-Take Screen
	THEN
	Step 6: Admin Provides Stock Take Details (Angular)
	Admin Provides Stock Take Details
	ONCLICK Save Button
	THEN
	Step 7: Validate Write-Off Stock Details (C#)
	Validate Details (C#)
	If (Input fields Valid == True)
	Continue to step 8
	Else If (Input fields Valid == false)
	Display Validation Error Modal "The information entered is in the incorrect format. Please try again."
	THEN
	Step 8: Display Confirmation Modal (Angular)

Display Confirmation Modal "Do you want to proceed with the Write-off?

THEN

Step 9: Confirm Write-off Stock (Angular)

Admin Confirms to proceed

ONCLICK Confirm Button

[ALT] ONCLICK Cancel Button → Terminate Use Case

THEN

Step 10: Save Write-off Stock Details and Generate WriteOffID (C#)

SQL INSERT

Write-Off Date

IN WRITEOFF ENTITY

SQL INSERT

Write-Off Quantity, Write-off Reason

IN PRODUCTITEM_WRITEOFFSTCOK ENTITY

THEN

Step 11: Update Quantity on Hand (C#)

SQL UPDATE

Quantity on Hand

IN PRODUCTITEM ENTITY

THEN

Step 12: Display Success Modal (Angular)

IF (StockTake Successful == True)

Display Success Modal "Write-Off was completed successfully."

ELSE IF (StockTake Successful == False)

Display Error Modal "There was an unexpected error when trying to do the write-off,

5.1.1. CUSTOMER SUBSYSTEM

Subsystem:	6. Customer
Use Case:	6.1 View Products
Language:	C#
Pseudo Code:	Step 1: Customer request to browse through the products by category.
	THEN
	Step 2: Customer clicks Products Menu Item (Angular)
	ON CLICK Products Menu Item
	THEN ON CLICK Products Dropdown menu Item
	THEN
	Step 3: Load Products Screen (Angular)
	SQL Select
	ProductItem information
	From ProductItem table
	Where ProductCategory matches with ProductCategory in ProductCategory table
	Retrieve the following: o ProductCategory_ID o ProductCategory_Desc o ProductItem_Name o ProductItem_Image
	THEN display the information for each product in that category in a card. Display Products Screen
	[ALT STEP 3]: The system fails to load delivery page(Angular) ON CLICK Product Menu Item
	THEN ON CLICK Products Dropdown menu Item
	System fails to load Products page.
	THEN display Error message, "There was an error loading the Products page. Please try again later."
	Terminate

Subsystem:	6. Customer
Use Case:	6.2 Search Product Details
Language:	C#
Pseudo Code:	Step 1: Customer request to search for a product.
	The system invokes 6.1 Search Products
	Step 2: Customer provides search criteria (Angular)
	ON CLICK Search Menu bar
	THEN
	Customer enters search criteria
	THEN
	The customer provides Product information(Angular)
	Step 3: Validate and Capture search details (Angular)
	ON CLICK Search icon-button
	THEN
	If(input not valid== true)
	Display error message, "Please ensure that the details you have entered are valid"
	else
	SQL Read
	ProductItem information
	From ProductItem table
	Where ProductItem matches with ProductItem in ProductItem table
	Read the following: o ProductItem_ID o ProductItem_Name THEN
	Step 3: Display retrieved results Screen (Angular)
	THEN display the information for each product in that category in a card. Display Products Screen with retrieved results
	Customer selects product they were looking for.

THEN

Step 4: Display retrieved results Screen (Angular)

ON CLICK product card

SQL Select

ProductItem information

From ProductItem table

SQL Select

CategoryType information

From CategoryType table

SQL Select

ProductCategory information

From ProductCategory table

Where ProductItem matches with ProductItem in ProductItem table

Read the following:

- CategoryType_Desc
- ProductCategory_Desc
- ProductItem_Name
- ProductItem_Desc
- o ProductItem_Image
- o ProductItem_Cost

THEN

Display Product Details Screen with retrieved results

[ALT STEP 4]: The system fails to load Product Details page(Angular)

ON CLICK Product Card

System fails to load Product Details page.

THEN display Error message, "There was an error loading the Product Details page. Please try again later."

Subsystem:	6. Customer
Use Case:	6.3 Search Product Details
Language:	C#
Pseudo Code:	Step 1: Customer request to search for a product.
	The customer request to browse through the specials and promotions.
	Step 2: Customer clicks Promotion Menu Item (Angular)
	ON CLICK Promotions Menu item
	THEN
	Step 3: Load Promotions Screen (Angular)
	SQL Select
	ProductItem information
	From ProductItem table
	SQL Select
	Product_Special information
	From Product_Special table
	SQL Select
	Special information
	From Special table
	Retrieve the following:
	THEN display the information for each product in that category in a card. Display Promotions Screen
	Step 4: Customer provides search criteria (Angular)
	ON CLICK Search Menu bar

THEN

Customer enters search criteria

THEN

The customer provides ProductItem information(Angular)

Step 5: Validate and Capture search details (Angular)

ON CLICK Search icon-button

THEN

If(input not valid== true)

Display error message, "Please ensure that the details you have entered are valid"

else

SQL Read

ProductItem information

From ProductItem table

Where ProductItem matches with ProductItem in ProductItem table

Read the following:

- ProductItem_ID
- o ProductItem_Name

THEN

Step 6: Display retrieved results Screen (Angular)

THEN display the information for each product in that category in a card.

Display Products Screen with retrieved results

Customer selects product they were looking for.

THEN

Step 7: Display retrieved results Screen (Angular)

ON CLICK Promotions card

SQL Select

ProductItem information

From ProductItem table

SQL Select

CategoryType information

From CategoryType table

SQL Select

ProductCategory information

From ProductCategory table

Where ProductItem matches with ProductItem in ProductItem table

Read the following:

- CategoryType_Desc
- ProductCategory_Desc
- o ProductItem_Name
- o ProductItem Desc
- ProductItem_Image
- Special_Price
- Special_EndDate

THEN

Display SpecialPromotion Details Screen with retrieved results

[ALT STEP 3]: The system fails to load delivery page(Angular)

ON CLICK Promotions Menu Item

THEN ON CLICK Promotions Dropdown menu Item

System fails to load Promotions page.

THEN display Error message, "There was an error loading the Promotions page. Please try again later."

Terminate

[ALT STEP 6]: The system fails to load Product Details page(Angular)

ON CLICK Promotion Card

System fails to load SpecialPromotion Details page.

THEN display Error message, "There was an error loading the Product Details page. Please try again later."

Subsystem:	6. Customer
Use Case:	6.4 Add to cart
Language:	C#
Pseudo Code:	Step 1: Customer request to add product item to cart.
	The customer requests to add a product item to cart.
	Step 2: Customer clicks add to cart button (Angular)
	ON CLICK 'Add to cart' button
	THEN
	Step 3: Load Confirm Add to cart prompt (Angular)
	ON CLICK Add button
	THEN
	Step 4: Capture ProductItem details (Angular)
	SQL Select
	CartLine_information
	From CartLine_table
	SQL Select
	ProductItem information
	From ProductItem _table
	Add the following: ○ CartLine_Quantity
	ProductItem_ID
	THEN Step 5: Update cart icon using added details (Angular)
	Update the cart icon in the page header. Using the CartLine_Quantity attribute in the CartLine entity
	[ALT Step 3:] The user clicks the cancel button. Terminated
	[ALT STEP 4]: The system fails add product items to cart (Angular) ON CLICK Add button
	System fails add product items to cart.
	THEN Display Error message, "There was an error while added the product to cart.

Please try again.		,	,	
-------------------	--	---	---	--

Subsystem:	6. Customer
Use Case:	6.5 Search Cart Item
Language:	C#
Pseudo Code:	Step 1: Customer request to see the cart.
	THEN
	Step 2: Customer clicks Cart icon on the menu- strip(Angular)
	ON CLICK Products Menu Item
	THEN
	Step 3: Load Cart Screen (Angular)
	SQL Select
	ProductItem information
	From ProductItem table
	SQL Select
	Product_Special information
	From Product_Special table
	SQL Select
	Special information
	From Special table
	Where ProductCategory matches with ProductCategory in ProductCategory table
	Retrieve the following: o ProductItem_Name o ProductItem_Image o Special_Price or ProductItem_Cost o Special_EndDate (if Special_Price is shown)
	THEN display the information for each product in a tablerow of the CartTable. Display Cart Screen
	[ALT STEP 3]: The system fails to load Cart page(Angular)

ON CLICK Cart icon-button

THEN

System fails to load Cart page.

THEN Display Error message, "There was an error loading the Cart page. Please try again later."

Subsystem:	6. Customer
Use Case:	6.6 Remove Cart Item
Language:	C#
Pseudo Code:	Step 1: Customer request to remove a product item from cart.
	The customer requests to add a product item to cart.
	Step 2: Customer clicks remove icon-button (Angular)
	In the Cart view the user navigates to Cart table
	ON CLICK Remove icon-button of specific cart item
	THEN
	Step 3: Load Confirm Remove the item from cart prompt (Angular)
	ON CLICK Remove button
	THEN
	Step 4: Remove ProductItem details (Angular)
	SQL Delete
	CartLine_information
	From CartLine_table
	SQL Select
	ProductItem information
	From ProductItem table
	Add the following: CartLine_QuantityProductItem_ID THEN
	Step 5: Update cart icon using added details (Angular)
	Update the cart icon in the page header. Using the CartLine_Quantity attribute in the <u>CartLine entity</u>
	[ALT Step 3:] The user clicks the cancel button. Terminated
	[ALT STEP 4]: The system fails remove product items from cart (Angular) ON CLICK Remove button

System fails remove product items from cart.

THEN

Display Error message, "There was an error while removing the product from cart.
Please try again."

Subsystem:	6. Customer
Use Case:	6.7 Checkout
Language:	C#
Pseudo Code:	Step 1: Employee request to go to checkout.
	THEN
	Step 2: Employee clicks the Checkout button (Angular)
	In the Cart View the user navigates to Checkout button
	ON CLICK Checkout button
	THEN
	Step 3: System checks if customer is logged (Angular)
	SQL Read
	AuditTrail information
	From table
	If(audittrail description == valid)
	Continue to next step
	else
	THEN [ALT STEP 3]: Customer is not logged in prompt(Angular) ON CLICK sign in button
	Load login page
	Else if
	ON CLICK register button
	Load register page
	THEN
	System fails to update delivery status.
	THEN display Error message, "There was an error while updating. Please try again later."
	Step 4: Load Checkout- Delivery or Collect prompt (Angular)

The customer selects the delivery optton

ON CLICK Proceed button

THEN

The system loads and displays delivery form.

Step 5: Validate and Update delivery information (Angular)

ON CLICK Proceed button

THEN

If(input incomplete== true)

Display error message, "Please ensure that information has been filled in"

else

SQL Read

Address information

From Address table

SQL Read

Province information

From Province table

SQL Read

City information

From City table

SQL Read

Suburb information

From Suburb table

Validate the following:

- Delivery_ID
- Courier_TrackingNumber

- Delivery_Date
- o Courier_Name
- o Courier Number
- Courier_Email
- o Address_Line1
- o Address Line2
- o Address Line3
- Address_PostalCode
- o Province Description
- City Description
- Suburb_Description

THEN

Step 6: The system captures and updates Delivery and Sale

The system uses a SQL insert to capture the AddressID using the Delivery table and generates a new deliveryID, the system also uses SQL Update to update the SaleOrder_ReceiveType in the Sale Entity

SQL Insert

Delivery information

From Delivery table

SQL Insert

Sale information

From Sale table

Update the following:

- AddressID (Delivery entity)
- DeliveryID (Sale entity)
- SaleOrder_ReceiveType

THEN

Step 7: Load Amount due prompt (Angular)

SQL Select

CartLine information

From CartLine table

Step 8: The user clicks the 'Checkout with PayPal' button (Angular)

ON CLICK VCS

THEN

The system system loads the payment gateway with the amount due.

THEN

The customer makes a payment through the Virtual card services (PayPal) and that system sends us the transaction details.

Step 8: The system uses a SQL insert to record the transaction in the sale table and generates a new SaleID.

SQL Update

Sale information

From Sale table

Update the following:

- SaleID
- Customer ID
- PaymentType ID
- o OrderStatus ID
- SaleOrder Desc
- SaleOrder Date using the current Date.
- o Payment Date using the current Date.
- Payment Amount calculating the total of the cart items.

The system generates a new SaleID by retrieving the last SaleID D in the Sale table and incrementing it by one.

Step 9: The system displays a successful message

THEN

Display message, "Payment Successful"

Terminate

[ALT Step 4]: Customer clicks the collection radio button(Angular)

Proceed to next step.

[ALT Step 5:] The customer selects the cancel button ON Click cancel

[ALT STEP 6]: The system fails update delivery (Angular) ON CLICK No

System fails to update delivery.

THEN display Error message, "There was an error while updating the delivery details.

Please try again."

Terminate

[ALT STEP 8]: An error occurs on the VCS system and the payment failed (Angular)

ON CLICK No

System fails to update delivery.

THEN display Error message, "Payment unsuccessful Please try again."

5.1.2. REPORTS SUBSYSTEM

6. Subsystem:	7. Reports Subsystem
Use Case:	7.1 Generate fast-selling product list
Language:	C#
Pseudo Code:	Step 1: The requests to generate a report
	Then
	Step 2: The admin clicks on the Reports menu option (Angular)
	Onclick the Reports Menu item
	Then
	Step 3: The system displays the "Reports" screen (Angular)
	Display Reports Screen
	Then
	Step 4: The admin selects the "Fast-selling product list" (Angular)
	Onclick Report type dropdown
	Then
	Step 5: The admin selects the required period.
	Select Report start & end date
	Then
	Step 6: The admin clicks the "Generate" button
	Onclick Generate Button
	Then
	Step 7: System retrieves report records from the database (C#)
	SQL Select
	ProductItem_ID, ProductItem_Name, ProductItem_Desc, CategoryType_ID from ProductItem entity,
	SQL Select
	SaleID, SaleOrder_Date, SaleOrderDesc from SALE entity and
	SQL Select

SaleLine_ID, ProductItem_ID, Sale_ID, SaleLine_Quantity from SaleLine Entity
Then
Step 8: The system sorts through the retrieved information and filters for the most purchased products (C#)
Calculate the most purchased product items by checking the SaleLine_Quantity present in each order. With the items being most ordered at the top.
The system also counts how many times the product items appeared in the different orders
Then
Step 9: The system displays the Fast-selling product list (Angular)
Display the fast-selling product list with columns for the quantity of product items ordered and also for the number of times the products were placed in unique orders.

Subsystem:	7. Reports Subsystem
Use Case:	7.2 Generate slow-selling product list
Language:	C#
Pseudo Code:	Step 1: The requests to generate a report
	Then
	Step 2: The admin clicks on the Reports menu option (Angular)
	Onclick the Reports Menu item
	Then
	Step 3: The system displays the "Reports" screen (Angular)
	Display Reports Screen
	Then
	Step 4: The admin selects the "Slow-selling product list" (Angular)
	Onclick Report type dropdown
	Then

Step 5: The admin selects the required period.

Select Report start & end date

Then

Step 6: The admin clicks the "Generate" button

Onclick Generate Button

Then

Step 7: System retrieves report records from the database (C#)

SQL Select

ProductItem_ID, ProductItem_Name, ProductItem_Desc, CategoryType ID from ProductItem entity,

SQL Select

SaleID, SaleOrder_Date, SaleOrderDesc from SALE entity and

SQL Select

SaleLine_ID, ProductItem_ID, Sale_ID, SaleLine_Quantity from SaleLine Entity

Then

Step 8: The system sorts through the retrieved information and filters for the least purchased products (C#)

Calculate the most purchased product items by checking the SaleLine_Quantity present in each order. With the items not purchased often at the top.

The system also counts how many times the product items appeared in the different orders

Then

Step 9: The system displays the Slow-selling product list (Angular)

Display the slow-selling product list with columns for the quantity of product items ordered and for the number of times the products were placed in unique orders.

Subsystem:

Use Case:	7.3 Generate most frequent buyers report
Language:	C#
Pseudo Code:	Step 1: The requests to generate a report
	Then
	Step 2: The admin clicks on the Reports menu option (Angular)
	Onclick the Reports Menu item
	Then
	Step 3: The system displays the "Reports" screen (Angular)
	Display Reports Screen
	Then
	Step 4: The admin selects the "Most frequent buyer report" (Angular)
	Onclick Report type dropdown
	Then
	Step 5: The admin clicks the "Generate" button (Angular)
	Onclick Generate Button
	Then
	Step 6: System retrieves report records from the database (C#)
	SQL Select
	Sale ID, SaleOrder Date, SaleOrder Desc, Delivery ID, Courier ID, Payment Date, Customer ID from the SALE entity and
	SQL Select
	Customer ID, Customer Name, Customer Surname, Customer Cellphone Number, Customer Email Address and Customer Business name from the CUSTOMER entity
	Then
	Step 7: The system sorts through the retrieved information and filters for the top results by using a count (C#)

Calculate the most frequent buyers by counting the total number of Sales made a particular customer in the past 4 weeks. Sort from most purchases to least purchases.
Then
Step 8: The system displays the most frequent buyers report (Angular)
Display customers who have made the most purchases and the number of purchases they have made as a report on the screen.

Subsystem:	7. Reports Subsystem
Use Case:	7.4 Generate most popular location report
Language:	C#
Pseudo Code:	Step 1: The requests to generate a report
	Then
	Step 2: The admin clicks on the Reports menu option (Angular)
	Onclick the Reports Menu item
	Then
	Step 3: The system displays the "Reports" screen (Angular)
	Display Reports Screen
	Then
	Step 4: The admin selects the "most popular location report" (Angular)
	Onclick Report type dropdown
	Then
	Step 5: The admin clicks the "Generate" button (Angular)
	Onclick Generate Button
	Then
	Step 6: System retrieves report records from the database (C#)
	SQL Select

Sale ID, SaleOrder Date, SaleOrder Desc, Delivery ID, Courier ID, Payment Date, Customer ID from the SALE entity

SQL Select

Customer ID, Customer Name, Customer Surname, Customer Cellphone Number, Customer Email Address and Customer Business name from the CUSTOMER entity

SQL Select

Address ID, Customer ID, Province ID, Address line 1, Address line 2, Address line 3, Address Postal Code from the Address entity

SQL Select

Province ID, Province Description from the Province entity

SQL Select

City ID, Province ID, City Description from the City entity

SQL Select

Suburb ID, City ID, Suburb Description from the Suburb entity

Then

Step 7: The system sorts through the retrieved information and filters for the areas which had the most sales in the past month (C#)

Calculate the number customers which have been placing the most orders.

Calculate the locations where these customers are located

Sort the information by the province where the most orders have been placed over the last month.

Then

Step 8: The system displays the most popular location report (Angular)

Display the locations (province – city – suburb) where the most purchases have been made in the past month

Subsystem:	7. Reports Subsystem
Use Case:	7.5 Generate Weekly Sale Order List
Language:	C#
Pseudo Code:	Step 1: The requests to generate a report
	Then
	Step 2: The admin clicks on the Reports menu option (Angular)
	Onclick the Reports Menu item
	Then
	Step 3: The system displays the "Reports" screen (Angular)
	Display Reports Screen
	Then
	Step 4: The admin selects the "Weekly Sale Order List" (Angular)
	Onclick Report type dropdown
	Then
	Step 5: The admin clicks the "Generate" button (Angular)
	Onclick Generate Button
	Then
	Step 6: System retrieves report records from the database (C#)
	SQL Select
	Sale ID, SaleOrder Date, SaleOrder Desc, Delivery ID, Courier ID, Payment Date, Customer ID from the SALE entity and
	SQL Select
	Customer ID, Customer Name, Customer Surname, Customer Cellphone Number, Customer Email Address and Customer Business name from the CUSTOMER entity
	Then
	Step 7: The system then selects the Sales which have occurred in the past week and puts them in a list with the corresponding customers who made those sale orders also in the list. (C#)

Select sales made in the past 7 days order by SaleOrder date (last placed first)
Then Step 8: The system then displays the weekly sale order list to the admin (Angular)
Display the table list of sales made over the past week according to the date the sales where placed. Table list contains customer and matching sale information.

