



CALIFORNIA STATE UNIVERSITY
FULLERTON

SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

For

***CAFETERIA
ORDERING SYSTEM***

by

Team Apollo

DEEPAK PRASANNA M.Y (893338442)

CHONGBEI WANG (893525725)

Submission date: 14/5/18

Table of Contents

1. INTRODUCTION.....	4
1.1 PURPOSE	4
1.2 SCOPE.....	4
1.3 Definitions, acronyms, and abbreviations	5
1.4 References.....	5
1.5 Overview write in terms of chapters/sections.....	5
2. OVERALL DESCRIPTION.....	6
2.1 Product Perspective	6
2.2 PRODUCT FEATURES	8
2.3 User Classes and Characteristics	10
2.4 Operating Environment.....	12
2.5 Design and Implementation Constraints	12
2.6 User Documentation	12
2.7 Assumptions and Dependencies	14
3. Use Case Diagram with Use-case descriptions	15
<i>Use case descriptions:.....</i>	<i>16</i>
4. External Interface Requirements	32
4.1 User Interfaces.....	32
4.2 Hardware Interfaces	41
4.3 Software Interfaces.....	41
4.4 Communications Interfaces	41
5. Other Nonfunctional Requirements	42
5.1 Performance Requirements.....	42
5.2 Safety Requirements.....	42
5.3 Security Requirements	42
5.4 Software Quality Attributes	42
6. Other Requirements.....	43
7. DATA FLOW DIAGRAM.....	44
7.1 LEVEL 0 DATA FLOW DIAGRAM.....	44
.....	44
7.2 LEVEL 1 DATA FLOW DIAGRAM.....	45
7.3 LEVEL 2 DATA FLOW DIAGRAMS:.....	46
8. Functional Requirements	60
9. Sequence Diagram	69

10.	<i>Class Diagrams</i>	80
10.1	Initial Class diagram	80
10.2	Modified Class Diagram.....	81
10.3	Detailed Class Diagram.....	83

PART - I

1. INTRODUCTION

This section discusses about the purpose, scope description and a short context about the various topics in this SRS document. In addition to this, this section also describes a list of abbreviations and definitions which will be used throughout the document.

1.1 PURPOSE

The purpose of this document is to give a detailed description of the requirements for the “Cafeteria Ordering System” software. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval.

1.2 SCOPE

Cafeteria ordering system is a web application that allows various customers in Clackamas, Oregon location to order food online and get it delivered. This software is platform independent, flexibly runs on all the operating systems and in all mobile devices. Cafeteria ordering System provides food customization option by producing recipes and ingredient lists to the patrons. In addition to this, Cafeteria ordering system also allows the users to order meals from other local restaurants and the same will be delivered based on user’s delivery request.

Restaurant owners can provide their restaurant information using the web-portal. This information will act as the bases for the search results displayed to the user. COS generates e-receipt which acts as a token for identifying the customer who placed the order through cafeteria ordering system.

All the payment functions for the food orders will be processed by the Payroll department. A payroll processor verifies the payment details provided by the user and approves the payroll. Post the payroll approval from the payroll processor, the order transaction for the order will be processed and the meal order placed by the user will be successful.

Corporate management access COS for tracking the various feedbacks made by the users. Based on user’s feedback, the corporate management group plans the business objectives. In addition to this, Corporate management group also fetches the activity reports from the cafeteria application to track the Cafeteria employees performance and to the effectiveness of Cafeteria Ordering System (COS).

Furthermore, the software needs both Internet and GPS connection to fetch and display results regarding the restaurants available in Oregon location. All system information such as the user profile details, restaurant menu details, employee details in Cafeteria is maintained in a database, which is located on a web-server. The software also interacts with the GPS-Navigator software which is required to be an already installed application on the user’s mobile phone. By using the

COS, users can view desired restaurant's list in the Oregon and be able to place the meal order and get it delivered. The application also has the capability of producing activity reports on monthly, weekly or daily basis to the owner of Cafeteria.

1.3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS

TERM	DEFINITION
COS	<i>Cafeteria Ordering System</i>
Patron	<i>A customer, especially a regular one, of a store, restaurant, or theater.</i>
User	<i>Someone who interacts with the web application</i>
Admin/Administrator	<i>System administrator who is given specific permission for managing and controlling the system</i>
Restaurant Owner	<i>Someone who has a restaurant and wants his restaurant to be a part the application</i>
Web-Portal	<i>A web application which present special facilities for restaurant owner</i>
GPS	<i>Global Positioning System</i>
GPS-Navigator	<i>An installed software on mobile phone which could provide GPS connection and data, show locations on map and find paths from current position to defined destination</i>
Stakeholder	<i>Any person who has interaction with the system who is not a developer.</i>

1.4 REFERENCES

- [1] IEEE Software Engineering Standards Committee, "IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications", October 20, 1998.
- [2] Wiegers, Karl, *Software Requirements (3rd Edition)*, Microsoft Press 2013
- [3] Standardized Statement Templates – beginning to write SRS
Link: <https://www.coursera.org/learn/srs-documents-requirements/lecture/1o2v0/standardized-statement-templates>

1.5 OVERVIEW

This Software Requirements Specification document consists of 10 main sections. Among those 8 sections, the second section discusses about the product perspectives such as product features, various user classes, design and implementation constraints in the second section.