Subsystem:	7. Reports Subsystem
Use Case:	7. 6 Generate Weekly Sale Order List
Language:	C#
Pseudo Code:	Step 1: The requests to generate a report
	Then
	Step 2: The admin clicks on the Reports menu option (Angular)
	Onclick the Reports Menu item
	Then
	Step 3: The system displays the "Reports" screen (Angular)
	Display Reports Screen
	Then
	Step 4: The admin selects the "Weekly Sale Order List" (Angular)
	Onclick Report type dropdown
	Then
	Step 5: The admin clicks the "Generate" button (Angular)
	Onclick Generate Button
	Then
	Step 6: System retrieves report records from the database (C#)
	SQL Select

Sale ID, SaleOrder Date, SaleOrder Desc, Delivery ID, Courier ID, Payment Date, Customer ID from the SALE entity and
SQL Select
Customer ID, Customer Name, Customer Surname, Customer Cellphone Number, Customer Email Address and Customer Business name from the CUSTOMER entity
Then
Step 7: The system then selects the Sales which have occurred in the past month and puts them in a list with the corresponding customers who made those sale orders also in the list. (C#)
Select sales made in the past 7 days order by SaleOrder date (last placed first)
Then
Step 8: The system then displays the monthly sale order list to the admin (Angular)
Display the table list of sales made over the past month (4 weeks) according to the date the sales were placed. Table list contains customer and matching sale information.

Subsystem:	7. Reports Subsystem
Use Case:	7.7 Generate Stock-Level Report
Language:	C#
Pseudo Code:	Step 1: The requests to generate a report
	Then
	Step 2: The admin clicks on the Reports menu option (Angular)
	Onclick the Reports Menu item
	Then
	Step 3: The system displays the "Reports" screen (Angular)
	Display Reports Screen
	Then

Step 4: The admin selects the "Stock-Level Report" (Angular)

Onclick Report type dropdown

Then

Step 5: The admin clicks the "Generate" button (Angular)

Onclick Generate Button

Then

Step 6: System retrieves report records from the database (C#)

SQL Select

ProductItem ID, ProductItem Name, ProductItem Desc, ProductItem Image, CategoryType ID, Quantity on Hand from the ProductItem entity

SQL Select

CategoryType ID, CategoryType Desc, ProductCategory ID from CategoryType entity

SQL Select

ProductCategory ID, ProductCategory Desc from ProductCategory entity

Then

Step 7: The system categorizes each ProductItem into its Category type and each category type into its Product Category (C#)

Sort retrieved product items into category types and then into product categories.

Order the information according to quantity on hand with the items with the highest quantity on hand being at the top.

Then

Step 8: The system then displays the Stock-level report with the necessary columns (Angular)

Display Product Category Desc, Category Type Desc, Product Item Name and Quantity on Hand as table columns with the rows populated according to the entries in the ProductItem table

Subsystem:	7. Reports Subsystem
Use Case:	7.8 Packing Report
Language:	C#
Pseudo Code:	Step 1: The requests to generate a report
	Then
	Step 2: The admin clicks on the Reports menu option (Angular)
	Onclick the Reports Menu item
	Then
	Step 3: The system displays the "Reports" screen (Angular)
	Display Reports Screen
	Then
	Step 4: The admin selects the "Packing Report" (Angular)
	Onclick Report type dropdown
	Then
	Step 5: The admin clicks the "Generate" button (Angular)
	Onclick Generate Button
	Then
	Step 6: System retrieves report records from the database (C#)
	SQL Select
	Sale ID, SaleOrder Date, SaleOrder Desc, OrderStatus ID, SaleOrder Assign, Payment Date, Payment Amount from the SALE entity
	SQL Select
	OrderStatus ID, OrderStatus Desc from the OrderStatus entity
	Then
	Step 7: The system sorts the retrieved sales from the first made to the most recent
	Select Sales which will have the order status of "ready for packing"

Sort retrieved sales by sale order date with the sales placed first at the top.
Then
Step 8: The system displays the list of Sales to the admin as the Packing Report (Angular)
Display all the Sale information of the sales with an order status of "ready for packing" as the Packing report

Subsystem:	7. Reports Subsystem
Use Case:	7.9 Deliver Report
Language:	C#
Pseudo Code:	Step 1: The requests to generate a report
	Then
	Step 2: The admin clicks on the Reports menu option (Angular)
	Onclick the Reports Menu item
	Then
	Step 3: The system displays the "Reports" screen (Angular)
	Display Reports Screen
	Then
	Step 4: The admin selects the "Delivery Report" (Angular)
	Onclick Report type dropdown
	Then
	Step 5: The admin clicks the "Generate" button (Angular)
	Onclick Generate Button
	Then
	Step 6: System retrieves report records from the database (C#)
	SQL Select
	Sale ID, SaleOrder Date, SaleOrder Desc, OrderStatus ID, SaleOrder Assign, Payment Date, Payment Amount from the SALE entity

SQL Select

OrderStatus ID, OrderStatus Desc from the OrderStatus entity

SQL Select

Delivery ID, Courier ID, Delivery Date, Address ID

SQL Select

Address ID, Customer ID, Province ID, Address line 1, Address line 2, Address line 3, Address Postal Code from the Address entity

SQL Select

Province ID, Province Description from the Province entity

SQL Select

City ID, Province ID, City Description from the City entity

SQL Select

Suburb ID, City ID, Suburb Description from the Suburb entity

SQL Select

Customer ID, Customer Name from the Customer entity

Then

Step 7: The system then filters through the retrieved sales and sorts them according to SaleOrder_Date.

Select Sales which will have the order status of "ready for delivery"

Sort retrieved sales by sale order date with the sales placed first at the top.

Then

Step 8: The system then displays the list of retrieved Sales to the admin in the correct order as the Delivery Report (Angular)

Display all the Sale, delivery, customer and location information of the sales with an order status of "ready for delivery" as the Delivery report

6.1.1. DELIVERY SUBSYSTEM

Subsystem:	8. Employee Delivery Shift	
Use Case:	8.1 Search Delivery Details	
Language:	C#	
Pseudo Code:	Step 1: Employee request to View deliveries	
	THEN	
	Step 2: Employee clicks Delivery Menu Item (Angular)	
	ON CLICK Delivery Menu Item	
	THEN ON CLICK Delivery Dropdown menu Item	
	THEN	
	Step 3: Load Delivery Screen (Angular)	
	SQL Select	
	Delivery information	
	From Delivery table	
	SQL Select	
	Courier information	
	From Courier table	
	SQL Select	
	Address information	
	From Address table	
	SQL Select	
	Province information	
	From Province table	
	SQL Select	
	City information	
	From City table	

SQL Select

Suburb information

From Suburb table

SQL Select

OrderStatus information

From OrderStatus table

Retrieve the following:

- Delivery_ID
- o Courier_TrackingNumber
- Delivery_Date
- Courier_Name
- o Courier Number
- Courier_Email
- o Address_Line1
- o Address Line2
- o Address_Line3
- Address_PostalCode
- Province_Description
- City_Description
- Suburb_Description
- OrderStatus_Desc

THEN display the information in a table.

Display Delivery Screen

[ALT STEP 3]: The system fails to load delivery page(Angular)

ON CLICK Delivery Menu Item

THEN ON CLICK Delivery Dropdown menu Item

System fails to load delivery page.

THEN display Error message, "There was an error loading the Delivery page. Please try again later."

Terminate

6.1.2. PRODUCTS SUBSYSTEM

6.1.3. SPECIALS SUBSYSTEM

Subsystem:	10. Special	
Use Case:	10.1 Add Special	
Language:	C#	
Pseudo Code:	Step 1: Request to add a special [Angular]	
	ON CLICK "Add Special" button	
	THEN	
	Step 2: Display Add Special screen [Angular]	
	Display Add Special Screen	
	ON CLICK "Add" button	
	THEN	
	Step 3: Capture, validate and save new special [Angular]	
	SQL SELECT Special	
	FROM Special entity	
	WHERE Special_Image == User Input,	
	Special_Description == User Input,	
	Special_Price == User Input,	
	Special_StartDate == User Input,	
	Special_EndDate == User Input	
	If(User Input == Valid)	
	THEN	
	[ALT Step 3]: Display validation error areas [Angular]	
	Else if(User Input == NotValid)	
	Display error areas by circling the input areas that need to be entered correctly.	

[ALT Step 3]: Admin clicks on the "Cancel" button [Angular]

ONCLICK "Cancel" button

THEN

Step 4: The system adds the new special on to the system.

SQL ADD

Special_ID, Special_Description, Special_Image, Special_StartDate, Special_EndDate, Special_Price

TO Special entity

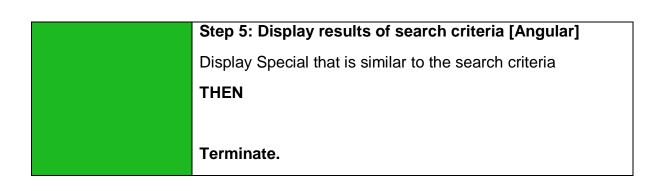
THEN

Step 5: The new special is added on to the table

Display Special screen

Terminate.

Subsystem:	10. Special	
Use Case:	10.2 Search Special	
Language:	C#	
Pseudo Code:	Step 1: Request to search special [Angular]	
	ON CLICK "Special" option	
	THEN	
	Step 2: Display Special screen [Angular]	
	SQL SELECT	
	All Specials	
	FROM Special table	
	THEN	
	Step 3: Admin enters search criteria [Angular]	
	Enter search criteria	
	THEN	
	Step 4: Capture and validate search criteria [Angular]	
	ON CLICK Search input field	
	SQL SELECT	
	Any Special	
	FROM Special table	
	WHERE Special == Search Criteria	
	THEN	
	[ALT Step 4]: Matching special could not be found [Angular]	
	If (Search criteria == NoMatch)	
	Display "Enter a new search criteria".	



Subsystem:	10. Special	
Use Case:	10.3 Update Special	
Language:	C#	
Pseudo Code:	Step 1: Request to update special [Angular]	
	ON CLICK "Special" option	
	THEN	
	Step 2: Display special screen [Angular]	
	SQL SELECT	
	All Specials	
	FROM Special table	
	THEN	
	Step 3: Invoke Use Case 10.10 Search Special [Angular]	
	Search for special	
	THEN	
	Step 4: Click on the update button [Angular]	
	ONCLICK "Update" button	
	THEN	
	Step 5: Display Update special screen [Angular]	
	Display update special screen	
	Step 6: Input updated details [Angular]	
	Fill input fields	
	THEN	
	Step 7: Admin clicks on the update button [Angular]	
	ON CLICK "Update" button	

[ALT Step 7]: Cancel update of special [Angular]

ON CLICK "Cancel" button

THEN

Step 8: Capture and validate special [Angular]

SQL UPDATE

Special

FROM Special Table

If(Admin Input == Filled && Valid)

Capture details

[ALT Step 8]: Validation failed

If (Admin Input == NotValid)

THEN

Step 9: Save updated special details [Angular]

SQL UPDATE

Special_Description, Special_Image, Special_StartDate, Special_EndDate, Special_Price

IN Special entity

THEN

Terminate

Language: C# Pseudo Code: Step 1: Request to delete special [Angular] ON CLICK "Special" option THEN Step 2: Display special screen [Angular] SQL SELECT All Specials FROM Special table THEN Step 3: Invoke Use Case 10.10 Search Special [Angular] Search for special THEN Step 4: Click on the delete button [Angular] ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special". THEN	Subsystem:	10. Special	
Pseudo Code: Step 1: Request to delete special [Angular] ON CLICK "Special" option THEN Step 2: Display special screen [Angular] SQL SELECT All Specials FROM Special table THEN Step 3: Invoke Use Case 10.10 Search Special [Angular] Search for special THEN Step 4: Click on the delete button [Angular] ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".	Use Case:	10.4 Delete Special	
ON CLICK "Special" option THEN Step 2: Display special screen [Angular] SQL SELECT All Specials FROM Special table THEN Step 3: Invoke Use Case 10.10 Search Special [Angular] Search for special THEN Step 4: Click on the delete button [Angular] ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".	Language:	C#	
Step 2: Display special screen [Angular] SQL SELECT All Specials FROM Special table THEN Step 3: Invoke Use Case 10.10 Search Special [Angular] Search for special THEN Step 4: Click on the delete button [Angular] ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".	Pseudo Code:	Step 1: Request to delete special [Angular]	
Step 2: Display special screen [Angular] SQL SELECT All Specials FROM Special table THEN Step 3: Invoke Use Case 10.10 Search Special [Angular] Search for special THEN Step 4: Click on the delete button [Angular] ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".		ON CLICK "Special" option	
SQL SELECT All Specials FROM Special table THEN Step 3: Invoke Use Case 10.10 Search Special [Angular] Search for special THEN Step 4: Click on the delete button [Angular] ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".		THEN	
SQL SELECT All Specials FROM Special table THEN Step 3: Invoke Use Case 10.10 Search Special [Angular] Search for special THEN Step 4: Click on the delete button [Angular] ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".			
All Specials FROM Special table THEN Step 3: Invoke Use Case 10.10 Search Special [Angular] Search for special THEN Step 4: Click on the delete button [Angular] ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".		Step 2: Display special screen [Angular]	
THEN Step 3: Invoke Use Case 10.10 Search Special [Angular] Search for special THEN Step 4: Click on the delete button [Angular] ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".		SQL SELECT	
Step 3: Invoke Use Case 10.10 Search Special [Angular] Search for special THEN Step 4: Click on the delete button [Angular] ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".		All Specials	
Step 3: Invoke Use Case 10.10 Search Special [Angular] Search for special THEN Step 4: Click on the delete button [Angular] ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".		FROM Special table	
Search for special THEN Step 4: Click on the delete button [Angular] ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".		THEN	
Search for special THEN Step 4: Click on the delete button [Angular] ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".			
THEN Step 4: Click on the delete button [Angular] ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".		Step 3: Invoke Use Case 10.10 Search Special [Angular]	
Step 4: Click on the delete button [Angular] ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".		Search for special	
ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".		THEN	
ONCLICK "Delete" button THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".			
THEN Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".			
Step 5: Confirm deletion of special [Angualr] Display confirm delete message "Are you sure you want to delete this special".			
Display confirm delete message "Are you sure you want to delete this special".		THEN	
Display confirm delete message "Are you sure you want to delete this special".		Step 5: Confirm deletion of special [Angualr]	
		Display confirm delete message "Are you sure you want to	
		·	
Step 6: Confirm deletion of special [Angular]		Step 6: Confirm deletion of special [Angular]	
ON CLICK "Confirm" button		ON CLICK "Confirm" button	
THEN		THEN	

[ALT Step 6]: Admin clicks on the "Cancel" button [Angular]

ON CLICK "Cancel" button

Step 8: Delete special

SQL DELETE

Special_ID, Special_Description, Special_Image, Special_StartDate, Special_EndDate, Special_Price

IN Special entity

THEN

Terminate

Subsystem:	10. Special	
Use Case:	10.5	
Language:	C#	
Pseudo Code:	Step 1: Request to send promotional emails [Angular]	
	ON CLICK "Special" option	
	THEN	
	Step 2: Display special screen [Angular]	
	SQL SELECT	
	All Specials	
	FROM Special table	
	THEN	
	Step 3: Click on the Send Promotions button [Angular]	
	ONCLICK "Send Promotions" button.	
	THEN	
	Step 4: Display Send promotions screen [Angular]	
	Display send promotions screen	
	THEN	
	Step 5: Select the specials that the admin wants to add in the email. [Angular]	
	Select the relevant specials.	
	THEN	
	Step 6: Read the information on the selected specials [Angular]	
	SQL READ	
	Special_ID, Special_Description, Special_Image, Special_StartDate, Special_EndDate, Special_Price	
	FROM Special table	

THEN

Step 7: The admin clicks on the Send email button [Angular]

ONCLICK "Send email" button.

[ALT Step 7]: Admin clicks on the cancel button [Angular]

ONCLICK "Cancel" button

THEN

Step 8: The system sends the email to the customers.

SQL READ

Customer_EmailAddress

FROM Customer table

System generates an email and sends it to the emails of the customers that are registered on the system.

THEN

Terminate.

CONCLUSION

This section concludes. It contained the Object-Oriented Design for the NKAP Bolting System. This included a class diagram for the entire system, state diagrams for each object in the class diagram as well as sequence diagrams for each Use Case and Pseudo Code.

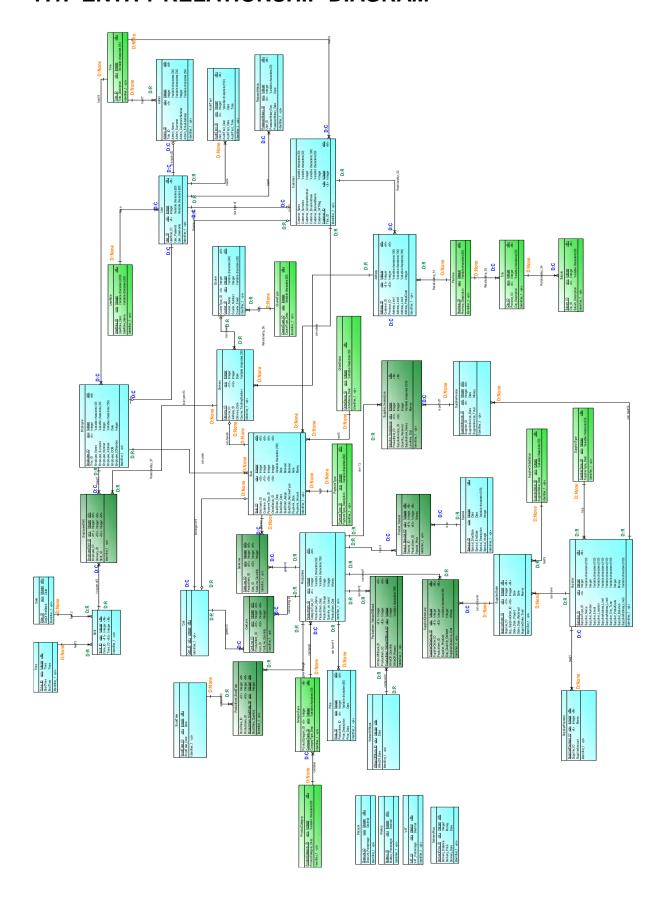
SYSTEM DATABASE DESIGN

7. Entity Relationship Model

INTRODUCTION

In this section, we will be discussing the technical Entity Relationship Diagram for NKAP Bolting Systems database.

7.1. ENTITY RELATIONSHIP DIAGRAM



CONCLUSION

This section contains the finalized third normal form technical Entity Relationship Diagram for NKAP Bolting.

8. Size Estimation of Proposed Database

INTRODUCTION

The size of the planned NKAP Bolting database will be estimated in this part, with each table in the Entity Relationship Diagram and their expected sizes multiplied by the expected rows for that table to generate a total size for that table, as well as a 5-year projection.

8.1. SIZE ESTIMATION OF PROPOSED DATABASE

Entity Name	Expected Rows	Entity (in bytes)	Total Size (In bytes)
Address	410	300	123000
Admin	3	140	420
Audit Trail	7600	182	1383200
Cart	410	10	4100
CartLine	410	40	16400
Category Type	65	70	4550
City	74	70	5180
Courier	5	110	550
Courier Type	4	40	160
Customer	410	250	102500
Date	6	18	108
Delivery	574	68	39032
Delivery Price	7	38	266
Discount	5	14	70
Employee	4	278	1112
EmployeeShift	4	40	160
Markup	7	14	98
OrderStatus	6	40	240

PasswordHistory	517	48	24816
PaymentType	1	60	60
Price	845	78	65910
Product_Special	150	40	6000
ProductCategory	13	60	780
ProductItem	845	250	211250
ProductItem_StockTake	52	40	2080
ProductItem_WrittenOffStock	48	140	6720
Entity Name	Expected Rows	Entity (in bytes)	Total Size (In bytes)
Province	9	60	540
Sale	2870	230	660100
SaleLine	2870	40	114800
Shift	5	30	150
Special	150	136	20400
StockTake	52	18	936
Suburb	740	70	6800
Supplier	93	900	83700
Supplier_InvoiceLine	83	150	12450
SupplierInvoice	83	38	3154
SupplierOrder	112	46	5152
SupplierOrderLine	112	90	10080
SupplierOrderStatus	2	60	180
SupplierPayment	90	30	3600
SupplierType	8	60	480
Time	4	18	72
Title	6	25	200
User	417	120	50040
UserRole	3	260	780
VAT	3	16	48

WrittenOffStock	48	18	864
Total Bytes			2 973 288
Total bytes (MB)			2.973288

8.2. FIVE YEAR PROJECTION

Year	Expected Growth	Expected Total Growth	Expected Size
Current	N/A	N/A	2 973 288
2022	5%	105%	3 121 952
2023	8%	113%	3 359 815
2024	13%	126%	3 746 341
2025	15%	143%	4 251 802
2026	20%	160%	4 757 261
Overall Total Bytes	4 757 261		
Overall Total (MB)	4.757261		

CONCLUSION

This section now contains the NKAP Bolting database will be estimated in this part, with each table in the Entity Relationship Diagram and their expected sizes multiplied by the expected rows for that table to generate a total size for that table, as well as a 5-year projection.

SYSTEM INTERFACE DESIGN

9. Input Design

INTRODUCTION

This section will cover the screens, web pages, and menus for the Website Application and Desktop Application, as well as their controls and functionality for the suggested NKAP Bolting System's functionality

- 9.1.1. LOGIN SUBSYSTEM
- 9.1.2. SUPPLIER SUBSYTEM
- 9.1.3. USER SUBSYSTEM
- 9.1.4. EMPLOYEE SUBSYSTEM
- 9.1.5. ADMIN SUBSYSTEM
- 9.1.6. CUSTOMER SUBSYSTEM
- 9.1.7. REPORTS SUBSYSTEM
- 9.1.8. DELIVERY SUBSYSTEM
- 9.1.9. PRODUCTS SUBSYSTEM
- 9.1.10. SPECIALS SUBSYSTEM

CONCLUSION

This section now contains the screens, web pages, and menus for the Website Application and Desktop Application, as well as their controls and functionality for the suggested NKAP Bolting System's functionality

10. Output Design

INTRODUCTION

In this section we will be looking at the output design of our system. The section will contain the reports and outputs that are generated by the system as well as a description of each report, when they will be run, and the possible entities involved with the reports.

Design Principles

No.	Design Issue	Design Guideline	Example
1.	Page Size	A4	N/A
2.	Page Orientation	Portrait	N/A
3.	NKAP Bolting Logo		BOLTING SPECIALISTS IN PASTENING SYSTEMS
4.	Date Report was generated	Font: Arial Bold Font Size: 12 Positioning: Left	Date Generated :21/06/21
5.	Company Name	Font: Arial Bold Font Size: 12 Positioning: Left	Company Name: NKAP Bolting CC
6.	Font	Arial	Collection
7.	Font Size	12	Collection
8.	Bar Chart		Provincial Graph 100 100 100 100 100 100 100 1
9.	Table Columns	Font: Arial Bold Font Size: 12	Collection/Delivery
10.	Table Rows	Font: Arial Font Size: 12	Collection
11.	Report Heading	Font: Arial Font Size: 36	Packing Report
	Number:	7.1	
Report	: Name:	Generate Fast Selling	g Product List

12.	Font Color	Black	Collection	
Short	Description:		This report displays the products which had the highest number of sales over a selected period.	
Frequency:		Period is adjustable. I	Period is adjustable. Usually generated on a monthly basis.	
Special Features: The Report period is adjustable.		adjustable.		

Selection Criteria					
Criteria	Default	Туре	'All' Sections?	Multiple Sections	Values
Start Date	N/A	Date picker	No	No	Any valid date
End Date	N/A	Date picker	No	No	Any valid date

Report: Fast Selli	Report: Fast Selling Product List						
	Report Fields						
Field Name	Format	Max Length	Justified	Source			
Company Name	Text	50	Left	Hardcoded			
Date	dd/mm/yyyy	10	Left	System.DateTime.Now			
				on generation of the			
				report.			
Logo	Image	N/A	Right	Hardcoded			
Fast-Selling	Text	30	Centre	Hardcoded			
Product List							
Report Summary	Text	13	N/A	Hardcoded			
Ranking	Numeric	1	N/A	<u>Hardcoded</u>			
Product	Text	50	N/A	ProductItem Entity			
				ProductItem_Name			
Full Report	Text	10	Centre	Hardcoded			
Product Item ID	Numeric	6	N/A	ProductItem Entity			
				ProductItem_ID			
Product Item	Text	50	N/A	ProductItem Entity			
Name				ProductItem_Name			
Quantity	Numeric	1000	N/A	Generated from Sale			
Purchased				ProductItem Entity			
Number of	Numeric	1000	N/A	Generated from Sale			
Purchases				entity			
For the period:	Text	13	Centre	Hardcoded			
Report period	dd/mm/yyyy	20	Centre	Start and End date			
				inputted			

Report Layout:

Company Name: xxxxxxxxxxxxxxxx Date Generated :99/99/9999





Report Summary				
Ranking Product				
9	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx			
9	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx			
9	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx			

XXXX XXXXX

Product Item ID	Product Item Name	Quantity Purchased	Number of Purchases
	Xxxxxxxxxxxxxxxxxx	9999	9999
999999	Xxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
999999	Xxxxxxxxxxxxxxxxxx	9999	9999
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxx		
999999	Xxxxxxxxxxxxxxxxxx	9999	9999
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999	9999
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxxx		
999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999	9999
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
999999	Xxxxxxxxxxxxxxxxxx	9999	9999
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999	9999
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999	9999
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		

Example Report:

Company Name: NKAP Bolting CC Date Generated :21/06/2021



Fast-Selling Product List

Report Summary				
Ranking	Ranking Product			
1	M8 Cap Nut			
2	3mm Socket Cap Screw			
3	M40 Hex Head Bolt			

Full Report

Product Item ID	Product Item Name	Quantity Purchased	Number of Purchases
023	3mm Socket Cap Screw	300	20
024	3.5mm Deck Screw	250	19
025	4mm Sheet Metal Screw	230	17
007	M10 Square Head Bolt	160	26
800	M16 Shoulder Head Bolt	150	23
009	M40 Hex Head Bolt	145	48
063	M8 Cap Nut	80	26
067	M3.5 Hex Nut	75	27

For the period: 03/04/2020 - 24/06/2020

Report Number:	7.2
Report Name:	Generate slow-selling product list

Short Description:	This report displays the products which had the lowest number of sales over a selected period.	
Frequency:	Period is adjustable. Usually generated on a monthly basis.	
Special Features:	The Report period is adjustable.	

Selection Criteria					
Criteria	Default	Туре	'All' Sections?	Multiple Sections	Values
Start Date	N/A	Date picker	No	No	Any valid date
End Date	N/A	Date picker	No	No	Any valid date

Report: Slow Sel	Report: Slow Selling Product List					
Report Fields						
Field Name	Format	Max Length	Justified	Source		
Company Name	Text	50	Left	Hardcoded		
Date	dd/mm/yyyy	10	Left	System.DateTime.Now on generation of the report.		
Logo	Image	N/A	Right	Hardcoded		
Slow-Selling Product List	Text	30	Centre	Hardcoded		
Product Item ID	Numeric	6	N/A	ProductItem Entity ProductItem_ID		
Product Item Name	Text	50	N/A	ProductItem Entity ProductItem_Name		
Quantity Purchased	Numeric	1000	N/A	Generated from Sale ProductItem Entity		
Number of Purchases	Numeric	1000	N/A	Generated from Sale entity		
For the period:	Text	13	Centre	Hardcoded		
Report period	dd/mm/yyyy	20	Centre	Start and End date inputted		

Report Layout:

Company Name: xxxxxxxxxxxxxxxxx

Date Generated :99/99/9999





Product Item ID	Product Item Name	Quantity Purchased	Number of Purchases
999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999	9999
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999	9999
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999	9999
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999	9999
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999	9999
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999	9999
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999	9999
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
999999	Xxxxxxxxxxxxxxxx	9999	9999
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		
	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx		

Example Report:

Company Name: NKAP Bolting CC Date Generated :21/06/2021



Slow-Selling Product List

Product Item ID	Product Item Name	Quantity Purchased	Number of Purchases
026	8mm Wood Screw	10	1
027	12mm Machine Screw	13	1
028	12mm Thread Cutting Screw	14	2
005	M40 Carriage Bolt	20	1
006	M8 Plow Bolt	20	2
004	M30 Eye Head Bolt	22	3
064	M12 Tee Nut	25	2
068	M20Acorn Nut	25	4

For the period: 03/04/2020 - 24/06/2020

Report Number:	7.3	
Report Name:	Generate most frequent buyers report	
Short Description:	This report shows the customers who have purchased most frequently during the course of one month.	
Frequency:	Monthly	
Special Features:	None	

Report: Most Frequent Buyer					
Report Fields					
Field Name	Format	Max Length	Justified	Source	
Company Name	Text	50	Left	Hardcoded	
Date	dd/mm/ yyyy	10	Left	System.DateTime.Now on	
				generation of the report.	
Logo	Image	N/A	Right	Hardcoded	
Most Frequent	Text	30	Centre	Hardcoded	
Buyer Report					
Report	Text	13	N/A	Hardcoded	
Summary					
Customer:	Text	9	N/A	Customer Entity	
				Customer_Name	
				Customer_Surname	
Purchases	Numeric	4	N/A	Hardcoded	
Made					
Full Report	Text	10	Centre	Hardcoded	
Customer ID	Numeric	6	N/A	Customer Entity	
				Customer_ID	
Customer Name	Text	60	N/A	Customer Entity	
				Customer_Name	
				Customer_Surname	
Business Name	Text	50	N/A	<u>Customer Entity</u>	
				Customer_BusinessName	
Contact Number	Numeric	10	N/A	<u>Customer Entity</u>	
				Customer_CellphoneNumber	
Email Address	Text	100	N/A	<u>Customer Entity</u>	
<u> </u>			1.1/2	Customer_EmailAddress	
No. of	Numeric	3	N/A	Generated from Sale and	
purchases				<u>Customer</u> entities	
made					

Report Layout:

Company Name: xxxxxxxxxxxxxxxx

Date Generated:99/99/9999





Report Summary				
Customer:	Purchases Made			
x0000000000000000000000000000000000000	9999			
X00000000000000000	9999			
X0000000000000000	9999			

XXXX XXXXX

Customer ID	Customer Name	Business Name	Contact Number	Email Address	No. of purchases made
999999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	999 999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	999
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	9999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	333
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
999999	XXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	999 999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	999
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	9999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	333
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
999999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	999 999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	999
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	9999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	333
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
999999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	999 999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	999
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	9999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	333
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
999999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
	XXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	999 999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	999
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	9999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	333
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
999999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	999 999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	999
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	9999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	333
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
999999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	999 999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	999
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	9999	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	535
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	

Example Report:

Company Name: NKAP Bolting CC Date Generated :21/06/2021



Most Frequent Buyer Report

Report Summary				
Customer:	Purchases Made			
S'nethemba Xulu	30			
Jerome Amenigy	26			
Divva Bagratee	20			

Full Report

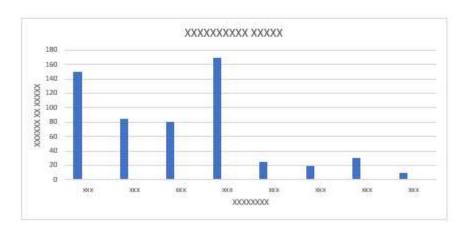
Customer ID	Customer Name	Business Name	Contact Number	<u>Email</u> <u>Address</u>	No. of purchases made
4579	S'nethemba Xulu	SAX Industries	085 478 0789	snethembax @gmail.com	22
3591	Jerome Amenigy	Rome EJ & Co Incorporate d	061 349 6490	jeromea@gm ail.com	18
6845	Divya Bagratee	Divs PTY (Ltd)	023 871 4841	divyab@gmai Lcom	16
9471	Charde Vries	DC PTY (Ltd)	085 692 0954	chardev@gm ail.com	14
6218	Sonali Badrinath	Nikita's Bolts Limited	089 761 4502	sonalib@gma il.com	9
9045	Jane Doe	J&J Doe Industries	093 720 5919	janed@yahoo .com	8
8723	John Doe	J&J Doe Industries	063 401 7630	iohnnyd@gm ail.com	7

Report Number:	7.4	
Report Name:	Generate most popular location report	
Short Description:	This report displays a list of areas where there is the highest concentration of sales over a month.	
Frequency:	Monthly	
Special Features:	Done	

Report: Most Popular Location						
Report Fields						
Field Name	Format	Max Length	Justified	Source		
Company Name	Text	50	Left	Hardcoded		
Date	dd/mm/yyyy	10	Left	System.DateTime.Now on generation of the report.		
Logo	Image	N/A	Right	Hardcoded		
Most Popular Location Graph	Text	50	Centre	Hardcoded		
Bar Chart	Chart	N/A	Centre	Generate Most Popular Location from Address Entity and Province Entity		
Province	Text	50	N/A	Province Entity Proivince_Desc		
Number of Sales	Numeric	1000	N/A	Generate from Sale Entity		





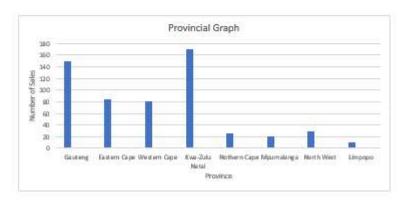


Example Report:

Company Name: NKAP Bolting CC Date Generated :21/06/2021



Most Popular Location Report



Report Number:	7.5
Report Name:	Generate Weekly Sale Order List
Short Description:	This report shows a list of orders placed over a 7-day period.
Frequency:	Weekly
Special Features:	None

Report: Weekly	Report: Weekly Sale Order						
		Report	Fields				
Field Name	Format	Max Length	Justified	Source			
Company Name	Text	50	Left	Hardcoded			
Date	dd/mm/ yyyy	10	Left	System.DateTime.Now on generation of the report.			
Logo	Image	N/A	Right	Hardcoded			
Weekly Sale Order Report	Text	30	Centre	Hardcoded			
Customer ID	Numeric	6	N/A	Customer Entity Customer_ID			
Customer Name	Text	60	N/A	Customer Entity Customer_Name Customer_Surname			
Business Name	Text	50	N/A	Customer Entity Customer_BusinessName			
Contact Number	Numeric	10	N/A	Customer Entity Customer_CellphoneNumber			
Email Address	Text	100	N/A	Customer Entity Customer_EmailAddress			
SaleID	Numeric	6	N/A	Sale Entity Sale_ID			
Sale Date	dd/mm/yyyy	10	N/A	Sale Entity SaleOrder_Date			

Company Name: xxxxxxxxxxxxxx

Date Generated: 99/99/9999





Customer ID	Customer Name	Business Name	Contact Number	Email Address	Sale ID	<u>Date</u>
999999	Xxxxxxxxx Xxxxxxxx Xxxxxxxx Xxxxxxx Xxxxxxx	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	999999	99/99/9999
999999	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	999999	99/99/9999
999999	Xxxxxxxxx Xxxxxxxx Xxxxxxxx Xxxxxxxx Xxxxxxx	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	999999	99/99/9999
999999	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx	Xxxxxxxxxx Xxxxxxxxxx Xxxxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	999999	99/99/9999
999999	Xxxxxxxxx Xxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxx	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	999999	99/99/9999
999999	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	999999	99/99/9999

Company Name: NKAP Bolting CC Date Generated :21/06/2021



Weekly Sale Order Report

+‡+

_							
	Customer ID	Customer Name	Business Name	Contact Number	Email Address	Sale ID	<u>Date</u>
	4579	S'nethemba Xulu	SAX Industries	085 478 0789	snethembax@gmail.com	001	14/06/2021
	3591	Jerome Amenigy	Rome EJ & Co Incorporated	061 349 6490	jeromea@gmail.com	002	14/06/2021
	6845	Divya Bagratee	Divs PTY (Ltd)	023 871 4841	divyab@gmail.com	003	17/06/2021
	9471	Charde Vries	DC PTY (Ltd)	085 692 0954	chardev@gmail.com	004	19/06/2021
	6218	Sonali Badrinath	Nikita's Bolts Limited	089 761 4502	sonalib@gmail.com	005	19/06/2021
	9045	Jane Doe	J&J Doe Industries	093 720 5919	janed@yahoo.com	006	19/06/2021
	8723	John Doe	J&J Doe Industries	063 401 7630	johnnyd@gmail.com	007	20/06/2021

Report Number:	7.6
Report Name:	Generate Monthly Sale Order List
Short Description:	This report shows a list of orders placed over a 4-week period.
Frequency:	Monthly
Special Features:	None

Report: Monthly	Report: Monthly Sale Order						
		Report	Fields				
Field Name	Format	Max Length	Justified	Source			
Company Name	Text	50	Left	Hardcoded			
Date	dd/mm/yyyy	10	Left	System.DateTime.Now on generation of the report.			
Logo	Image	N/A	Right	Hardcoded			
Monthly Sale Order Report	Text	30	Centre	Hardcoded			
Customer ID	Numeric	6	N/A	Customer Entity Customer_ID			
Customer Name	Text	60	N/A	Customer Entity Customer_Name Customer_Surname			
Business Name	Text	50	N/A	Customer Entity Customer_BusinessName			
Contact Number	Numeric	10	N/A	Customer Entity Customer_CellphoneNumber			
Email Address	Text	100	N/A	Customer Entity Customer_EmailAddress			
SaleID	Numeric	6	N/A	Sale Entity Sale_ID			
Date	dd/mm/yyyy	10	N/A	Sale Entity SaleOrder_Date			

Company Name: xxxxxxxxxxxxxx

Date Generated:99/99/9999





Customer ID	Customer Name	Business Name	Contact Number	Email Address	Sale ID	<u>Date</u>
999999	Xxxxxxxxx Xxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxx	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	999999	99/99/9999
999999	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	999999	99/99/9999
999999	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	999999	99/99/9999
999999	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx	Xxxxxxxxxx Xxxxxxxxxx Xxxxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	999999	99/99/9999
999999	Xxxxxxxxx Xxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxx	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	999999	99/99/9999
999999	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	999999	99/99/9999

Company Name: NKAP Bolting CC Date Generated :21/06/2021



Monthly Sale Order Report

Customer ID	Customer Name	Business Name	Contact Number	Email Address	Sale ID	<u>Date</u>
4579	S'nethemba Xulu	SAX Industries	085 478 0789	snethembax@gmail.com	001	14/06/2021
3591	Jerome Amenigy	Rome EJ & Co Incorporated	061 349 6490	jeromea@gmail.com	002	14/06/2021
6845	Divya Bagratee	Divs PTY (Ltd)	023 871 4841	divyab@gmail.com	003	17/06/2021
9471	Charde Vries	DC PTY (Ltd)	085 692 0954	chardev@gmail.com	004	19/06/2021
6218	Sonali Badrinath	Nikita's Bolts Limited	089 761 4502	sonalib@gmail.com	005	19/06/2021
9045	Jane Doe	J&J Doe Industries	093 720 5919	janed@yahoo.com	006	19/06/2021
8723	John Doe	J&J Doe Industries	063 401 7630	johnnyd@gmail.com	007	20/06/2021

Report Number:	7.7
Report Name:	Generate Stock-Level Report
Short Description:	This report displays all the products available on the system as well as the quantity on hand of those products at a given time.
Frequency:	Monthly
Special Features:	None