The third section describes the use case in detail and various use case descriptions for the same. The fourth section discusses about the various user interfaces, hardware interfaces and software

interfaces required for the Cafeteria Ordering System. The fifth and sixth section explains the non – functional requirements and other requirements that have to be considered while developing the COS.

The seventh section depicts the Data flow diagrams for Cafeteria Ordering System which mainly focuses on the data flow between various entities and processes. Eighth section describes about the functional requirements in a detailed process in expected Input, Output, Process and Error format. The last two sections explain the sequential flow of various process and the class diagrams of Cafeteria Ordering System.

2. OVERALL DESCRIPTION

2.1 PRODUCT PERSPECTIVE

The cafeteria ordering system has evolved as a solution to achieve the business objectives and to facilitate the food orders. The Cafeteria Ordering System replaces the traditional manual and telephone processes for ordering and picking up meals in the Process Impact cafeteria. The COS mainly consists of User Interface, Relational Database and a web-portal. Through user interface, patron, menu manager, meal processor, meal deliverer interacts and update the information pertaining to the order.

A system administrator maintains all the data regarding the user profile, restaurant profile and takes care of the overall cafeteria ordering system software. Figure 2.1 shows the pictorial representation of Cafeteria Ordering System.

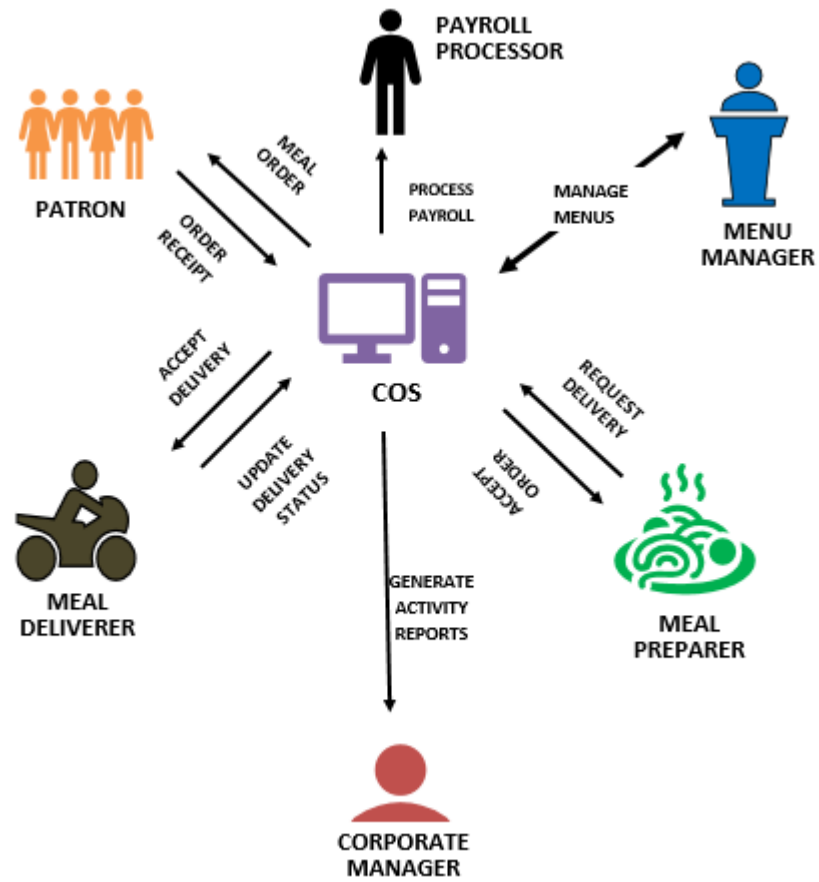


Figure 2.1 Simple diagrammatic representation of Cafeteria Ordering System

2.2 PRODUCT FEATURES

The following list offers a brief outline and description of the main features and functionalities of the Cafeteria Ordering system. The features are split into two major categories: core features and additional features. Core features are essential to the application's operation, whereas additional features simply add new functionalities. The latter features will only be implemented as time permits.