Report: Stock-Lev	Report: Stock-Level							
		Report Fiel	ds					
Field Name	Format	Max Length	Justified	Source				
Company Name	Text	50	Left	Hardcoded				
Date	dd/mm/ yyyy	10	Left	System.DateTime.Now on generation of the report.				
Logo	Image	N/A	Right	Hardcoded				
Stock-level Report	Text	30	Centre	Hardcoded				
Product Category	Text	50	N/A	ProductItem Entity ProductCategory_Desc				
Category Type	Text	50	N/A	ProductItem Entity CategoryType_ID				
Product Item ID	Numeric	6	N/A	ProductItem Entity ProductItem_ID				
Product Item Name	Text	50	N/A	ProductItem Entity ProductItem_Name				
Quantity on Hand	Numeric	1000	N/A	ProductItem Entity Quantity_on_Hand				





Product Category	Category Type	Product Item ID	Product Item Name	Quantity on Hand
	x0000000000000000000000000000000000000	999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999
xxxxxxxxx	x0000000000000000000000000000000000000	999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999
	x0000000000000000000000000000000000000	999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999
	x0000000000000000000000000000000000000	999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999
xxxxxxxxx	x0000000000000000000000000000000000000	999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999
	x0000000000000000000000000000000000000	999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999
xxxxxxxxxx	x0000000000000000000000000000000000000	999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999
	x0000000000000000000000000000000000000	999999	Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	9999

Company Name: NKAP Bolting CC Date Generated :21/06/2021



Stock-Level Report

Product Category	Category Type	Product Item ID	Product Item Name	Quantity on Hand
	Socket Cap	023	3mm Socket Cap Screw	360
Screws	Deck	024	3.5mm Deck Screw	1150
	Sheet Metal	025	4mm Sheet Metal Screw	130
	Square Head	007	M10 Square Head Bolt	600
Bolts	Shoulder	800	M16 Shoulder Head Bolt	50
	Hex Head	009	M40 Hex Head Bolt	45
Nuts	Сар	063	M8 Cap Nut	600
11015	Hex	067	M3.5 Hex Nut	750

Report Number:	7.8
Report Name:	Packing Report
Short Description:	This report shows a list of orders which require packing (where
	the Sale order status = "Needs packing")
Frequency:	Daily
Special Features:	None

Report: Packing Report							
		Report Fie	lds				
Field Name	Format	Max Length	Justified	Source			
Company Name	Text	50	Left	Hardcoded			
Date	dd/mm/ yyyy	10	Left	System.DateTime.No w on generation of the report.			
Logo	Image	N/A	Right	Hardcoded			
Packing Report	Text	14	Centre	Hardcoded			
SaleID	Numeric	6	N/A	Sale Entity Sale_ID			
Description	Text	150	N/A	Sale Entity SaleOrder_Desc			
Date	dd/mm/yyyy	10	N/A	Sale Entity SaleOrder_Date			
Collection/Delivery	Text	20	N/A	SaleEntity SaleOrder_RecieveType			
Assigned To	Text	30	N/A	Sale Entity SaleOrder_Assign			
Payment Date	dd/mm/yyyy	10	N/A	Sale Entity Payment_Date			





Sale ID	Description	<u>Date</u>	Collection/Delivery	Assigned To	Payment Date
999999		99/99/9999	x000000000	-	99/99/9999
999999		99/99/9999	x000000000	-	99/99/9999
999999		99/99/9999	x000000000X	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	99/99/9999
999999		99/99/9999	x000000000	-	99/99/9999
999999		99/99/9999	x000000000	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	99/99/9999
999999		99/99/9999	x000000000x	x0000000000000000000000000000000000000	99/99/9999
999999		99/99/9999	x000000000x	-	99/99/9999

Company Name: NKAP Bolting CC Date Generated :21/06/2021



Packing Report

Sale ID	Description	<u>Date</u>	Collection/Delivery	Assigned To	Payment Date
001		14/06/2021	Collection	-	14/08/2021
002		14/06/2021	Collection	-	14/08/2021
003		17/06/2021	Delivery	S'nethemba Xulu	17/08/2021
004		19/06/2021	Collection	-	19/06/2021
005		19/06/2021	Delivery	Jerome Amenigy	19/06/2021
006		19/06/2021	Delivery	Divya Bagratee	19/08/2021
007		20/06/2021	Collection	-	20/08/2021

Report Number:	7.9
Report Name:	Delivery Report
Short Description:	This report shows a list of orders which require packing (where the Sale order status = "Ready for delivery")
Frequency:	Daily
Special Features:	None

Report: Delivery Report							
		Report	Fields				
Field Name	Format	Max Length	Justified	Source			
Company Name	Text	50	Left	Hardcoded			
Date	dd/mm/yyyy	10	Left	System.DateTime.Now on generation of the report.			
Logo	Image	N/A	Right	Hardcoded			
Delivery Report	Text	30	Centre	Hardcoded			
SaleID	Numeric	6	N/A	Sale Entity Sale_ID			
Date	dd/mm/yyyy	10	N/A	Sale Entity SaleOrder_Date			
Customer ID	Numeric	6	N/A	Customer Entity Customer_ID			
Customer Name	Text	60	N/A	Customer Entity Customer_Name Customer_Surname			
Contact Number	Numeric	10	N/A	Customer Entity Customer_CellphoneNumber			
Email Address	Text	100	N/A	Customer Entity Customer_EmailAddress			
Province	Text	50	N/A	Address Entity Province_ID			
City & Suburb	Text	100	N/A	City Entity City_ID Suburb Entity Suburb_ID			
Address	Text	300	N/A	Address Entity Address_Line1 Address_Line2 Address_Line3 Address_PostalCode			

Company Name: xxxxxxxxxxxxxxxx Date Generated :99/99/9999





Sale ID	<u>Date</u>	Customer ID	Customer Name	Contact Number	Email Address	Address
999999	99/99/9999	999999	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	X0000000000000000000000000000000000000
999999	99/99/9999	999999	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	X0000000000000000000000000000000000000
999999	99/99/9999	999999	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	X0000000000000000000000000000000000000
999999	99/99/9999	999999	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxx Xxxxxxx	999 999 9999	X0000000000000000000000000000000000000	X0000000000000000000000000000000000000
999999	99/99/9999	999999	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx	999 999 9999	X0000000000000000000000000000000000000	X0000000000000000000000000000000000000
999999	99/99/9999	999999	Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx Xxxxxxxxx	999 999 9999	X0000000000000000000000000000000000000	X0000000000000000000000000000000000000

Company Name: NKAP Bolting CC Date Generated :21/06/2021



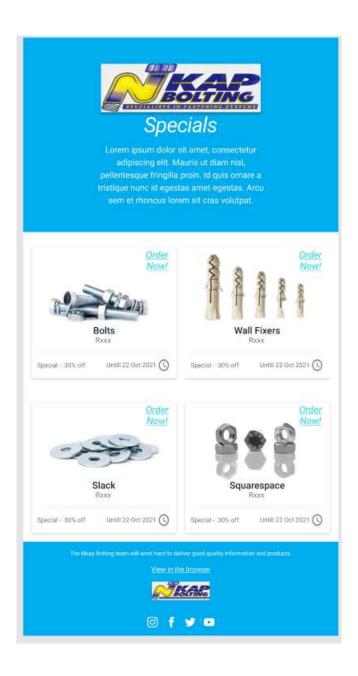
Delivery Report

Sale ID	<u>Date</u>	Customer ID	Customer Name	Contact Number	Email Address	<u>Address</u>
001	14/06/2021	4579	S'nethemba Xulu	085 478 0789	snethembax@gmail.com	313 Etsy Street Lynnwood Pretoria Gauteng
002	14/06/2021	3591	Jerome Amenigy	061 349 6490	jeromea@gmail.com	314 Etsy Street Lynnwood Pretoria Gauteng
003	17/06/2021	6845	Divya Bagratee	023 871 4841	divyab@gmail.com	315 Etsy Street Lynnwood Pretoria Gauteng
004	19/06/2021	9471	Charde Vries	085 692 0954	chardev@gmail.com	316 Etsy Street Lynnwood Pretoria Gauteng
005	19/06/2021	6218	Sonali Badrinath	089 761 4502	sonalib@gmail.com	317 Etsy Street Lynnwood Pretoria Gauteng
006	19/06/2021	9045	Jane Doe	093 720 5919	janed@yahoo.com	13 Etsy Street Lynnwood Pretoria Gauteng
007	20/06/2021	8723	John Doe	063 401 7630	johnnyd@gmail.com	13 Etsy Street Lynnwood Pretoria Gauteng

Reset Password Email



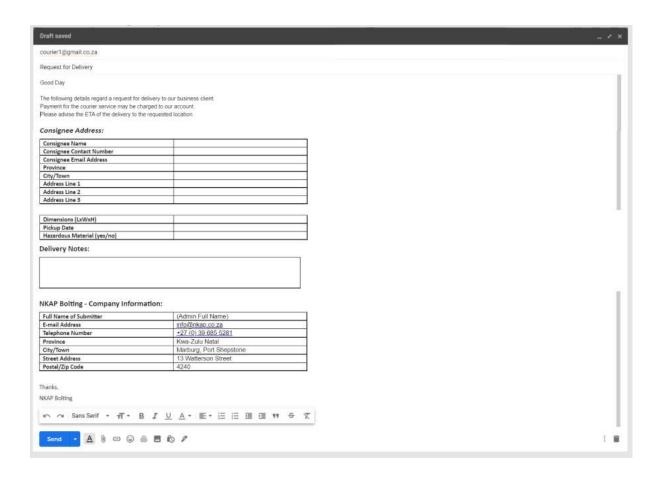
Specials Email



Order Ready for Collection Notification



Courier Delivery Request Email



CONCLUSION

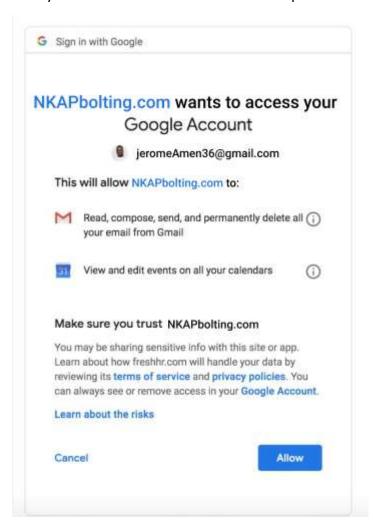
This concludes the output design section of the deliverable, the section contained descriptions of each report, when they will be run, and the possible entities involved in the reports. All reports have been listed and explained with examples

11. Other Interfaces

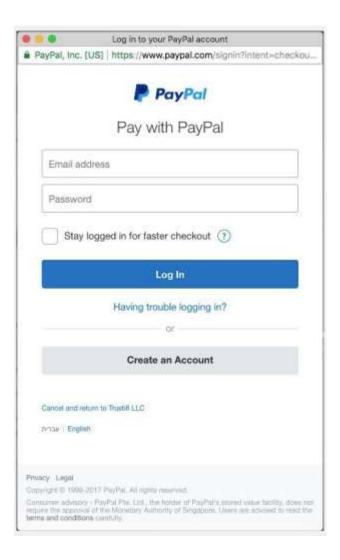
INTRODUCTION

11.1.EXTERNAL APPLICATION INTERFACE TECHNICAL

The system will make use of Gmail to send promotional emails.



The system will make use of PayPal as a virtual card system used during checkout



11.2. DESCRIPTION

CONCLUSION

PACKAGE DESIGN SPECIFICATIONS

12. Test Specifications

INTRODUCTION

This section will contain the detailed test specifications for 6 core use cases in our system. It includes test data, test scenarios, procedures, and the test plan. Each test scenario will account for correct and incorrect scenarios.

12.1.TEST PLAN

12.2.TEST SCENARIOS & PROCEDURES

12.3.TEST DATA

<u>Test Case 1</u>						
Use Case:	1.1 Login					
System:	NKAP Bolting System					
Sub-System:	1					
Short Description:	The use case describes the event where the user wants to log in to the system. The use case begins when the user enters their Log-in details (username and password). The system will verify the details. The use case ends when the user is logged in to the system.					
Pre-Conditions:	The user should have registered previously.					
Expected Functionality:	 The user should be logged in to the system if their login details are correct. 					

Test	Test	Data	Expecte	d Result	Actual	Result	Pass/F
Scenario	Correct Test Data	Incorrect Test Data	Correct Test Data	Incorrec t Test Data	Corre ct Test Data	Incorr ect Test Data	ail
Sonali Badrinath wishes to login to the system. She enters these login details: Usernam e: SonaliB Password : S_onali23 4	Username: SonaliB Password: S_onali234	Usernam e: sonaliB Password : S_onali23 4	System logs user in to the system.	Pop-up message : Incorrect usernam eor passwor d			
Divya Bagratee wishes to login to	Username: DivsB Password:	Usernam e: DIVSB	Pop-up message : Incorrect	System logs user in to the system.			

the system. She enters these login details: Usernam e: DivsB Password : wassupW orld	wassupWo rld	Password : Worldwas sup	usernam eor passwor d			
Devine Vries wishes to login to the system. She enters these login details: Usernam e: Devine_V 01 Password : [Blank]	Username: Devine_V0 1 Password: 370Rocks	Usernam e: Devine_V 01 Password : [Blank]	Pop-up: All fields are required.	System logs user in to the system.		
Jerome Amenigy wishes to login to the system. He enters these login details: Usernam e: [Blank] Password : [Blank]	Username: JJAmen* Password: positiveEn ergy	Usernam e: [Blank] Password : [Blank]	Pop-up: All fields are required	System logs user in to the system.		
S'nethem ba Xulu wishes to	Username: sne_Xulu Password:	Usernam e: Sne_Xulu	System logs user in to the	Pop-up message		
login to	Sne1234	- 5.10_/\did	system.	Incorrect		

the	Password	usernam	
system.	:	eor	
She	Sne1234	passwor	
enters		d	
these			
login			
details:			
Usernam			
e:			
sne_Xulu			
Password			
:			
Sne1234			

	Test Case 2
Use Case:	2.5 Place Supplier Order
System:	NKAP Bolting System
Sub-System:	2
Short Description:	This use case describes the event where the admin navigates to the "Place Supplier Order" screen in order to generate a list of what the business wants to order from the supplier. The admin will select all the items they wish to purchase and the quantity. This list will be saved on to the system in the SupplierOrder and SupplierOrderLine tables.
Pre-Conditions:	 The admin is logged on to the system.
Expected Functionality:	 The admin should have placed a supplier order and the information should be saved in the SupplierOrder and SupplierOrderLine tables

Test	Test	Data	Expected	l Result	Actua	Result	Pass/Fa
Scenari	Correct	Incorrec	Correct	Incorre	Correc	Incorre	il
0	Test	t Test	Test Data	ct Test	t Test	ct Test	
	Data	Data		Data	Data	Data	
Select	-	-	Display	-			
Place			Place				
Supplier			Supplier				
order			Order				
button			screen.				
Choose	Supplier:	Supplier:	Display	Circle			
Supplier	Makro	Null	Supplier	the			
				control			
				type in			
				red			
Choose	Categor	Categor	Display	Circle			
Categor	y: Bolts	y:	category	the			
у		Null		control			

				type in red		
Select confirm button	-	-	Validation passed. Show the product items list.	Circle the control types in red		
Select product items	-	-	Select items.	-		
Add quantity	Quantity : 0 and above	Quantity : Below 0	Display quantity amount	Circle the control type in red		
Select place order	-	-	Validation passed. Save the informatio n.	-		

	Test Case 3					
Use Case:	2.6 Receive Supplier Order					
System:	NKAP Bolting System					
Sub-System:	2. Supplier					
Short Description:	This use case describes the event where the admin receives the order from the supplier. The admin will check if the order is correct by comparing it with the invoice. Then they will capture the order on to the system. The use case ends when the product item quantity on hand is updated.					
Pre-Conditions:	 The order should have been sent already. 					
Expected Functionality:	 The received supplier order should have been captured and the quantity on hand is updated. 					

Test	Test	Expected	d Result	Actua	Pass/Fa		
Scenari o	Correct Test Data	Incorre ct Test Data	Correct Test Data	Incorre ct Test Data	Correc t Test Data	Incorre ct Test Data	il
Select order date	Order Date: 5/15/2021	Order Date: Null	Display order date	Circle the control type in red			
Enter invoice number	Invoice Number: 01055PQ R	Invoice Number: Null	Display invoice number	Circle the control			

				type in		
Select	Invoice	Invoice	Dioploy	red Circle		
invoice	Date:	Date:	Display invoice	the		
date	12/6/2021	Null	date	control		
date	12/0/2021	INGII	date	type in		
				red		
Enter	Invoice	Invoice	Display	Circle		
invoice	Total: 12	Total:	invoice	the		
total	565	Null	total	control		
				type in		
				red		
Select	-	-	Display	Circle		
Add			modal	the		
Button				control		
				types in		
				red		
Enter	Name:	Name:	Display	Circle		
product	Screw	null	product	the		
item	bolt		item	control		
name			name	type in		
Enter	Oughtity:	Quantity	Dioploy	red Circle		
the	Quantity: 25	Quantity : 0 or -2	Display the	the		
quantity	23	. 0 01 -2	quantity	control		
quartity			quantity	type in		
				red		
Enter	Cost: 12	Cost:	Display	Circle		
product		Null	the cost	the		
item				control		
cost				type in		
				red		
Select	-	-	Display	-		
the add			the			
button			Receive			
			Supplier			
			order			
Select	_	_	screen. Display	_		
cancel			the	_		
button			Receive			
Dation			Supplier			
			order			
			screen.			
Select	-	-	Save the	Circle		
Save			informatio	the		
button			n in the	control		
			tables.	types in		
				red		

	Test Case 3						
Use Case:	5.2 Pack Order						
System:	NKAP Bolting System						
Sub-System:	Administration Back Office						
Short Description:	This use case describes the event where an employee searches for orders which require packing and proceeds to pack the order. Once the employee has packed the order, they will indicate so on the system and the system will update the order status of the order appropriately in the Sale Entity . If the Order is for collection, the system will notify the customer that their order is ready for collection via SMS. The use case concludes when the Order has been packed and the status has been updated accordingly.						
Pre-Conditions:	 The Employee must be logged in to the system. 						
	 The sale must exist on the system and have an order status of "Needs Packing". 						
Expected Functionality:	 The system should update the status of the sale to "Ready for Delivery" or "Ready for Collection" depending on the Receival type of the sale in the Sale Entity. The system should notify the customer via SMS that their order is ready for collection if the Receival Type is "Collection". The system should display a success notification once the order has been successfully packed. 						

Test	Test	Data	Expected I	Result	Actua	l Result	Pass/Fail
Scenario	Correct	Incorrect	Correct Test	Incorrect	Correct	Incorrect	
	Test Data	Test	Data	Test	Test	Test	
		Data		Data	Data	Data	
1. Click the			Invoke Use				
Administration			Case 5.1				
Back Office			Search Online				
Menu item on			Sales				
the navigation							
bar and click							
the "Pack							
Order"							
dropdown							
menu item.							
2. Click "Pack			Display Pack				
Order" Button			Order Screen				
3. Capture	Has this		 Update 	Display			
Pack Order	order		Order	Error			
Information	been		Status to	Modal if			
and click	packed?			Pack			

"Proceed"	Check		"Ready for	Order		
button.	Checkbox		Collection".	Process		
				Failed.		
	For	•	[ALT]			
	Delivery		Update			
	Check		Order			
	Checkbox		Status to			
			"Ready for			
	For		Delivery".			
	Collection		Bonvory .			
	Check	•	(If For			
	Checkbox		Collection			
	CHOCKECK		Checkbox			
			=			
			- Checked)			
			Notify			
			Customer			
			via SMS			
			that their			
			order is			
			Ready for			
			Collection.			
			Conection.			
			Diopley			
		•	Display Success			
			Modal			
			Once Pack			
			Order			
			Process is			
			complete.			

	Test Case 5
Use Case:	Checkout Order
System:	NKAP Bolting System
Sub-System:	6.7
Short Description:	This use case describes the event where a customer would like to Checkout all the items in their cart. The system checks if the customer is logged in. The system asks if they would like to pick up the order or if it should be delivered, which is followed by a delivery form that the customer has to fill in if they want it to be delivered. The system displays a modal with the order total. The customer requests to pay through the Virtual Card Service (PayPal). The transaction takes place on their system and once complete we save the transaction details on our system. The use case ends when the system displays a Payment successful popup.
Pre-Conditions:	The customer should have product items in their cart.

Expected Functionality:	 The system should update the <u>Delivery Entity</u> with the delivery address of the customer The system should calculate and display the amount due The system should load and redirect the customer to the payment gateway, in order to complete the transaction. The system should retrieve the transaction details from the VCS system The system should update the <u>Sale Entity</u> with the transaction information
	 The system should record the transaction. The system should display a message informing the customer that the transaction was successful

Test	Test	t Data	Expecte	d Result	Actua	l Result
Scenario	Correct Test Data	Incorrect Test Data	Correct Test Data	Incorrect Test Data	Correct Test Data	Incorre Test Data
1. The customer clicks the 'Checkout' button			The system checks to see if the customer is logged in. The system prompts the customer to enter delivery details (Alt)	Not logged in- please sign in or register as a new user prompt shown by the system.		
3. The customer selects the delivery option and clicks the proceed button.	One radio button option selected, proceed button clicked.		The system displays delivery form. The system uses a SQL insert to capture the AddressID using the Delivery table and generates a new deliveryID, the system also uses SQL Update to update the SaleOrder_ReceiveType in the Sale Entity. The system validates the information using a	The customer selects the cancel button Terminate Use Case The customer chooses Collection radio button. The system uses SQL Update query to update SaleOrder_ReceiveType to 'collection' in the Sale Entity.		
			SQL read from the Delivery entity Address entity			

			T		 ·
		,	Province entity		l
'	'		City entity Suburb positive		
	'	!	Suburb entity		
4. The customer enters their address information and click the proceed button.	4 textboxes filled in, Two dropdown box items selected, Proceed button clicked	Not all information valid. Not all required information provided.	The system captures and validates the information to the Address entity, Province entity, Suburb entity and City entity The system displays a message box with the amount due and prompts the user to select a payment method. The system displays a message box with the amount due and prompts the user to select a payment method.	System fails to update the delivery details- There was an error while updating the delivery details message.	
5. The user clicks the 'Checkout with PayPal' button			The system loads the payment gateway with the amount due.		
6. The customer makes a payment through the Virtual card services (PayPal).	Select Virtual Card Service to checkout		The system uses a SQL insert to record the transaction in the sale table and generates a new SaleID.		
7.Virtual card services: Their system displays the payment			The system uses a SQL insert to record the transaction in the sale table and generates a new SaleID.	System shows message —Payment unsuccessful as there was an error on the VCS system and the payment failed.	

successful		The system displays a		
message		Payment successful		
		message box.		

CONCLUSION

This section contained the test plan for the systems 6 core use cases, test scenarios and procedures, and test data.

13. Hardware & Software Requirements

INTRODUCTION

This section contains descriptions of the required hardware and software for the system to run.

Minimum Requirements:

Computer Hardware		
Hardware Requirements	Requirements	Explanation
Storage	4GB	For a Windows 10 operating system, the minimum requirements will be 4GB. Storage is required to access and store data.
RAM	2GB	Minimum system requirement for 64-bit Windows 10 is 2GB.
CPU	Intel Core i3	Required to execute system instructions.
Input Device	Keyboard & Mouse	None
Output Device	Monitor & Printer	None
Computer Software		
Software Requirements	Requirements	Explanation
Operating System	Microsoft Windows 10 x64	To manage hardware and software resources.
Database Management Server	SQL Server Management studio 2014	Required to operate the system.
Browser	Google Chrome Microsoft Edge	Browser on which system can be accessed.
Development Environment	Visual Studio 2015 (16.10.1) Visual Studio Code (1.5.7)	These environments will be used to develop and maintain the system.

Recommended Requirements:

Computer Hardware				
Hardware Requirements	Requirements	Explanation		
Storage	10GB	This will be used to store the data.		
RAM	4GB RAM	Enough to process the requests in the system.		
CPU	Intel Core i5 or higher	Higher CPU generates better performance.		
Input Device	Keyboard & Mouse	Main input device.		
Output Device	Monitor & Printer	Main output device and		
		printing out the reports.		
Computer Software				
Software Requirements	Requirements	Explanation		
Operating System	HTML5 Compatible	The operating system that		
	operating system. Eg. Microsoft Windows	can support the system.		
Database Management Server	SQL Server Management studio 2014	Required to operate the system.		
Browser	Google Chrome Microsoft Edge	Browser on which system can be accessed.		
Development	Visual Studio 2015	These environments will		
Environment	(16.10.1)	be used to develop and		
	Visual Studio Code (1.5.7)	maintain the system.		

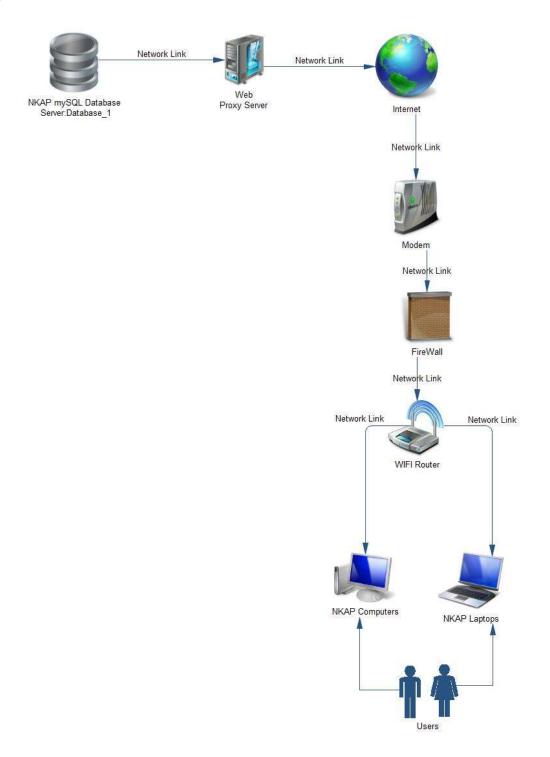
CONCLUSION

This section contained the hardware and software requirements needed to run the system. It specified the minimum requirements for a computer.

14. Network / Web Layout Specifications

INTRODUCTION

In this section we will be looking at the network specifications needed to implement and operate the system.



CONCLUSION

This section included the network specifications and layout. \\

15. Validation

INTRODUCTION

This section will detail the validation of each procedure for each of the requirements in the NKAP Bolting system.

15.1.1. LOGIN SUBSYSTEM

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Login	1.1 Login	Step 1: Request to login to the system.	None	Login	None	None
		Step 2: Show login screen.	None	Login	None	None
		Step 3: Enter login details	None	Login	None	None
		Step 4: User clicks on the log-in button	None	Login	None	None
		Step 5: The system validates the login details	User	None	None	None
		[ALT: Login details are incorrect]				
		Step 6: The user logs in to the system.	None	None	Login()	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Forgot Password	1.2. Forgot Password	Step 1: User clicks on the forgot password link.	None	Login	None	None
		Step 2: Display the Forgot Password screen.	None	Forgot Password	None	None
		Step 3: User enters their email address.	None	Forgot Password	None	None
		Step 4: User clicks on the Submit button. [ALT: user clicks on the Cancel button]	None	Forgot Password	None	None
		Step 5: System validates the email address.	Customer Admin	None	None	None
		Step 6: The system reads the email address of the user.	None	None	None	None
		Step 7: Generate an email with a link.	None	None.	None	None

Step 8: The system	None	Outcome	None	None
notifies the user of		pop-up.		
the email sent to				
them.				

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Reset Password	1.3.	Step 1: User clicks on the Reset Password link.	None	None	None	None
		Step 2: Display Reset Password page.	None	Reset Password	None	None
		Step 3: Enter new password.	None	Reset Password	None	None
		Step 4: The user clicks on submit. [ALT: User clicks on Cancel button]	None	Reset Password	None	None
		Step 5: Validate new password. [ALT: Passwords don't match]	User	Reset Password	None	None
		Step 6: The system saves the new password.	User	None	ResetPassw ord()	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Logout	1.4.	Step 1: User requests to logout	None	The current screen the user logs out from.	None	None
		Step 2: The user clicks on the logout option.			None	None
		Step 3: Display logout confirmation pop-up.	None	The current screen the user logs out from.	None	None
		Step 4: User confirms the logout.	None	The current screen the	None	None

[ALT: User cancels log-out]		user logs out from.		
Step 5: Logout information is added	Audit Trail	None.	None	None
to the audit trail.				
Step 6: The user gets logged out of the system.	None	None	Logout()	None

15.1.2. SUPPLIER SUBSYTEM

Functional Requireme nt	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Add Supplier	2.1	Step 1: Request to add new supplier.	None	Supplier	None	None
		Step 2: Select add supplier button.	None	Supplier	None	None
		Step 3: Display Add Supplier screen.	None	Add Supplier	None	None
		Step 4: Enter supplier details.	None	Add Supplier	None	None
		Step 5: Admin clicks on the add button.	None	Add Supplier	None	None
		[ALT: Admin clicks on the Cancel button]				
		Step 6: The system validates the information	Supplier	None	None	None
		Step 7: System saves the information	Supplier	None	AddSuppli er()	None

Functional Requireme nt	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Search Supplier	2.2.	Step 1: The admin requests to search a supplier.	None	None	None	None
		Step 2: Enter search criteria	None	Supplier	None	None
		Step 3: Validate the search criteria	Supplier	None	SearchSu pplier()	None
		Step 4: Read the suppliers matching the search criteria	Supplier	None	None	None
		Step 5: Display the search results.	None	Supplier	None	None

Functional Requireme nt	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Update Supplier	2.3.	Step 1: Request to update a supplier.	None	None	None	None
		Step 2: Invoke Use Case 2.2 Search Supplier	None	None	None	None
		Step 3: Click on the update button.	None	Supplier	None	None

Step 4: Display Update Supplier screen	Supplier	Update Supplier	None	None
Step 5: Enter new supplier information.	None	Update Supplier	None	None
[Alt: Admin clicks on the Cancel button]				
Step 6: Validate the new information	Supplier	None	None	None
[ALT: Validation unsuccessful]				
Step 7: System saves updated information.	Supplier	None	UpdateSu pplier()	None
[ALT: Couldn't save updated supplier information]				

Functional Requireme nt	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Delete Supplier	2.4.	Step 1: Admin requests to delete a supplier.	None	None	None	None

Step 2: Invoke Use Case 2.2 Search Supplier	None	None	None	None
Step 3: The admin clicks on the Delete button.	None	Supplier	None	None
Step 4: Display the delete modal.	None	Delete Modal	None	None
Step 5: Admin confirms deletion.	None	Delete modal	None	None
[ALT: Admin clicks on the cancel button]				
Step 6: The system deletes the supplier from the system.	Supplier	None	DeleteSup plier()	None
Step 7: The rows get updated in the table.	None	Supplier	None	None

Functional Requireme nt	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Place Supplier Order	2.5.	Step 1: Request to place supplier order.	None	None	None	None
		Step 2: Select the Place	None	Home	None	None

Supplier Order option.				
Step 3: Display the Place Supplier Order screen.	None	Place Supplier Order	None	None
Step 4: Admin selects the supplier and category.	Supplier Category Items	Place Supplier Order	None	None
Step 5: They click on the submit button.	None	Place Supplier Order	None	None
Step 6: Display a list of the product items in that category.	None	Place Supplier Order	None	None
Step 7: Select the product items and quantity.	None	Place Supplier Order	None	None
Step 8: Click on Place Order button.	None	Place Supplier Order	None	None
[ALT: Admin clicks on the Cancel button]				
Step 9: System captures the items.	None	None	None	None
Step 10: The system will save the information.	Supplier OrderLin e	None	PlaceSup plierOrder()	None

Functional Requireme nt	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Receive Supplier Order	2.6.	Step 1: Check supplier invoice matches with items delivered.	None	None	None	None
		Step 2: Admin accepts the delivery order.	None	None	None	None
		Step 3: Request to receive supplier order.	None	None	None	None
		Step 4: Invoke use case 2.2 Search Supplier.	None	None	None	None
		Step 5: Display search results	None	None	None	None
		Step 6: Click on the Receive Order button.	None	Supplier	None	None
		Step 7: Display Receive Supplier Order screen.	None	Receive Supplier Order	None	None
		Step 8: Select supplier order they want.	None	Receive Supplier Order	None	None
		Step 9: Enter relevant order and date information	None	Receive Supplier Order	None	None

Step 10: Click on the Add button	None	Receive Supplier Order	None	None
Step 11: Display Add invoice items modal	None	Add Invoice Items	None	None
Step 12: Enter the valid information.	None	Add Invoice Items	None	None
Step 13: Click on the add button	None	Add Invoice Items	None	None
Step 14: System save and update information	SupplierI nvoiceLin e	None	None	None
Step 15: Click on save button.	None	Receive Supplier Order	None	None
Step 16: The information is saved.	Supplier SupplierI nvoice	None	None	None
Step 17: The system updates the quantity on hand.	ProductIt em	None	None	None

Functional Requireme nt	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Capture Supplier Payment	2.7	Step 1: Request to capture payment	None	None	None	None

Step 2: Invoke Use case 2.2 search Supplier	None	None	None	None
Step 3: Display the search results.	None	Supplier	None	None
Step 4: Click on the Capture Payment button.	None	Supplier	None	None
Step 5: Display Capture Payment screen.	None	Capture Payment	None	None
Step 6: Add amount paid to supplier	None	Capture Payment	None	None
Step 7: Validate information	None	None	None	None
Step 8: Calculate the supplier balance	Supplier	None	None	None
Step 9: Add information to the system.	Supplier Supplier Payment	None	CapturePa yment()	None

15.1.3. USER SUBSYSTEM

Functional	Use	Narrative Steps	ERD	Screens	Methods	Reports
Requirement	Case		Entities			
	No.					
Register New Customer	3.1	Step 1: Request to Register	None	None	None	None
		Step 2: Click User Icon	None	Homescreen	None	None
		Step 3: Display User Dropdown menu	None	Homescreen	None	None
		Step 4: Click Register button on login screen	None	Login	None	None
		Step 5: Display Register new customer screen	None	Register New Customer	None	None
		Step 6: Provide Customer Details	None	Register New Customer	None	None
		Step 7: Validate Customer Details [ALT] Validation failed	None	None [ALT] Validation Error Modal	Validate Customer details()	None
		Step 8: Display User Details Screen	None	User Details	None	None
		Step 9: Provide User Details	None	Register Customer	None	None
		Step 10: Validate User Details [ALT] Validation failed	None	None [ALT] Validation Error Modal	Validate User Details()	None
		Step 11: Display Confirmation Screen	None	Confirm Details	None	None
		Step 12: Confirm details [ALT] Details not confirmed	None	Confirm Details	None	None
		Step 13: Save Customer and User details in database	Customer User	None	Register Customer()	None
		Step 14: Display success modal [ALT] Registration failed	None	Success modal [ALT] Error Modal	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Search Customer	3.2	Step 1: Request to search for customer	None	None	None	None
		Step 2: Click Search Customer dropdown menu item.	None	Homescreen	None	None
		Step 3: Display search customer screen	None	Search Customer	None	None
		Step 4: Provide Search Criteria	None	Search Customer	None	None
		Step 5: Search for matching record in database	Customer	None	Search Customer()	None
		Step 6: Display matching records in datagridview [ALT] No matching records found	None	Search Customer [ALT] Error Modal	None	None
		Step 7: Select Customer to View	None	Search Customer	None	None
		Step 8: Display View Customer Screen.	None	View Customer	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Update Customer	3.3	Step 1: Customer wants to update details	None	None	None	None
		Step 2: Admin requests customer name	None	None	None	None
		Step 3: Customer Provides name	None	None	None	None
		Step 4: Admin searches for customer	Customer	Search Customer	Search Customer()	None
		Step 5: System invokes Use Case 3.2 Search Customer	None	None	None	None
		Step 6: Admin clicks update button	None	Search Customer	None	None
		Step 7: Display Update customer screen	None	Update Customer	None	None
		Step 8: Admin request customer information	None	None	None	None
		Step 9: Customer Provides information	None	None	None	None
		Step 10: Admin provides updated customer information	None	None	None	None
		Step 11: Validate Customer Information [ALT] Validation failed	None	None [ALT] Validation error modal	Validate Updated details()	None
		Step 12: Display Confirm Update modal	None	Confirm Update modal	None	None
		Step 13: Admin request confirmation from customer	None	None	None	None
		Step 14: Customer provides confirmation	None	None	None	None

[ALT] Update not confirmed				
Step 15: Admin clicks confirm button	None	Confirm Update modal	None	None
Step 16: Update Customer details in database	Customer	None	Update Customer()	None
Step 17: Display success modal [ALT] Update failed	None	Success modal [ALT] Error modal	None	None
Step 18: Admin informs customer that information was updated	None	None	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Delete Customer	3.4	Step 1: Customer wants to delete account	None	None	None	None
		Step 2: Admin requests customer name	None	None	None	None
		Step 3: Customer Provides name	None	None	None	None
		Step 4: Admin searches for customer	Customer	Search Customer	Search Customer()	None
		Step 5: System invokes Use Case 3.2 Search Customer	None	None	None	None
		Step 6: Admin clicks delete button	None	Search Customer	None	None
		Step 7: Display confirm deletion modal	None	Confirm Deletion Modal	None	None
		Step 8: Admin request confirmation from customer	None	None	None	None
			Step 9 Customer provides confirmation [ALT] Update not confirmed	None	None	None
		Step 10: Admin clicks confirm button	None	Confirm Deletion modal	None	None
		Step 11: Delete Customer details from database	Customer User	None	Delete Customer()	None
		Step 12: Display success modal [ALT] Update failed	None	Success Modal [ALT] Error modal	None	None
		Step 13: Admin informs customer that their account was deleted	None	None	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
View Profile	3.5	Step 1: Customer Request to view profile	None	None	None	None
		Step 2: Customer clicks User icon then View Profile dropdown menu item	None	Homescreen	None	None
		Step 3: Read Customer information from database	Customer	None	View Profile()	None
		Step 4: Display View Profile [ALT] Error displaying profile	None	View Profile [ALT] Error Modal	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Update Profile	3.6	Step 1: Customer Request to view profile	None	None	None	None
		Step 2: Click User icon then View Profile dropdown menu item	None	Homescreen	None	None
		Step 3: Invoke Use Case 3.5 View Profile	Customer	None	View Profile()	None
		Step 4: Click Update Button	None	View Profile	None	None
		Step 5: Display Update Profile screen	None	Update Profile	None	None
		Step 6: Provided updated profile information	None	Update Profile	None	None
		Step 7: Validate Profile Information [ALT] Validation failed	None	None [ALT] Validation Error Modal	None	None
		Step 8: Display Confirm Update Modal	None	Confirm Update Modal	None	None
		Step 9: Customer Clicks confirm button [ALT] Update not confirmed	None	Confirm Update Modal	None	None
		Step 10: Update customer details in database	Customer	None	Update Customer()	None
		Step 11: Display success modal [ALT] Update failed	None	Success Modal [ALT] Error Modal	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Delete Profile	3.7	Step 1: Customer Request to view profile	None	None	None	None
		Step 2: Click User icon then View Profile dropdown menu item	None	Homescreen	None	None
		Step 3: Invoke Use Case 3.5 View Profile	Customer	None	View Profile()	None
		Step 4: Click Delete Button	None	View Profile	None	None
		Step 5: Display confirmation modal	None	Confirmation modal	None	None
		Step 6: Customer clicks confirm [ALT] Deletion not confirmed	None	Confirmation modal	None	None
		Step 7: Delete Customer from database	Customer User	None	Delete Customer()	None
		Step 8: Display success modal [ALT] Deletion failed	None	None	Success Modal [ALT] Error Modal	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Add User Role	3.8	Step 1: Admin Request to Add User Role	None	None	None	None
		Step 2: Click User Menu Item and Add User Role dropdown menu item	None	Homescreen	None	None
		Step 3: Display Add User Role Screen	None	Add User Role	None	None
		Step 4: Admin Provides User Role information	None	Add User Role	None	None
		Step 5: Validate User Role details	None	None	Validate User Role()	None
		Step 6: Display confirmation modal	None	Confirmation Modal	None	None
		Step 7: Admin clicks confirm button [ALT] Add User Role not confirmed	None	Confirmation Modal	None	None
	Step 8: Add User Role to database	User Role	None	Add UserRole()	None	
		Step 9: Display Success Modal [ALT] Add User Role failed	None	Success Modal [ALT] Error Modal	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Search User Role	3.9	Step 1: Admin Request to Search User Role	None	None	None	None
		Step 2: Click User Menu Item and Searcg User Role dropdown menu item	None	Homescreen	None	None
		Step 3: Display Search User Role Screen	None	Search User Role	None	None
		Step 4: Admin Provides Search Criteria	None	Search User Role	None	None
		Step 5: Search for matching record in database	UserRole	None	Search UserRole ()	None
	Step 6: Display matching records in datagridview [ALT] No matching records found	None	Search User Role [ALT] Error Modal	None	None	
		Step 7: Select User Role to View	None	Search User Role	None	None
		Step 8: Display User Role	UserRole	View User Role	None	None