CORE FEATURES

1. ORDER MEAS FROM THE CAFETERIA MENU TO BE PICKED UP OR DELIVERED:

- ◆ User shall place a meal order from the cafeteria menu.
- ◆ Menu manager is responsible for creating and maintaining the 'Cafeteria menu.'
- ◆ User has the option to pick-up the meal at the restaurant or to get the meal delivered to his desired address
- ◆ In addition to this, user can place multiple items from the menu in a single order.

2. ORDER MEALS FROM OTHER LOCAL RESTAURANTS TO BE DELIVERED:

- ◆ User shall place a meal order from the other local restaurants menu.
- ◆ Menu manager is also responsible for creating and maintaining the 'Other local restaurant's menu.'
- ◆ User has the option to pick-up the meal at the restaurant or to get the meal delivered to his desired address.
- ◆ Cafeteria meal deliverer is responsible for delivering the meal to the user's desired address.
- ◆ User has the option to pick-up the meal at the restaurant or to get the meal delivered to his desired address.
- ◆ In addition to this, user can place multiple items from the menu in a single order.

3. CREATE, VIEW, MODIFY, AND DELETE MEAL SERVICE SUBSCRIPTIONS:

- ◆ User can subscribe a meal plan from the cafeteria menu or from the local restaurant menu.
- ◆ Menu manager is responsible for creating and maintaining the various subscription plans for the patrons.
- ◆ User has the option to pick-up the meal he subscribed at the restaurant or to get the meal delivered to his desired address.
- ◆ Cafeteria meal deliverer is responsible for delivering the meal to the user's desired address.
- ◆ User has the option to pick-up the meal at the restaurant or to get the meal delivered to his desired address.
- ◆ In addition to this, user can place multiple subscriptions from the menu in a single

order.

4. REGISTER FOR MEAL PAYMENT OPTIONS:

- ◆ User can register his card details in COS application.
- ◆ User shall maintain those details under 'My Account' button.
- ◆ COS will prompt user to use the registered card details or pay using new card at the time of checking out the order before placing it.
- ◆ If the user has not registered his card details, user can pay at the time of checking out the order before placing it.

5. REQUEST MEAL DELIVERY:

- ◆ User has the option to schedule his delivery at his desired time.
- ◆ User can change his delivery address before the meal getting picked up by the meal deliver.

6. CREATE, VIEW, MODIFY AND DELETE CAFETERIA MENUS:

- ◆ Menu manager is the corporate boss for operating 'Cafeteria Ordering System'.
- ◆ Menu manager is responsible for creating and maintaining the 'Cafeteria menu.' And 'Other Local Restaurants menu.'
- ◆ Menu manager is responsible for creating and maintaining the various subscription plans for the patrons.
- ◆ Menu manager can create, view, modify and delete cafeteria menu.

7. ORDER CUSTOM MEALS THAT AREN'T ON THE CAFETERIA MENU

- ◆ User shall order a new meal that is not available in cafeteria menu.
- ◆ User shall provide the style of the food and the ingredients to customize the order.

8. PRODUCE RECIPES AND INGREDIENT LISTS FOR CUSTOM MEALS FROM CAFETERIA

- ◆ Menu manager shall create and maintain the recipes and ingredients list available in cafeteria.
- ◆ Menu manager tracks the available ingredients and updates it frequently in COS.
- ◆ Menu manager is also responsible for creating and maintaining the 'Cafeteria menu' and 'Other Local Restaurants menu.'
- ◆ While ordering the custom meals, user picks up the items from 'Recipes and Ingredients list' and places a custom order.

9. PROVIDE COS ACCESS OUTSIDE CAFETERIA INTRANET

- ◆ COS will be accessible outside the cafeteria intranet.
- ◆ COS makes use of web-server which hosts the COS functionalities and data that is accessible from elsewhere.

ADDITIONAL FEATURES

10. SET/REMOVE DEFAULT CUISINE:

- ◆ User has the preference to select a default cuisine.
- ◆ User can select Chinese cuisine, Thai food cuisine, Indian cuisine, Native American cuisine, Asian cuisine, Mediterranean cuisine and other types of cuisines.
- ◆ Preferring a default cuisine will make the user to adapt his native food style, making the user experience more efficient.
- ◆ User also can change or cancel the selected cuisine, at the time he wishes to deselect the selected cuisine.

11. LOGIN THROUGH FACEBOOK AND SHARE ACTIVITIES IN FACEBOOK:

- ◆ User has the preference to register with COS application through Facebook.
- ◆ User can share his activities with COS in Facebook, like ‘Ordered and Enjoyed Chicken Biryani from Cafeteria – Simply Delicious’.
- ◆ Sharing those activities in Facebook, builds up Cafeteria reputation among the customers.
- ◆ Sharing activities in Facebook, promotes Marketability for Cafeteria restaurant.

12. ADD MY FEEDBACK:

- ◆ User has the feature to add his feedback about the Quality of food, Quality of Service and Experience in ordering the food through cafeteria.
- ◆ Feedback feature will be active to the user after two hours post the meal delivery.
- ◆ User’s feedback will be monitored directly by the Menu-Manger.

2.3 USER CLASSES AND CHARACTERISTICS

The various user classes for Cafeteria Ordering System are as follows,

PATRON

- A Patron is a Process Impact employee who wants to order meals to be delivered from the company cafeteria.
- There are about 600 potential Patrons, of which 300 are expected to use the COS an average of 5 times per week each.
- Patrons will sometimes order multiple meals for group events or guests. An estimated 60 percent of orders will be placed using the corporate intranet, with 40 percent of orders being placed from home or by smartphone or tablet apps

- All the personal details of Patron will be accessible under ‘Account settings’ label. Patron can change/edit his card details, delivery details, order history and default cuisine preference under the account settings option.

CAFETERIA STAFF

- The Process Impact cafeteria employs about 20 Cafeteria Staff who will receive orders from the COS, prepare meals, package them for delivery, and request delivery.
- Most of the Cafeteria Staff will need training in the use of the hardware and software for the COS

MENU MANAGER

- The Menu Manager is a cafeteria employee who establishes and maintains daily menus of the food items available from the cafeteria. Menu Manager maintains the recipes and ingredients along-side with the menus.
- Patrons make use of the ingredients and recipe list to place a customized meal order in Cafeteria Ordering System. Some menu items may not be available for delivery. The Menu Manager will also define the cafeteria’s daily specials. The Menu Manager will need to edit existing menus periodically
- Menu Manager creates a new menu, updates a menu and maintains the recipes and ingredients list updated with the Cafeteria stock.

MEAL DELIVERER

- As the Cafeteria Staff prepare orders for delivery, they will issue delivery requests to a Meal Deliverer’s smartphone.
- The Meal Deliverer will pick up the food and deliver it to the Patron.
- A Meal Deliverer's other interactions with the COS will be to confirm that a meal was (or was not) delivered.

CORPORATE MANAGEMENT

- The corporate management manages the employee payroll details and maintains the employee data.
- The corporate management occasionally uses the COS. They mainly track the COS activity and fetch the monthly, weekly and daily reports from COS

PAYROLL DEPARTMENT

- All the payment functions for the food orders will be processed by the Payroll department.
- A payroll processor verifies the payment details provided by the user and approves the payroll.

- Post the payroll approval from the payroll processor, the order transaction for the order will be processed and the meal order placed by the user will be successful.

2.4 OPERATING ENVIRONMENT

OPERATING ENVIRONMENT	DESCRIPTION
Mobile Platform	<i>The COS shall operate on a server running the current corporate-approved versions of Red Hat Linux and Apache HTTP Server.</i>
Web browser version	<i>The COS shall operate correctly with the following web browsers: Windows Internet Explorer versions 7, 8, and 9; Firefox versions 12 through 26; Google Chrome (all versions); and Apple Safari versions 4.0 through 8.0.</i>
Server Environment	<i>The COS shall operate on a server running the current corporate-approved versions of Red Hat Linux and Apache HTTP Server.</i>

2.5 DESIGN AND IMPLEMENTATION CONSTRAINTS

CONSTRAINTS NUMBER	DESCRIPTION
Constraint 1	<i>The Cafeteria Ordering system's code shall be programmed in Java programming language, considering the software's flexibility to operating in different platforms.</i>
Constraint 2	<i>The system's design, code, and maintenance documentation shall conform to the Process Impact Intranet Development Standard, Version 1.3</i>
Constraint 3	<i>The system shall use My SQL database for storing the information</i>
Constraint 4	<i>Signal Protocol shall be configured in Cafeteria Ordering System for ensuring the security from third party hackers.</i>
Constraint 4	<i>Only the menu manager shall have the access to edit the menus.</i>
Constraint 5	<i>System Administrator will be responsible for maintaining the delivered software</i>

2.6 USER DOCUMENTATION

This section elaborates the user manual and tutorials for various user class.

User Manual:

User manual for Downloading and general registration of Cafeteria Ordering System application from Mobile store

Step 1: Download Cafeteria Ordering application from mobile store

Step 2: Login/Register with your credentials

Step 3: After Successful login, please enter your mobile number and preferred delivery address

Step 4: Please enter your card details under payment section

Step 5: General Register is complete; the user is now ready to place the order.

Tutorials:

Tutorial Ordering a meal in mobile application:

Step 1: Once the general registration is complete; the user is now eligible to place meal order.

Step 2: Search by restaurant or Search by meal in the search bar to select your food and click Search

Step 3: Select the quantity of the food

Step 6: Click proceed to checkout to finalize your order

Step 7: Select delivery type and click pay.

Step 8: You will receive an acknowledgement displaying “Transaction Successful” alongside with an e-receipt with your order. Another e-receipt will also be sent to your registered mobile number through SMS.