Functional	Use	Narrative Steps	ERD	Screens	Methods	Reports
Requirement	Case		Entities			
	No.					
Update User Role	3.10	Step 1: Admin Request to Update User Role	None	None	None	None
		Step 2: Click User Menu Item and Search User Role dropdown menu item	None	Homescreen	None	None
		Step 3: Invoke Use Case 3.9 Search Role	None	None	Search User Role()	None
		Step 4: The Admin clicks the update button	None	None	None	None
		Step 5: Display Update User Role Screen	None	Update UserRole	None	None
		Step 6: Provide updated UserRole information	None	Update UserRole	None	None
		Step 7: Validate UserRole Information [ALT] Validation failed	None	None [ALT] Validation Error Modal	None	None
		Step 8: Display Confirm Update Modal	None	Confirm Update Modal	None	None
		Step 9: Admin Clicks confirm button [ALT] Update not confirmed	None	Confirm Update Modal	None	None
		Step 10: Update UserRole details in database	UserRole	None	Update UserRole()	None
		Step 11: Display success modal [ALT] Update failed	None	Success Modal [ALT] Error Modal	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Delete User Role	3.11	Step 1: Admin Request to Delete User Role	None	None	None	None
		Step 2: Click User Menu Item and Search User Role dropdown menu item	None	Homescreen	None	None
		Step 3: Invoke Use Case 3.9 Search Role	None	None	Search User Role()	None
		Step 4: The Admin clicks the delete button	None	None	None	None
		Step 5: Display confirmation modal	None	Confirmation modal	None	None
		Step 6: Admin clicks confirm [ALT] Deletion not confirmed	None	Confirmation modal	None	None
		Step 7: Delete UserRole from database	UserRole	None	Delete User Role()	None
		Step 8: Display success modal [ALT] Deletion failed	None	None	Success Modal [ALT] Error Modal	None

15.1.4. EMPLOYEE SUBSYSTEM

Functional	Use	Narrative Steps	ERD	Screens	Methods	Reports
Requirement	Case No.		Entities	N.I.	NI	NI
Create Delivery Shift	4.1	Step 1: Admin wants to add Delivery Shift	None	None	None	None
		Step 2: admin will click on the "Create Delivery Shift" menu item.	None	Home screen	None	None
		Step 3: Display Create Delivery Shift Modal	None	Create Delivery Shift	None	None
		Step 4: Admin Provides Delivery Shift details	None	Create Delivery Shift	None	None
		Step 5: Admin clicks Add button [ALT] Admin clicks cancel	None	Create Delivery Shift	None	None
		Step 6: Display Confirm to Proceed modal	None	Proceed to Confirm Modal	None	None
		Step 7: Admin clicks Yes button [ALT] Admin clicks No button	None	Proceed to Confirm Modal	None	None
		Step 8: Validate Delivery shift information [ALT] Validation failed.	Date Time Shift	None [ALT[Validation unsuccessful modal	Validate Delivery Shift Details()	None
		Step 9: Save Delivery Shift Information	Date Time Shift	None	Save Delivery Shift Details()	None
		Step 10: Display success modal	None	Success Modal	None	None
		[ALT] Delivery Shift creation failed		[ALT] Unsuccessful modal		

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Search Delivery Shift	4.2	Step 1: Admin wants to search for a delivery shift.	None	None	None	None
		Step 2: Admin clicks Search Delivery Shift	None	Home screen	None	None
		Step 3: Display Search Delivery Shift Screen	None	Search Delivery Shift	None	None
		Step 4: Admin inputs search criteria and clicks search icon.	None	Search Delivery Shift	None	None
		Step 5: Validate Search Criteria [ALT] Error Searching for matching record	Date	None [ALT] Error Modal	Validate Search()	None
		Step 6: Read Delivery Shift details from Database	Shift Date Time Employee- Shift	None	Search Employee()	None
		Step 7: Display Search Results in a table	None	Search Delivery Shift	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Update Delivery Shift	4.3	Step 1: Admin wants to search then Update Delivery Shift	None	Search Delivery Shift	None	None
		Step 2: System Invokes Use Case 4.2 Search Delivery Shift	None	Search Delivery Shift	Search Delivery Shift()	None
		Step 3: Display Update Delivery Shift Modal	None	Update Delivery Shift Modal	None	None
		Step 4: Provide Delivery Shift Information & clicks 'Done' button	None	Update Delivery Shift Modal	None	None
		[ALT] Admin Clicks cancel button				
		Step 5: Validate Updated Information	Time Date	None [ALT] Validation	Validate Delivery Details()	None
		[ALT] Validation failed		Unsuccessful Modal		
		Step 6: Update Delivery Shift in Database	Time Date	None	Update Delivery Shift()	None
		Step 7: Display Success modal	None	Success modal	None	None
		[ALT] Update Failed		[ALT] Error Modal		

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Delete Delivery Shift	4.4	Step 1: The admin wans to delete a Delivery Shift	None	None	None	None
		Step 2: System Invokes Use Case 4.2 Search Delivery Shift	None	Search Delivery Screen	Search Delivery Shift()	None
		Step 3: Admin selects delete option	None	Search Delivery Screen	None	None
		Step 4: Display Confirm to Proceed modal	None	Confirm to Proceed Modal	None	None
		Step 5: Admin clicks Yes button [ALT] Admin clicks No button	None	Confirm to Proceed Modal [ALT] Search Delivery Screen	None	None
		Step 6: Delete Delivery Shift from database	Shift	None	Delete Delivery Shift()	None
	Step 7: Remove Delivery Shift from Search Delivery Shift table	None	Search Delivery Shift	None	None	
		Step 8: Display success modal	None	Success Modal	None	None
		[ALT] Deletion Failed		[ALT] Unsuccessful Modal		

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Add Employee	4.5	Step 1: Admin requests to add a new employee	None	None	None	None
		Step 2: Admin clicks "Employee" menu item on the navigation bar.	None	Home screen	None	None
		Step 3: The system displays the Employee dropdown menu item	None	Home screen	None	None
		Step 4: The customer clicks the "Add Employee" dropdown menu item.	None	Home screen	None	None
		Step 5: The system responds by loading the "Add Employee" modal	None	Add Employee Modal	None	None
		Step 6: The employee communicates their details to the admin	None	None	None	None
		Step 7: Admin enters employee details and clicks the next button	None	Add Employee Modal	None	None
		Step 8: The system validates the captured employee information	Employee	None	Validate Employee Details ()	None
		[Alt Step 8]: The system fails to validate the inputted information and displays a Validation Unsuccessful Modal.		Validation Unsuccessful Modal		
		Step 9: The system enables and	None	Add Employee Modal	None	None

disables controls in				
the modal				
Step 10: The admin enters the temporary login details for the employee	None	Add Employee Modal	None	None
Step 11: The system validates the captured information	User	None	Validate Employee Details ()	None
[Alt Step 11]: The system fails to validate the inputted information and displays a Validation Unsuccessful Modal.		Validation Unsuccessful Modal		
Step 12: The system displays a modal for the admin to confirm the addition of the employee information	None	Confirm to proceed modal	None	None
Step 13: The admin selects the "Yes" button [Alt Step 13]: The admin clicks on the "No" button which makes the admin	None	Confirm to proceed modal Add Employee Modal	None	None
Return to step 7 Step 14: The system uses a SQL insert query to save the new employee information in the Employee and User Tables	Employee User	None	Add Employee ()	None
Step 15: The system saves the new employee details in the relevant entities and displays a success modal	Employee User	Success modal	None	None

[Alt Step 15]: The	Unsuccessful	
new employee could	modal	
not be added to the		
system. The system		
will display the		
unsuccessful modal		

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Search Employee	4.6	Step 1: Admin wants to search for an employee	None	None	None	None
		Step 2: Admin clicks Search Employee	None	Home screen	None	None
		Step 3: Display Search Employee Screen	None	Search Employee	None	None
		Step 4: Input search criteria & clicks Search button	None	Search Employee	None	None
		Step 5: Validate Search Criteria [ALT] No matches found.	Employee	None [ALT] Error Modal	Validate Search()	None
		Step 6: Read Employee Details from Database	Employee	None	Search Employee()	None
		Step 7: Display Search Results in a table	None	Search Employee	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
	4.7.	Step 1: Admin wants to search then Update Employee	None	Search Employee	None	None
		Step 2: System Invokes Use Case 4.6. Search Employee	None	Search Employee	Search Employee()	None
		Step 3: Display Update Employee Modal	None	Update Employee Modal	None	None
		Step 4: Provide Employee Information & clicks 'Done' button	None	Update Employee Modal [ALT] Search	None	None
		[ALT] Admin Clicks Cancel button		Employee		
		Step 5: Validate Updated Information [ALT] Validation failed	Employee	None [ALT] Validation Unsuccessful	Validate Employee Details()	None
				Modal		
		Step 6: Update Delivery Shift in Database	Employee	None	Update Employee()	None
		Step 7: Display Success modal	None	Success modal	None	None
		[ALT] Update Failed		[ALT] Unsuccessful Modal		

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Delete 4.8. Delivery Shift	4.8.	Step 1: The admin requests to delete an Employee	None	Search Employee	None	None
		Step 2: System Invokes Use Case 4.6. Search Employee	None	Search Employee	Search Employee()	None
		Step 3: Admin clicks on delete option	None	Search Employee	None	None
		Step 4: Display Confirm to Proceed modal	None	Confirm to Proceed Modal	None	None
		Step 5: Admin clicks Yes button [ALT] Admin clicks No button	None	Confirm to Proceed Modal [ALT] Search Employee Screen	None	None
		Step 6: Delete Employee from database	Employee	None	Delete Employee()	None
		Step 7: Remove Employee from Search Delivery Shift table	None	Search Employee Screen	None	None
		Step 8: Display success modal	None	Success Modal	None	None
		[ALT] Deletion Failed		[ALT] Unsuccessful Modal		

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Assign Order for Delivery/Cour ier		Step 1: The admin wants to assign an order to be delivered to the customer	None	None	None	None
		Step 2: The admin clicks on the "Assign Order Delivery" sub menu item.	None	Home Screen	None	None
		Step 3: Display Assign Order Delivery Modal	None	Assign Order Delivery Modal	None	None
		Step 4: The admin clicks on the Courier button.	None	Assign Order Delivery Modal	None	None
		[ALT delivery] The admin clicks on the Delivery Button	None	None	None	None
		Step 5: Display Courier Modal	None	Courier Modal	None	None
		[ALT delivery] Display Delivery Modal	None	[ALT delivery] Delivery Modal	None	None
		Step 6: Admin wants to search for the courier.	None	None	None	None
		[ALT courier] Admin clicks on Cancel Button.	None	[ALT Courier] Courier Modal to Assign Order Delivery	None	None
		[a. ALT delivery] Admin clicks employee name from dropdown & clicks 'Done' button.	None	[a. ALT delivery] Delivery Modal	None	None
		[b. ALT delivery] Admin clicks Cancel button.	None	[b. ALT delivery] Delivery Modal to	None	None

		Assign Order Delivery		
Step 7: System Invokes Use Case 8.4. Search Courier	None	Courier Modal	Search Courier ()	None
[ALT delivery] Display Confirmation Modal	None	[ALT delivery] Confirm to Proceed Modal	None	None
Step 8: Admin clicks email address of chosen courier.	None	Courier Modal	None	None
[a. ALT delivery] Admin clicks on Yes button.	None	[a. ALT delivery] Confirm to Proceed Modal	None	None
[b. ALT delivery] Admin clicks on No button.	None	[b. ALT delivery] Confirm to Proceed Modal to Assign Order Delivery.	None	None
Step 9: Directs admin to Web browser to compose email.	None	Web Browser Email Web Application Screen	None	None
[ALT delivery] Save delivery information in database.	[ALT delivery] Delivery Sale Employee Employee- Shift	[ALT delivery] None	None	None
Step 10: Admin enters order information for delivery.	None	Web Browser Email Web Application Screen	None	None
[a. ALT delivery] Display Success Modal	None	[a. ALT delivery] Success Modal	None	None

[b. ALT delivery] Order could not be assigned for delivery.	None	[b. ALT delivery] Error Modal	None	None
[c. ALT delivery] Employee cannot be assigned to deliver order as they are not assigned to an appropriate shift. Display Error Modal.	None	[c. ALT delivery] Error Modal	None	None
Step 11: Admin sends email & navigates back to Courier modal.	None	Courier modal	None	None
Step 12: Admin clicks 'Done' button [ALT] Admin Clicks Cancel button	None	Courier Modal [ALT courier] Assign Order Delivery Modal	None	None
Step 13: Display message	None	Attention Modal	None	None
Step 14: Admin clicks on the OK button. [ALT] Admin clicks	None	Attention Modal [ALT courier] Courier		
on the Cancel button. Step 15: Save courier information	Delivery Sale	modal None	Assign Courier	None
in the database. Step 16: Display success modal	None	Success Modal	Delivery () None	None
[ALT] Saving of information was unsuccessful.		[ALT courier] Error Modal		

Functional	Use	Narrative Steps	ERD	Screens	Methods	Reports
Requirement	Case No.		Entities			
Assign Delivery Shift	4.10.	Step 1: Admin wants to assign a delivery shift an employee	None	None	None	None
		Step 2: System Invokes Use Case 4.2. Search Delivery Shift	None	Search Delivery Shift	Search Delivery Shift()	None
		Step 3: Admin clicks on Assign button	None	Search Delivery Shift	None	None
		Step 4: Display Assign Delivery Shift Modal.	None	Assign Delivery Shift Modal	None	None
		Step 5: Admin selects employee name.	None	Assign Delivery Shift Modal	None	None
		Step 6: Admin clicks Proceed button [ALT] Admin clicks Cancel button	None	Assign Delivery Shift Modal [ALT] Home	None	None
		Cancer batton		screen		
		Step 7: Display Confirm to proceed modal	None	Confirm To Proceed modal	None	None
		Step 8: Admin clicks Yes button	None	Confirm to Proceed	None	None
		[ALT] Admin clicks No button		[ALT] Assign Delivery Shift Modal		
		Step 9: Save Employee details in database	Employee- Shift	None	Assign Employee()	None
		Step 10: Display success modal	None	Success Modal	None	None
		[ALT] Failed to Assign Employee to delivery shift.		[ALT] Unsuccessful Modal		

Functional	Use	Narrative Steps	ERD	Screens	Methods	Reports
Requirement	Case No.	Ctoro 4 : A dissipa	Entities	Nege	Nega	None
Cancel Delivery Shift	4.11.	Step 1: Admin wants to cancel a delivery shift that was assigned to an employee	None	None	None	None
		Step 2: System Invokes Use Case 4.2. Search Delivery Shift	None	Search Delivery Shift	Search Delivery Shift()	None
		Step 3: Admin clicks on Cancel button	None	Search Delivery Shift	None	None
		Step 4: Display Confirm Cancellation modal	None	Confirm To Proceed modal	None	None
		Step 5: Admin clicks Yes button	None	Confirm to Proceed	None	None
		[ALT] Admin clicks No button		[ALT] Assign Delivery Shift Modal		
		Step 6: Remove Employee details in database	Employee- Shift	None	Cancel Employee Shift()	None
		Step 7: Display success modal	None	Success Modal	None	None
		[ALT] Failed to remove employee information from delivery shift.		[ALT] Cannot be Cancelled Modal		
		Step 8: Remove employee name in search results table	None	Search Delivery Shift	None	None

15.1.5. ADMIN SUBSYSTEM

16. Function al Require ment	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Search Online Sales	5.1	Step 1: An employee/ admin requests to search for an online sale	None	None	None	None
		Step 2: An employee/ admin Clicks Administration Back Office Menu item and clicks Search Online Sales dropdown menu item	None	Homescreen	None	None
		Step 3: Display Search Online Sales Screen	None	Search Online Sales	None	None
		Step 4: An employee/ admin selects a filter option	None	Search Online Sales	None	None
		Step 5: Retrieve matching sale records from database	Sale	None	Search Online Sale()	None
		Step 6: Display matching sale records in datagrid view	None	Online Sales	None	None
		Step 7: Employee/Admin selects sale to view	None	Online Sales	None	None
		Step 8: Display View Online Sale Screen	Sale	View Sale	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Pack Order	5.2	Step 1: An employee requests to pack an order	None	None	None	None
		Step 2: The employee clicks Administration Back Office Menu item and clicks Pack Order dropdown menu item	None	Homescreen	None	None
		Step 3: Invoke Use Case 5.1 Search Online Sales	None	None	Search Online Sale()	None
		Step 4: Employee clicks Pack Order Button	None	View Sale	None	None
		Step 5: Display Pack Order Screen	None	Pack Order	None	None
		Step 6: Employee provides Pack Order Details [ALT] Employee clicks cancel	None	Pack Order	None	None
		Step 7: System Updates Order Status to "Ready for Collection" [ALT] Order Status updated to "Ready for Delivery" (Skip to Step 9)	Sale	None	Update Order Status()	None
		Step 8: System Notifies customer that Order is ready for collection	None	None	None	None
		Step 9: Display success modal [ALT] Pack Order process failed	None	Success modal [ALT] Error modal	None	None

Functional	Use	Narrative Steps	ERD	Screens	Methods	Reports
Requirement	Case		Entities			
Do Stock- Take	No. 5.3	Step 1: An admin requests to do stock-take	None	None	None	None
		Step 2: The Admin clicks Administration Back Office Menu item and clicks Do Stock- Take dropdown menu item	None	Homescreen	None	None
		Step 3: Invoke Use Case 9.10 Search Product	None	None	Search Product()	None
		Step 4: Admin clicks Do-Stock Take Button	None	Products	None	None
		Step 5: Display Do Stock-Take Screen	None	Stock-Take	None	None
		Step 6: Admin provides Stock-Take Details [ALT] Admin clicks cancel button	None	Stock-Take	None	None
		Step 7: Validate Stock Take details [ALT] Validation Failed	None	None [ALT] Validation Error Modal	Validate Stock Take Details()	None
		Step 8: Display confirmation modal	None	Confirmation modal	None	None
		Step 9: Admin clicks confirm button	None	Confirmation modal	None	None
		Step 10: Save Stock Take Details	Stock Take ProductItem - StockTake	None	Save StockTake Details()	None
		Step 11: Update Quantity on hand	ProductItem	None	Update Quantity on Hand()	None
		Step 12: Display success modal [ALT] Stock Take failed	None	Success modal [ALT] Error Modal	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
View Delivery Shift Schedule	5.4	Step 1: An employee requests to view the delivery shift schedule	None	None	None	None
		Step 2: The employee clicks Administration Back Office Menu item and clicks View Delivery Shift Schedule dropdown menu item	None	Homescreen	None	None
		Step 3: System reads the shift information from the database	Shift EmployeeS hift	None	Read Shift Schedule Information ()	None
		Step 4: Display Delivery Shift Schedule [ALT] Error displaying Delivery Shift Schedule	None	Schedule	None	None