Step 9: Please show your e-receipt SMS to the meal deliverer to confirm the order ownership.

Tutorial for ordering a meal in web portal:

Step 1: Please enter the below URL in your web browser

<https://www.cafeteriaonlineorder.com/>

Step 2: Login/register with your credentials

Step 2: Fill address and card details to complete registration.

Step 2: Search by restaurant or Search by meal in the search bar to select your food and click Search

Step 3: Select the quantity of the food

Step 6: Click proceed to checkout to finalize your order

Step 7: Select delivery type and click pay.

Step 8: You will receive an acknowledgement in the web portal displaying “Transaction Successful” alongside with an e-receipt with your order. Another e-receipt will also be sent to the user’s registered mobile number through SMS.

Step 9: The user ought to display e-receipt SMS to the meal deliverer to confirm the order ownership.

Tutorial for Meal Processor:

Step 1: The meal processor shall be registered with Cafeteria Ordering System through web portal

Step 2: Accept Order

Step 3: Update Order Processing Status

ORDER STATUS	DESCRIPTION
<i>Order Accepted</i>	This order processing status elaborates that the order has been received and the order is processing.
<i>Delivered</i>	This order status elaborates that the meal request is complete and it is pending for delivery pickup at store.

Tutorial for Meal deliverer:

Step 1: Download Cafeteria Ordering application from mobile store

Step 2: Login with employee ID

Step 3: Accept “Delivery pickup”

Step 4: Update delivery status

DELIVERY STATUS	DESCRIPTION
<i>Out for delivery</i>	This delivery status describes that the order has been picked up at the store and will be delivered to the user shortly.
<i>Delivered</i>	This delivery status indicates that the order has been successfully delivered to the patron.

2.7 ASSUMPTIONS AND DEPENDENCIES

Assumption:

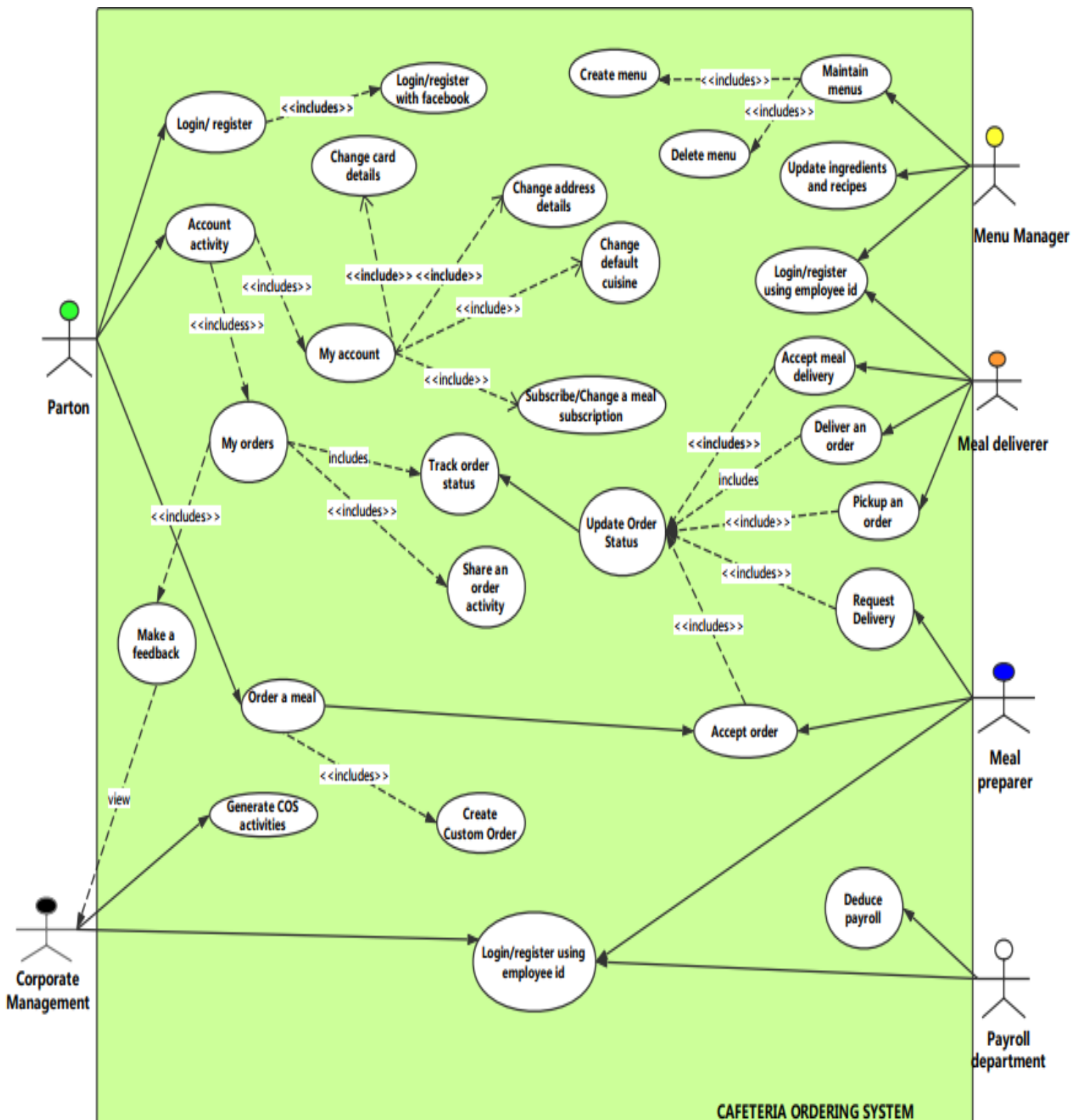
The cafeteria is open for breakfast, lunch, and supper every company business day in which employees are expected to be on site.

Dependency 1:

The Cafeteria ordering system is highly dependent on the internet connection, as COS will be a web application, in which all the orders and order processing will be carried out only through internet connectivity.

Dependency 2:

The operation of the COS depends on changes being made in the Cafeteria database to update the availability of food items as COS accepts meal orders.



USE CASE DESCRIPTIONS:

Use Case ID:	UC-1		
Use Case Name:	Register with Cafeteria application		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	29/4/18
Actors:	End User		
Description:	This use case helps user to register with Cafeteria ordering system in his/her device.		
Trigger:	User registers with Cafeteria application to order his favorite meal from his home or office rather than reaching restaurant.		
Preconditions:	1. End user shall own a personal smart mobile phone or a personal device that supports internet connectivity. 2. End user shall own a personal phone number and an email-id or Facebook account.		
Postconditions:	User is registered with Cafeteria application and User owns a unique 'user-id' for his identification with COS.		
Normal Flow:	1.0 User installs Cafeteria ordering system application from google play store or from apple store 2.0 User provides his mobile number and email id for registration 3.0 COS sends a randomly generated verification code to user's phone. 4.0 User enters the verification code and registers with COS. 5.0 COS generates a separate 'user id' for user's unique identification 6.0 User enters his personal password for registering with COS. 7.0 COS displays default cuisine screen including list of cuisine available with COS. 8.0 User provides his choice. 9.0 COS displays a 'registration completed' message to the user.		
Alternative Flows: [Alternative Flow 1 – Not in Network]	2.1. User chooses signup with Facebook 2.1.1 Cafeteria application requests the user to authorize his Facebook profile to invoke the information. 2.1.2. Cafeteria application requests the user to provide his mobile number. 8.1 User selects <i>skip this step</i> option. 8.2 COS automatically assumes the default cuisine as 'ALL'		
Exceptions:	2E. <u>Invalid phone number</u> -2E1. COS displays an error message if the user's mobile number is invalid. 2E. <u>Invalid email id</u> -2E2. COS displays an error message if the user's email id is invalid. 2.1.E.1 <u>Facebook account not authorized</u> COS displays an error message if the user fails to authorize his Facebook account for registration. 2.1.E.2 <u>Invalid phone number</u> COS displays an error message if the user's mobile number is invalid. 3.0.E.1 <u>Incorrect verification code</u> COS displays an error message if the generated verification code is incorrect.		
Includes:	1. The user has a valid email address to access Google play store for		

	downloading COS.
Frequency of Use:	Approximately 50 users a day and 500 users in a month.
Assumptions:	<ol style="list-style-type: none"> 1. The End user understands English 2. COS assumes the default cuisine preference for user as 'ALL' if the user skips the default cuisine setting step (step 8.0 in normal flow)
Priority	High