Functional Requirement	Use Case	Narrative Steps	ERD Entities	Screens	Methods	Reports
rtoquiromoni.	No.					
Write-off Stock	5.5	Step 1: An admin requests to Write-Off Stock	None	None	None	None
		Step 2: The admin clicks Administration Back Office Menu item and clicks Write-Off Stock dropdown menu item	None	Homescreen	None	None
		Step 3: Invoke Use Case 9.10 Search Product	None	None	Search Product()	None
		Step 4: Admin clicks Write-Off Stock Button	None	Products	None	None
		Step 5: Display Do Write-Off Stock Screen	None	Write-Off Stock	None	None
		Step 6: Admin provides Write-Off Details [ALT] Admin clicks cancel button	None	Write-Off Stock	None	None
		Step 7: Validate Write- Off details [ALT] Validation Failed	None	None [ALT] Validation Error Modal	Validate Write-Off Details()	None
		Step 8: Display confirmation modal	None	Confirmation modal	None	None
		Step 9: Admin clicks confirm button	None	Confirmation modal	None	None
		Step 10: Save Write- Off Details	WriteOff ProductItem - WriteOff	None	Save Write-Off Details()	None
			Step 11: Update Quantity on hand	ProductItem	None	Update Quantity on Hand()
		Step 12: Display success modal [ALT] Write-Off failed	None	Success modal [ALT] Error Modal	None	None

16.1.1. CUSTOMER SUBSYSTEM

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
View Product	6.1	Step 1: The customer request to browse through the products by category.	None	None	None	None
		Step 2: The customer clicks on the 'Products' tab on the menu-strip and selects a product category from the sub-items:	None	Homepage	None	None
		Step 3: The system loads the Products screen with the following elements(showing the selected product category):	Productite m entity	Product	searchProd ucts()	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
View Products Detail	6.2	Step 1: The customer requests to search for a product.	None	Product	None	None
		Step 2: The system invokes Use Case 6.1 Search Products.	None	Product	None	None
		Step 3: The customer enters the product search criteria into the search textbox.	Productite m entity	Product	None	None
		Step 4: The customer clicks on the search icon.	None	None	validateSear ch()	None
		Step 5: The system captures and validates the information against the Productitem table	Productite m enity	None	searchCate gory()	None

Step 6: The system uses a read query to read the matching entries from the Product Category entity. The system retrieves	Productite m enity	None	searchCate gory()	None
Step 7: The system displays the retrieved search results.	None	None	searchCate gory()	None
Step 8: The admin clicks on the product they were searching for.	None	None	None	None
Step 9: The system loads the Product Details view with the following elements	None	Product Details	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Search Specials and Promotions	6.3	Step 1: The customer request to browse through the specials and promotions.	None	Promotions	None	None
		Step 2: The customer clicks on the 'Promotions' tab on the menu-strip.	None	Promotions	None	None
		Step 3: The system loads the Products screen with the following elements(showing the selected product category):	ProductIte m entity, Product_S pecial entity, Special entity	Promotions	None	None
		Step 4: The customer enters the product search criteria into the search textbox.	None	None		None
		Step 5: The customer clicks on the search icon.	Productite m enity	None	validateSear ch()	None
		Step 6: The system captures and	Productite m enity	None	searchProd uctItem()	None

validates the information against the Productitem table				
Step 7: The system uses a read query to read the matching entries from the Product Category entity. The system retrieves:	None	None	searchProd uctItem()	None
Step 8: The system displays the retrieved search results.	None	None	searchProd uctItem()	None
Step 9: The admin clicks on the product special they were searching for	None	Promotions	None	None
Step 10: The system loads the SpecialProduct Details view with the following elements(showing the selected product category):	ProductIte m entity, Product_S pecial entity, Special entity	SpecialProd uct Details	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Add item to cart	6.4	Step 1: The customer requests to add a product item to cart.	None	Promotions	None	None
		Step 2: The customer clicks the 'add to cart' button on either the Product view, Product Details view or Promotions view	None	Promotions	confirmAddt oCart()	None
		Step 3: The system loads the Products screen with the following elements(showing the selected product category):	ProductIte m entity, Product_S pecial entity, Special entity	Promotions	None	None

Step 4: The customer enters the product search criteria into the search textbox.	None	None	None	None
Step 5: The customer clicks on the search icon.	CartLine enity	None	addToCart()	None
Step 6: The system captures and validates the information against the Productitem table	Productite m None	None	successfully Added()	None
Step 7: The system uses a read query to read the matching entries from the Product Category entity. The system retrieves:	CartLine entity	None	updateCartl con()	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Search Cart	6.5	Step 1: The customer request to see the cart.	None	None	None	
		Step 2: The customer clicks on the 'Cart' icon on the menu-strip.	None	None	None	
		Step 3: The system loads the Cart view with the following elements	ProductIte m entity, Product_S pecial entity, Special entity	Cart	None	

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Remove Cart Item	6.6	Step 1: The customer request to remove an item from the cart.	None	None	None	
		Step 2: The customer clicks on the 'Remove icon	None	None	None	

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Checkout Order	6.7	Step 1: The customer request to go to checkout.	None	None	None	
		Step 2: The customer clicks the 'Checkout' button located in the Cart view.	None	None	None	
		Step 3: The system checks to see if the customer is logged in	None	Cart	ConfirmLogi n()	
		Step 4: The system prompts the	None	Cart	None	None

customer to enter delivery details.				
Step 5: The system retrieves and removes the data to from CartLine table	None	Cart	None	None
Step 6: The system displays a successfully removed from cart message box	Address entity, Province entity, Suburb entity and City entity	Cart	AddAddress ()	None
Step 7: The system updates the cart icon in the page header. Using the CartLine_Quantity attribute in the CartLine entity to show the amount of items in the cart.	None	Cart	PaymentMe ssage()	None
Step 8: The user select a Virtual Card Service method of choice.	None	Cart	None	None
Step 9: The system loads the payment gateway with the amount due.	None	Cart	None	None
Step 10: Virtual card services: Their system displays the payment successful message	None	Cart	None	None
Step 11: The system records the transaction	None	Cart	None	None
Step 12: The system displays a successfully added to cart message box and generates the invoice.	None	Cart	None	None

16.1.2. REPORTS SUBSYSTEM

Functional	Use Cook No.	Narrative Steps	ERD Entition	Screens	Methods	Reports
Requirement	Case No.	Chan 1. The requirement	Entities	Nego	Nene	Nana
Generate fast-	7.1	Step 1: The requests	None	None	None	None
selling product		to generate a report	None	None	None	None
list		Step 2: The admin clicks on the Reports	None	None	None	None
		menu option				
		Step 3: The system	None	Reports	None	None
		displays the "Reports"	None	Reports	None	None
		screen				
		Step 4: The admin	None	Reports	None	None
		selects the "Fast-				
		selling product list"				
		Step 5: The admin	None	Reports	None	None
		selects the required				
		period.				
		Step 6: The admin	None	Reports	None	None
		clicks the "Generate"				
		button				
		Step 7: The system	ProductIte	None	None	None
		retrieves the required	m			
		information from the	Sale			
		database tables using	SaleLine			
		a SQL Read.				
		Step 8: The system	None	None	None	None
		compiles and				
		generates the retrieved information				
		into the Fast-selling				
		product list.				
		Step 9: The system	None	Reports	None	Fast-selling
		displays the Fast-	140110	Reports	140110	product list
		selling product list in				product not
		the report panel				
	1	the report parier				

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Generate Slow-selling	7.2	Step 1: The requests to generate a report	None	None	None	None
product list		Step 2: The admin clicks on the Reports menu option	None	None	None	None
		Step 3: The system displays the "Reports" screen	None	Reports	None	None

Step 4: The admin selects the "Slow- selling product list"	None	Reports	None	None
Step 5: The admin selects the required period.	None	Reports	None	None
Step 6: The admin clicks the "Generate" button	None	Reports	None	None
Step 7: The system retrieves the required information from the database tables using a SQL Read.	ProductIte m Sale SaleLine	None	None	None
Step 8: The system compiles and generates the retrieved information into the Slow-selling product list.	None	None	None	None
Step 9: The system displays the Slow-selling product list in the report panel	None	Reports	None	Slow-selling product list

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Generate most frequent	7.3	Step 1: The requests to generate a report	None	None	None	None
buyers report		Step 2: The admin clicks on the Reports menu option	None	None	None	None
		Step 3: The system displays the "Reports" screen	None	Reports	None	None
		Step 4: The admin selects the "Most frequent buyer report"	None	Reports	None	None
		Step 5: The admin clicks the "Generate" button	None	None	None	None
		Step 6: The system retrieves the required information from the database entities	Sale Customer	None	None	None

s r a r	Step 7: The system sorts through the retrieved information and filters for the top results by using a count	None	None	None	None
c f	Step 8: The system displays the most requent buyers report	None	None Reports	None	Most frequent buyers report

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Generate most popular	7.4	Step 1: The requests to generate a report	None	None	None	None
location report		Step 2: The admin clicks on the Reports menu option	None	None	None	None
		Step 3: The system displays the "Reports" screen	None	Reports	None	None
		Step 4: The admin selects the "most popular location report"	None	Reports	None	None
		Step 5: The admin clicks the "Generate" button	None	None	None	None
		Step 6: The system retrieves the required information from the database entities	Customer Sale Address Province City Suburb	None	None	None
	Step 7: The system sorts through the retrieved information and filters for the areas which had the most sales in the past month	None	None	None	None	
		Step 8: The system displays the most popular location report	None	Reports	None	Most popular location report

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Generate Weekly Sale	7.5	Step 1: The requests to generate a report	None	None	None	None
Order List		Step 2: The admin clicks on the Reports menu option	None	None	None	None
		Step 3: The system displays the "Reports" screen	None	Reports	None	None
		Step 4: The admin selects the "Weekly Sale Order List"	None	Reports	None	None
		Step 5: The admin clicks the "Generate" button	None	None	None	None
		Step 6: The system retrieves the required information from the database entities	Customer Sale	None	None	None
		Step 7: The system then selects the Sales which have occurred in the past week and puts them in a list with the corresponding customers who made those sale orders also in the list.	None	None	None	None
		Step 8: The system then displays the weekly sale order list to the admin	None	Reports	None	Weekly sale order list

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Generate Monthly Sale	7.6	Step 1: The requests to generate a report	None	None	None	None
Order List		Step 2: The admin clicks on the Reports menu option	None	None	None	None
		Step 3: The system displays the "Reports" screen	None	Reports	None	None

Step 4: The admin selects the "Monthly Sale Order List"	None	Reports	None	None
Step 5: The admin clicks the "Generate" button	None	None	None	None
Step 6: The system retrieves the required information from the database entities	Customer Sale	None	None	None
Step 7: The system then selects the Sales which have occurred in the past month and puts them in a list with the corresponding customers who made those sale orders also in the list.	None	None	None	None
Step 8: The system then displays the monthly sale order list to the admin	None	Reports	None	Monthly sale order list

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Generate	7.7	Step 1: The requests	None	None	None	None
Stock-Level		to generate a report				
Report		Step 2: The admin clicks on the Reports menu option	None	None	None	None
		Step 3: The system displays the "Reports" screen	None	Reports	None	None
		Step 4: The admin selects the "Stock- Level Report"	None	Reports	None	None
		Step 5: The admin clicks the "Generate" button	None	None	None	None
		Step 6: The system retrieves the required	ProductCat egory	None	None	None
		information from the	CategoryTy			
		database entities	ре			

	ProductIte			
	m			
Step 7: The system categorizes each ProductItem into its Category type and each category type into its Product Category.	None	None	None	None
Step 8: The system then displays the Stock-level report with the necessary columns.	None	Reports	None	Stock-Level Report

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Packing Report	7.8	Step 1: The requests to generate a report	None	None	None	None
		Step 2: The admin clicks on the Reports menu option	None	None	None	None
		Step 3: The system displays the "Reports" screen	None	Reports	None	None
		Step 4: The admin selects the Packing Report"	None	Reports	None	None
		Step 5: The admin clicks the "Generate" button	None	None	None	None
		Step 6: The system retrieves the required information from the database entities	Sale OrderStatus	None	None	None
		Step 7: The system sorts the retrieved sales from the first made to the most recent	None	None	None	None
		Step 8: The system displays the list of Sales to the admin as the Packing Report	None	Reports	None	Packing Report

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
•	7.9	Stop 1: The requests		None	None	None
Delivery Report	7.9	Step 1: The requests	None	None	None	None
кероп		to generate a report Step 2: The admin	None	None	None	None
		clicks on the Reports	None	None	None	None
		menu option				
		Step 3: The system	None	Reports	None	None
		displays the "Reports"	None	Reports	None	None
		screen				
		Step 4: The admin	None	Reports	None	None
		selects the Delivery	None	Reports	None	None
		Report"				
		Step 5: The admin	None	None	None	None
		clicks the "Generate"				
		button				
		Step 6: The system	Sale	None	None	None
		retrieves the required	OrderStatus			
		information from the	Delivery			
		database entities,	Address			
		which have the order	Province			
		status of "ready for	City			
		delivery" using a SQL	Suburb			
		Read.	Customer			
		Step 7: The system	None	None	None	None
		then filters through				
		the retrieved sales				
		and sorts them				
		according to				
		SaleOrder_Date.				
		Step 8: The system	None	Reports	None	Delivery
		then displays the list				Report
		of retrieved Sales to				
		the admin in the				
		correct order as the				
		Delivery Report				

16.1.3. DELIVERY SUBSYSTEM

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Search Delivery Status	8.1	Step 1: The employee request to view deliveries.	None	None	None	None
		Step 2: The employee clicks on the 'Delivery' tab from the menu-strip.	None	None	None	None
		Step 3: The system loads the Delivery screen with the following elements:	Delivery entity, Courier entity, Address entity, Province entity, City entity, Suburb entity, OrderStatu s	Delivery	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Accept delivery	8.2	Step 1: The employee request to view deliveries.	None	None	None	None
		Step 2: The employee clicks on the 'Delivery' tab from the menu-strip.	None	Delivery	None	None
		Step 3: The system prompts the customer to accept or update the delivery:	None	Delivery	deliverystat us()	None
		Step 4: The employee clicks the confirm button.	None	Delivery	None	None
		Step 5: The system updates all the relevant information, including the OrderStatus icon in the DeliveryStatus table.	None	Delivery	None	None

Step 6: The system	None	None	acceptedMe	None
displays a message			ssage()	
box that informs the				
employee that the				
delivery has been				
accepted.				

Functional	Use	Narrative Steps	ERD	Screens	Methods	Reports
Requirement	Case No.		Entities			
Add Courier	8.3	Step 1: Request to	None	None	None	None
		add courier				
		Step 2: Click on Add	None	Home	None	None
		Courier		Screen		
		Step 3: Display Add	None	Add Courier	None	None
		Courier modal		Modal		
		Step 4: Enter	None	Add Courier	None	None
		courier details		Modal		
		Step 5: Click on Add	None	Add Courier	None	None
		button.		Modal		
		[Al T. Admin aliaka				
		[ALT: Admin clicks on Cancel button]		[ALT: Add		
		on Cancer buttorij		Courier		
				Modal to		
				Home		
				screen]		
		Step 6: Display	None	Confirmatio	None	None
		confirmation modal		n modal		
		Step 7: Select Yes	None	Confirmatio	None	None
		button.		n modal		
		[ALT: Admin selects		[ALT:		
		No button]		Confirmatio		
		i vo battorij		n Modal to		
				Courier		
				Modal]		
		Step 8: Validate	Courier	None	Validate	None
		courier details			Courier	
					Details()	
		[ALT: Validation was		[ALT:	, v	
		unsuccessful.]		Validation		
				Unsuccessf		
				ul Modal]		
	Step 9: Create new	Courier	None	AddCourier(None	
		courier in the)	
		system.				
		Step 10: Courier	None	Success	None	None
		successfully added.		modal		

[ALT: Courier not	[ALT:	
successfully added]	Unsuccessf	
	ul Modal]	

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Search Courier	8.4	Step 1: Request to search for a courier.	None	None	None	None
		Step 2: Click on the search courier option.	None	Home Screen	None	None
		Step 3: Display Search courier	None	Search Courier Screen	None	None
		Step 4: Enter the search criteria.	None	Search Courier Screen	None	None
		Step 5: Validate the information.	Courier	None	Validate Courier Name()	None
		[ALT: Validation failed]		[ALT: Error Modal]		
		Step 6: Read the couriers that match the search criteria	Courier	None	Search Courier()	None
		Step 7: Show the search results.	None	Search Courier Screen	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Update courier	8.5	Step 1: Request to update courier details.	None	None	None	None
		Step 2: Invoke Use Case 8.4 Search Courier & select Update option	None	Search Courier Screen	Search Courier()	None
		Step 3: Display Update Courier Modal	None	Update Courier Modal	None	None
		Step 4: Enter new information.	None	Update Courier Modal	None	None

[ALT: Admin clicks on the Cancel button]				
Step 5: Validate information on the courier.	Courier	None	Validate Courier Details()	None
[ALT: Validation failed]		[ALT: Validation Unsuccessf ul Modal]		
Step 6: Save the updated information.	Courier	None	UpdateCour ier()	None
Step 7: Display success modal.	None	Success Modal	None	None
[ALT: Information could not be added to the system.]		[ALT: Unsuccessf ul Modal		

16.1.4. PRODUCTS SUBSYSTEM

17. Function	Use	Narrative Steps	ERD	Screens	Methods	Reports
al	Case No.		Entities			
Require ment						
Add Product	9.1	Step 1: The admin	None	Products	None	None
Category		requests to add a		Screen		
		new product				
		category				
		Step 2: The admin	None	Products	None	None
		clicks on the "Add" button in the		Screen		
		Products Screen.				
		Step 3: The system	None	Add Product	None	None
		then displays the		category		
		Add Product		modal		
		category modal	None	Add Product	None	None
		Step 4: The admin enters the specified	None	category	inone	None
		product category's		modal		
		details.				
		Step 5: The admin	None	Add Product	None	None
		clicks on the "Add"		category		
		button		modal		
		[ALT] The user				
		clicks on "cancel"				
		which terminates				
		the use case	D 1 10 1	N	.	
		Step 6: The system captures the product	ProductCat	None	None	None
		captures the product	egory			
		and validates it				
		[ALT] The provided				
		product category details were not in				
		the right format; the				
		system will outline				
		the place where the				
		validation error				
		occurred.	D. 1. (0.)	NI.	NI.	NI
		Step 7: The system	ProductCat	None	None	None
		saves the captured information in the	egory			
		ProductCategory				
		Table				
		[[ALT] Add		
		[ALT] The system		Product		
	<u> </u>	cannot save the		Category	<u> </u>	

l ne	ew product	Error	
ca	itegory; the	MODAL	
sy	stem then		
di	splays a modal		
te	lling the admin		
th	at the addition		
Wa	as unsuccessful.		