Use Case ID:	UC-2		
Use Case Name:	Order a meal		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	29/4/18
Actors:	End User		
Description:	This use case helps user to place a meal order		
Trigger:	User orders a meal from his location to get it delivered or picked up by him at his convenience.		
Preconditions:	End user is registered with COS		
Postconditions:	User successfully places a meal order and e-receipt and SMS receipt will be generated as a token of meal confirmation		
Normal Flow:	<ol style="list-style-type: none"> 1.0 User opens the COS application. 2.0 COS displays <i>home screen</i> including <i>customize a meal</i>, <i>Most rated recipes</i>, <i>account activity</i>, <i>meal plan</i> and a <i>search bar</i> on the top of the screen. 3.0 User enters his meal's keyword or the restaurant's keyword on the search bar. 4.0 COS displays list of available options for the keywords entered by the user. 4.0 User selects an option among those options and selects the quantity 5.0 User selects 'Add to my Dine' option to have the meal order in his basket. 6.0 User enters the delivery details and clicks '<i>proceed to checkout</i>'. 7.0 At the checkout screen, user reconfirms his order and clicks '<i>finalize and pay</i>' 8.0 COS generates e-receipt and a SMS receipt to the user's registered mobile number with an 'Order placed successfully' acknowledgement. 		
Exceptions:	<ol style="list-style-type: none"> 3.0.E.1 No results found COS displays an error message if the user enters an incorrect search keyword. 4.0.E.1 Invalid address or zip code. COS displays an error message if the user enters an incorrect address details. 7.0.E.1 Transaction not processed COS displays an error message if the user's card details are incorrect or expired and cancels the transaction. 		
Includes:	<ol style="list-style-type: none"> 1. COS displays integrated menu for the keyword entered by the user. <ol style="list-style-type: none"> 1.1. Integrated menu includes the recipe list from Cafeteria menu and from other local restaurant's menu. 2. The user has a valid email address or valid Facebook account. 3. User has sufficient amount in his bank to pay for the meal. 		

Frequency of Use:	Approximately 50 users a day and 500 users in a month.
Assumptions:	The End user understands English
Priority	High

Use Case ID:	UC-3		
Use Case Name:	Order a customized meal		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	07/5/18
Actors:	End User		
Description:	This use case helps user to place a customized meal order		
Trigger:	User might order his own recipe. Cafeteria provides the space for the user to create his own recipe.		
Preconditions:	End user is registered with COS		
Postconditions:	User successfully places a customized meal order and e-receipt and SMS receipt will be generated as a token of meal confirmation		
Normal Flow:	1.0 User opens the COS application. 2.0 COS displays <i>home screen</i> including <i>customize a meal</i> , <i>Most rated recipes</i> , <i>account activity</i> , <i>meal plan</i> and a <i>search bar</i> on the top of the screen. 3.0 User selects <i>customize a meal</i> option 4.0 COS displays customized meal option screen including – <i>Ingredients list and recipe list</i> 5.0 User selects the ingredients or recipe and those quantities which he feels to have on his recipe. 6.0 User shall make a mandatory note for his <i>cooking instructions</i> or an <i>additional note</i> for his customized meal. 7.0 User selects ‘ <i>Add to my Dine</i> ’ option to have the meal order in his basket. 8.0 User enters the delivery details and clicks ‘ <i>proceed to checkout</i> ’. 9.0 At the checkout screen, user reconfirms his order and clicks ‘ <i>finalize and pay</i> ’ 10.0 COS generates e-receipt and a SMS receipt to the user’s registered mobile number with an ‘ <i>Customized meal Order placed successfully</i> ’ acknowledgement.		
Exceptions:	4.0.E.1 Invalid address or zip code. COS displays an error message if the user enters an incorrect address details. 7.0.E.1 Transaction not processed COS displays an error message if the user’s card details are incorrect or expired and cancels the transaction.		
Includes:	4. COS displays integrated menu for the keyword entered by the user. 1.2. Integrated menu includes the recipe list from Cafeteria menu and from other local restaurant’s menu. 5. The user has a valid email address or valid Facebook account. 6. User has sufficient amount in his bank to pay for the meal.		
Frequency of Use:	Approximately 50 users a day and 500 users in a month.		
Assumptions:	The End user understands English		
Priority	High		

Use Case ID:	UC-4		
Use Case Name:	Subscribe a meal plan		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	4/5/18
Actors:	End User		
Description:	This use case helps user to subscribe a meal plan		
Trigger:	Rather than placing a meal order daily, user wishes to subscribe a meal plan, so that meal will be delivered to the user regularly.		
Preconditions:	End user is registered with COS End user is not subscribed with meal plan		
Postconditions:	User successfully subscribes a meal plan		
Normal Flow:	1.0 User opens the COS application. 2.0 COS displays <i>home screen</i> including <i>customize a meal</i> , <i>Most rated recipes</i> , <i>account activity</i> , <i>meal plan</i> and a <i>search bar</i> on the top of the screen. 3.0 COS displays ' <i>Meal plan</i> ' screen including ' <i>subscribe a meal plan</i> ' option. 4.0 User selects meal plan option 5.0 COS displays list of meal plans 6.0 User selects an option among those options and selects the quantity 7.0 User selects ' <i>Add to my subscription</i> ' option to have the meal order in his basket. 8.0 User enters the delivery details and clicks ' <i>proceed to checkout</i> '. 9.0 At the checkout screen, user reconfirms his order and clicks ' <i>finalize and pay</i> ' 10.0 COS generates e-receipt and a SMS receipt to the user's registered mobile number with an ' <i>Meal plan successfully subscribed</i> ' acknowledgement.		
Exceptions:	8.0.E.1 Invalid address or zip code. COS displays an error message if the user enters an incorrect address details. 9.0.E.1 Transaction not processed COS displays an error message if the user's card details are incorrect or expired and cancels the transaction.		
Includes:	The user has a valid email address or valid Facebook account.		
Frequency of Use:	Approximately 20 users a day, mainly depends on user's interest.		
Assumptions:	The End user understands English		
Priority	High		