Functional	Use	Narrative Steps	ERD	Screens	Methods	Reports
Requirement	Case No.	Harrativo Otopo	Entities	Corocno	motriodo	Roporto
Search Product Category	9.2	Step1: The admin requests to search for a product category	None	None	None	None
		Step 2: The admin clicks on the "Products" screen.	None	None	None	None
		Step 3: The system responds by loading the "Products" screen	None	Products	None	None
		Step 4: The admin selects the "Product Category" radio button.	None	Products	None	None
		Step 5: The admin enters the product category search criteria	None	Products	None	None
		Step 6: The admin clicks on the search icon.	None	Products	None	None
		Step 7: The system captures and validates the information against the Product Category table.	ProductCat egory	None	None	None
		[ALT]: The validation fails, and the admin is prompted to enter a new search query				
		Step 8: The system retrieves the	ProductCat egory	None	None	None

matching entries in the ProductCategory Table				
Step 9: The system displays the retrieved search results in a table.	None	Products	None	None
Step 10: The admin selects the product category	None	Products	None	None

Functional	Use	Narrative Steps	ERD	Screens	Methods	Reports
Requirement	Case No.		Entities			
Update Product Category	9.3	Step 1: The admin requests to update a Product Category	None	None	None	None
		Step 2: The system invokes UC 9.2 "Search Product Category"	ProductCat egory	Products	None	None
		Step 3: The admin clicks on the update button	None	Products	None	None
		Step 4: The system responds by loading the "Update Product Category" modal	None	Update Product Category modal	None	None
		Step 5: The admin edits the Product Category	None	Update Product Category modal	None	None
		Step 6: The admin clicks the "Update" button [ALT] The admin clicks on the "Cancel" button which terminates the use case	None	Update Product Category modal	None	None
		Step 7: The system captures and validates the changes made to the product category entered by the admin against the	ProductCat egory	None	None	None

product category table.				
[ALT] The validation of the entered information fails; the system will outline the information that does not meet the validation requirements				
Step 8: The system updates the ProductCategory table	ProductCat egory	None	None	None
[ALT] The system will display a modal stating the update was unsuccessful		[ALT] Update Product Category Error Modal		

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Delete Product Category	9.4	Step 1: The admin requests to delete a Product Category	None	None	None	None
		Step 2: The system invokes UC 9.2 "Search Product Category"	ProductCat egory	Products	None	None
		Step 3: The admin clicks on the delete button	None	Products	None	None
		Step 4: The system displays the delete product category modal	None	Delete Product Category modal	None	None
		Step 5: The admin clicks on the "Yes" button	None	Delete Product Category modal	None	None
		[ALT] The admin clicks on the "No" button and the use case is terminated.				
		Step 6: The system removes the selected product category from the	ProductCat egory	None	None	None

product category entity.				
[ALT] The system will display a modal informing the admin that the deletion of the product category was unsuccessful.				
Step 7: The system removes the deleted product category from the table in the products screen.	None	None	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Add Category Type	9.5	Step 1: The admin requests to add a new category type	None	None	None	None
		Step 2: The admin clicks on the "Add" button in the Products Screen.	None	Products	None	None
		Step 3: The system retrieves the product categories	ProductCat egory	None	None	None
		Step 4: The system then displays the Add category type modal	None	Add category type modal	None	None
		Step 5: The admin selects the Product Category that he wants to add the category type to.	None	Add category type modal	None	None
		Step 6: The admin enters the category type's details	None	Add category type modal	None	None
		Step 7: The admin clicks on the add button	None	Add category type modal	None	None
		[ALT]: The user clicks on "cancel" which terminates the use case				
		Step 8: The system captures the	CategoryT ype	None	None	None

category type's details and validates it against the CategoryType Table [ALT]: The provided category type's details were not in the right format; the system will outline the place where the validation error				
occurred. Step 9: The system saves the captured information in the	CategoryT ype	None	None	None
CategoryType Table [ALT] The system cannot save the new category type. The system displays a modal telling the admin that the addition was unsuccessful.		[ALT] Add Category type Error modal		

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Search Category Type	9.6	Step1: The admin requests to search for a category type	None	None	None	None
		Step 2: The admin clicks on the "Products" screen.	None	None	None	None
		Step 3: The system responds by loading the "Products" screen	None	Products	None	None
		Step 4: The admin selects the "Category Type" radio button.	None	Products	None	None
		Step 5: The admin enters the category type search criteria	None	Products	None	None
		Step 6: The admin clicks on the search icon.	None	Products	None	None

Step 7: The system captures and validates the information against the category type table.	CategoryT ype	None	None	None
[ALT]: The validation fails, and the admin is prompted to enter a new search query				
Step 8: The system retrieves the matching entries in the CategoryType Table	CategoryT ype	None	None	None
Step 9: The system displays the retrieved search results in a table.	None	Products	None	None
Step 10: The admin selects the category type	None	Products	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports		
Update Category Type	9.7	Step 1: The admin requests to update a Category Type	None	None	None	None		
		Step 2: The system invokes UC 9.6 "Search Category Type"	CategoryT ype	Products	None	None		
		Step 3: The admin clicks on the update button	None	Products	None	None		
			responds by loading the "Update Category Type" modal	Category Type"	None	Update Category Type modal	None	None
		Step 5: The admin edits the Category Type	None	Update Category Type modal	None	None		
		Step 6: The admin clicks the "Save" button	None	Update Category Type modal	None	None		

[ALT] The admin clicks on the "Cancel" button which terminates the use case				
Step 7: The system captures and validates the changes made to the Category Type entered by the admin against the CategoryType table. [ALT] The validation of the entered information fails; the system will outline the information that does not meet the validation requirements	CategoryT	None	None	None
Step 8: The system updates the CategoryType table [ALT] The system	CategoryT ype	None [ALT]	None	None
will display a modal stating the update was unsuccessful		Update Category Type Error Modal		

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Delete Category Type	9.8	Step 1: The admin requests to delete a Category Type	None	None	None	None
		Step 2: The system invokes UC 9.6 "Search Category Type"	CategoryT ype	Products	None	None
		Step 3: The admin clicks on the delete button	None	Products	None	None
		Step 4: The system displays the delete Category Type modal	None	Delete Category Type modal	None	None

Step 5: The admin clicks on the "Yes" button [ALT] The admin clicks on the "No" button and the use case is terminated.	None	Delete Product Category modal	None	None
Step 6: The system removes the selected Category Type from the CategoryType entity. [ALT] The system will display a modal informing the admin that the deletion of the Category Type	CategoryT ype	None	None	None
was unsuccessful. Step 7: The system removes the deleted Category Type from the table in the products screen.	None	None	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Add Product Item	9.9	Step 1: The admin requests to add a new Product Item	None	None	None	None
		Step 2: The admin selects Product Item radio button	None	Products	None	None
		Step 3: The admin clicks on the "Add" button in the Products Screen.	None	Products	None	None
		Step 4: The system retrieves the Category Types	CategoryT ype	None	None	None
		Step 5: The system then displays the Add Product Item modal	None	Add Product Item modal	None	None
		Step 6: The admin selects the Category Type that he wants	None	Add Product Item modal	None	None

to add the category type to.				
Step 7: The admin enters the Product Item details	None	Add Product Item modal	None	None
Step 8: The admin clicks on the add button	None	Add Product Item modal	None	None
[ALT]: The user clicks on "cancel" which terminates the use case				
Step 9: The system captures the Product Item details and validates it against the ProductItem Table	ProductIte m	None	None	None
[ALT]: The provided category type's details were not in the right format; the system will outline the place where the validation error occurred.				
Step 10: The system saves the captured information in the ProductItem Table	ProductIte m	None	None	None
[ALT] The system cannot save the new product item. The system displays a modal telling the admin that the addition was unsuccessful.		[ALT] Add Product Item Error modal		

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Search Product Item	9.10	Step1: The admin requests to search for a Product Item	None	None	None	None

Step 2: The admin clicks on the "Products" screen.	None	None	None	None
Step 3: The system responds by loading the "Products" screen	None	Products	None	None
Step 4: The admin selects the "Product Item" radio button.	None	Products	None	None
Step 5: The admin enters the Product Item search criteria	None	Products	None	None
Step 6: The admin clicks on the search icon.	None	Products	None	None
Step 7: The system captures and validates the information against the ProductItem table. [ALT]: The validation fails, and	ProductIte m	None	None	None
the admin is prompted to enter a new search query				
Step 8: The system retrieves the matching entries in the ProductItem Table	ProductIte m	None	None	None
Step 9: The system displays the retrieved search results in a table.	None	Products	None	None
Step 10: The admin selects the Product Item	None	Products	None	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Update Product Item	9.11	Step 1: The admin requests to update a Product Item	None	None	None	None
		Step 2: The system invokes UC 9.10	ProductIte m	Products	None	None

"Coord Droduct				
"Search Product Item"				
Step 3: The admin clicks on the update button	None	Products	None	None
Step 4: The system responds by loading the "Update Product Item" modal	None	Update Product Item modal	None	None
Step 5: The admin edits the Product Item	None	Update Product Item modal	None	None
Step 6: The admin clicks the "Save" button	None	Update Product Item modal	None	None
[ALT] The admin clicks on the "Cancel" button which terminates the use case				
Step 7: The system captures and validates the changes made to the Product Item entered by the admin against the ProductItem table.	ProductIte m	None	None	None
[ALT] The validation of the entered information fails; the system will outline the information that does not meet the validation requirements				
Step 8: The system updates the ProductItem table	ProductIte m	None	None	None
[ALT] The system will display a modal stating the update was unsuccessful		[ALT] Update Product Item Error Modal		

Functional	Use	Narrative Steps	ERD	Screens	Methods	Reports
Requirement	Case No.		Entities			
Delete Product Item	9.12	Step 1: The admin requests to delete a Product Item	None	None	None	None
		Step 2: The system invokes UC 9.10 "Search Product Item"	ProductIte m	Products	None	None
		Step 3: The admin clicks on the delete button	None	Products	None	None
		Step 4: The system displays the delete Product Item modal	None	Delete Product Item modal	None	None
		Step 5: The admin clicks on the "Yes" button	None	Delete Product Item modal	None	None
		[ALT] The admin clicks on the "No" button and the use case is terminated.				
		Step 6: The system removes the selected Product Item from the ProductItem entity.	ProductIte m	None	None	None
		[ALT] The system will display a modal informing the admin that the deletion of the Product Item was unsuccessful.				
		Step 7: The system removes the deleted Product Item from the table in the products screen.	None	None	None	None

17.1.1. SPECIALS SUBSYSTEM

18. Function al Require ment	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Add Special	10.1 Add Special	Step 1: Request to add new special.	None	Special	None	None
		Step 2: Select add special button.	None	Special	None	None
		Step 3: Display Add Special screen.	None	Add Special	None	None
		Step 4: Enter special details.	None	Add Special	None	None
		Step 5: Admin clicks on the add button. [ALT: Admin clicks	None	Add Special	None	None
		on the Cancel button]				
		Step 6: The system validates the information	Special	None	None	None
		Step 7: System saves the information	Special	None	AddSpecial()	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Search Special	10.2	Step 1: The admin requests to search a special.	None	None	None	None
		Step 2: Enter search criteria	None	Special	None	None
		Step 3: Validate the search criteria	Special	None	None	None
		Step 4: Read the specials matching the search criteria	Special	None	None	None
		Step 5: Display the search results.	None	Special	SearchSpec ial()	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Update Special	10.3.	Step 1: Request to update a special.	None	None	None	None

Step 2: Invoke Use Case 10.10 Search Special	None	None	None	None
Step 3: Click on the update button.	None	Special	None	None
Step 4: Display Update Special screen	Special	Update Special	None	None
Step 5: Enter new special information. [Alt: Admin clicks on the Cancel button]	None	Update Special	None	None
Step 6: Validate the new information [ALT: Validation unsuccessful]	Special	None	None	None
Step 7: System saves updated information. [ALT: Couldn't save updated special information]	Special	None	UpdateSpec ial()	None

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Delete Special	10.4.	Step 1: Admin requests to delete a special.	None	None	None	None
		Step 2: Invoke Use Case 10.10 Search Special	None	None	None	None
		Step 3: The admin clicks on the Delete button.	None	Special	None	None
		Step 4: Display the delete modal.	None	Delete Modal	None	None
		Step 5: Admin confirms deletion.	None	Delete modal	None	None
		[ALT: Admin clicks on the cancel button]				

Step 6: The system	Special	None	DeleteSpeci	None
deletes the special			al()	
from the system.				
Step 7: The rows	None	Special	None	None
get updated in the				
table.				

Functional Requirement	Use Case No.	Narrative Steps	ERD Entities	Screens	Methods	Reports
Send Promotional Emails	10.5	Step 1: Request to send promotional email.	None	None	None	None
		Step 2: Admin clicks on the Send Promotions button.	None	Supplier	None	None
		Step 3: Display Send Promotions screen.	None	Send Promotions	None	None
		Step 4: Select the specials.		Send Promotions	None	None
		Step 5: Read all the specials in the table.	Special	None	None	None
	Step 6: Click on send email. [ALT: Admin click on the Cancel button]		None None	Send Promotions	None	None
		Step 7: System generates email.		None	None	None
		Step 8: Read the customer email and send email.	Customer	None	None	None

This section has the details of the validation of each procedure for each of the requirements in the NKAP Bolting system.

PROJECT RELATED DOCUMENTS

12. Sign-off by Client

INTRODUCTION

This section contains our Client Sign-Off

FORMALLY SIGNED PAGE OF CLIENTS AGREEMENT TO

CONTENT

IN	ıT	\Box	\frown	\Box	1 1	\sim	ГΙ	\sim	NI
II۱		Г	U	u	U	C	H	U	I۷

Deliverable Name: Technical Specification

Deliverable Number: 4 of 12

Delivered on Date:

Client Name: NKAP Bolting

Approving Client Representative: Mr Anandh Badrinath

Project Manager: Divya Bagratee

Group Leader: Jerome Amenigy

Deliverable Sign-Off

, as the approving client representative on behalf o KAP Bolting, declare that I have read this document in its entirety and hereby state nat I understand and fully agree with its content.
Anandh Badrinath
igned on This Date
grou or the zate

This section concludes. It contained the Client Sign-Off.

13. Complexity

INTRODUCTION

The complexity matrix has been filled out according to the capabilities of the DS3 Solutions members.

INF370 Complexity Matrix 2021

Topic	Le vel		Marks	MAX
	For online applications: Responsive web design			
	For desktop applications: Form design according to design principles (Schneiderman's golden rule on navigation applies here)	*	3	
	Appropriate use of grids/tables		3	
	Appropriate use of tabs/links		3	
	Use of graphs in an appropriate business context		4	
	The storage and display of graphical information, like		3	
	photos with a good business reason			
	Working e-mail automatically generated from the		2	
	database in an appropriate business context			
	SMS messages automatically generated from the		2	
	system in an appropriate business context			
1. Special GUI	Extensive user-friendly search facility		3	4
1. Special GOI	At least one use of a tree to display data from the		3	4 2
	database			
	Able to dynamically modify a data tree structure and		4	
	in doing so adjusting the data in the database			
	At least one use of a calendar view of data			
	(not a date/time picker; not a plug-in such as Google		3	
	calendar)			
	Uploading a file into the system with appropriate		3	
	business reason			

	The use of audio/video in an appropriate business		3	
	context		•	
	At least one use of an administrator configurable		3	1
	timer in an appropriate business context		-	
	At least 30 tables used (4 member groups) or 40	*	6	
	tables used (5 member groups)			
2. Database access	Full referential integrity on all tables	*	6	1
2. Database access	At least one use of master-detail table relationships (Schneiderman's golden rule on system status applies	*	3	5
	here)			
	At least 3 simple list reports in a reporting tool (no control breaks, no graphs, single table)	*	3	
2 Passarta	At least 2 transactional report with 2 or more control breaks (with heading and calculated values/totals, multiple tables)	*	6	
3. Reports	At least 1 report with adjustable criteria		3	1 5
	At least 1 management report using a graph		3	
4. Flexibility	All data that can change in future should not be hard coded but maintained in a sub-module of the system (e.g. Lookup tables)		6	1
	Some business rules are not hard coded but maintained in a sub-module of the system.		6	. 2
5. Error handling	All system-generated errors are trapped, and consistent, user-friendly error messages are displayed		6	1
	Appropriate data validation on all input fields		6	2
6. Help	At least one menu item or other control that opens up a complete help document (HTML, PDF, Help-file)		3	
	Extensive context-sensitive help. E.g. calling Help on a specific screen/function will automatically open the		6	1
	specific help for that screen/function.			5

Topic	Le vel	Marks	MAX
	Search Facility on Help	3	
	Extensive use of hints	3	

				1
	Logon screen with user ID and password and fixed user profiles		3	
	Applying two factor authentication with applicable business reason.		3	_
7. Security	Encrypted passwords in database	*	1	1
	Flexible user profiles (i.e. you can dynamically add user profiles that will enable/disable access to certain		6	3
	parts of the system)			
8. Audit Trail	An audit trail of all transactions in the system showing at least date, time, user, transaction type, critical data (such as amount and quantity of transaction)		6	9
	Able to search the audit trail on any of the following: date, user, transaction type		3	
	For a desktop application: Fully functional installation disks that take care of application installation		3	
	requirements (install and uninstall)			
	For an online application: Deployment of application to a publicly accessible web server		3	
9. Deployment	For a mobile application: Deployment to an App Market place (such as the PlayStore or the AppStore)		6	1 5
	Deployment of the database to a remote database server		3	-
10. Backup and Restore	A backup and restore subsystem exists that backup/restore all data (system may exit during restore)		3	3
11. Import/Export Data	Able to open Word or Excel and automatically place data in it based on the parameters provided (with a good business reason)		6	
	XML or JSON: At least 1 XMLor JSON file for Importing or Exporting of data (with good business reason)		3	9
	Simple Link to an external INPUT device using plug- and-play technology, such as a swipe card reader, bar code reader, etc. or a native component such as a QR reader, a GPS component, etc.		3	
	Loose Link to an external INPUT device using device specific software. Data or images must seamlessly be stored in the database, but device specific software is visible to the user. (This could include a digital camera, scanner, voice recording device, thump print reader, etc.)		6	

12. External INPUT device	Tight Link to an external INPUT device using device specific software. Data or images must seamlessly be stored in the database, but device specific software is not visible to the user. (This could include a digital camera, scanner, voice recording device, thump print reader, etc.)	9	
			1 8
	Integrate an existing web service into your application (with good business reason)	3	
13. External APPLICATION / Services	A fully functional link to an installed external application system exists and the interface must be shown to work on the external system. Note that this excludes Microsoft Office Applications	6	9

Topic	Le vel		Marks	MAX
	Appropriate business use of static views on an alternative platform.		3	
14. Multiplatform	Appropriate use of dynamic views on an alternative platform (i.e. data is displayed from the system's database)		3	
processing for an appropriate business reason	Appropriate use of substantial dynamic views on an alternative platform (i.e. both reading and writing data from the system's database)	0	9	
	Uploading a file through an alternative platform onto the system's database.		3	
	Substantial processing on a third platform (i.e. both reading and writing data from the system's database)		9	2 7
	The use of a data layer to facilitate interaction between your database and your business layer		3	
15. Programm ing Principles	The use of an API to facilitate interaction between your business layer and your presentation layer		6	
	Comprehensive use of stored procedures and/or triggers and/or jobs.		3	1 2
16. Innovative addition to the system	Any very advanced innovative addition to the system (e.g. machine learning, AI, block chain, text mining, IOT, etc.)	#	1 - 9	9

Maximum Complexity Marks	222
Complexity Marks Required (5 members in team)	150
Selected Complexity Marks	165

DS3Solutions has selected 165 complexity matrix marks out of the 222 marks.

Team 1 Deliverable 3&4

14. General

INTRODUCTION

This section contains the team sign-off.

14.1. TEAM SIGN-OFF

We, Group $1 - DS^3$ Solutions, hereby declare that this work is our own and we are fully satisfied with the work contained within this document. We declare that all members of this group have contributed equally and fairly to this deliverable and submit this document as our final and complete Functional Specification.

Project: NKAP Bolting POS

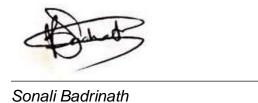
Divya Bagratee – Project Manager

Jerome Amenigy – Team Leader

S'nethemba Xulu

Chardé Vries





CONCLUSION

This section concludes. It contained the Team Sign-Of

Project: NKAP Bolting POS





This document contained our technical specification for the NKAP Bolting System. We looked at the system architecture design which included a Full set of Use Case Diagrams and Design Use Case Documentation of each Use Case. We also looked at Object-Oriented Design and System Database Design which included the Entity Relationship model and the size estimation of the proposed database. We then looked at the System Interface Design which included the inputs, outputs as well as other interfaces used in our system and lastly the Package Design Specifications which included the test specifications, hardware and software requirements, network/web layout specifications, and validations.