Use Case ID:	UC-5		
Use Case Name:	Track order status		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	07/5/18
Actors:	End User		
Description:	This use case helps user to track his order		
Trigger:	User wishes to track his meal order, so that user shall plan his activity with respect to order delivery.		
Preconditions:	1.User shall hold a current order 2.End user is registered with COS		

Postconditions:	User successfully tracks his order
Normal Flow:	1.0 User opens the COS application. 2.0 COS displays <i>home screen</i> including <i>customize a meal</i> , <i>Most rated recipes</i> , <i>account activity</i> , <i>meal plan</i> and a <i>search bar</i> on the top of the screen. 3.0 User selects account activity option 4.0 COS displays account activity screen including order history, my account options 5.0 User selects <i>order history</i> option 6.0 COS displays list of orders in order history section including the active order 7.0 User selects active order option 8.0 COS displays the details of active order intimating the current order status
Exceptions:	3.0.E.1 No orders found COS displays an error message if the user don't hold an active order or any previous meal orders
Includes:	1. COS displays integrated menu for the keyword entered by the user. 1.3. Integrated menu includes the recipe list from Cafeteria menu and from other local restaurant's menu. 2. The user has a valid email address or valid Facebook account. 3. User has sufficient amount in his bank to pay for the meal.
Frequency of Use:	Approximately 50 users a day and 500 users in a month.
Assumptions:	The End user understands English
Priority	High

Use Case ID:	UC-6		
Use Case Name:	Manage my account		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	29/4/18
Actors:	End User		
Description:	This use case helps user to manage his Cafeteria account		
Trigger:	User can fill his/her address, card details, Default meal preference, meal subscription in advance to avoid the haste at the time of placing a meal order.		
Preconditions:	End user is registered with COS		
Postconditions:	User successfully changes or ensures his Cafeteria profile details.		
Normal Flow:	1.0 User opens the COS mobile application. 2.0 COS displays <i>home screen</i> with <i>popularly rated recipes</i> , <i>Customize a meal</i> , <i>order history</i> , <i>account activity</i> and a <i>search bar</i> on the top of the screen. 3.0 User selects <i>account activity</i> option on the screen. 4.0 COS displays <i>account activity</i> screen including order history, my account options 5.0 User selects my account option 5.0 COS provides various options – <i>Payment</i> , <i>Default cuisine</i> , <i>Address</i> . 6.0 User provides his details 7.0 User clicks save. 8.0 COS generates an acknowledgement intimating <i>Changes saved</i>		

	<i>successfully.</i>
Alternative Flows: [Alternative Flow 1 – Not in Network]	4.1 User selects <i>Payment</i> 4.1.1 User enters his card details. 4.2 User selects <i>Default Cuisine</i> 4.2.1 User selects his favorite cuisine among the list of options displayed in Cafeteria application. 4.3 User selects <i>Address</i> 4.3.1 User enters his address details
Exceptions:	4.1.E.1 Incorrect card details COS displays an error message if the user enters an incorrect card details. 4.3.E.1 Invalid address or zip code. COS displays an error message if the user enters an incorrect address details.
Includes:	1. The user has a valid email address or to access Google play store for downloading COS. 2. The user has a valid Facebook account.
Frequency of Use:	Approximately 50 users a day and 500 users in a month.
Assumptions:	The End user understands English
Priority	High

Use Case ID:	UC-7		
Use Case Name:	Make a feedback		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	30/4/18
Actors:	End User		
Description:	This use case assists the user to make a feedback on the meal order.		
Trigger:	1. User wishes to log his opinion to the meal order provided by 2. Cafeteria. User's opinions help the Cafeteria to improve the business.		
Preconditions:	1. End user is registered with COS 2. Ordered meal has been successfully delivered to the user		
Postconditions:	User successfully logs his feedback about the meal order		
Normal Flow:	1.0 User opens the COS application. 2.0 COS displays <i>home screen with popularly rated recipes, Customize a meal, order history, account activity</i> and a <i>search bar</i> on the top of the screen. 3.0 User selects account activity option on the screen. 4.0 COS displays account activity screen including order history, my account options		

	5.0 User selects <i>my Orders</i> 6.0 COS displays list of orders for the user including the current order. 7.0 User selects an option among those options. 8.0 User selects Feedback option to log his opinion about the order. 9.0 User fills his opinions pertaining to the order and 'clicks submit' button. 10.0 COS displays an acknowledgement to the user intimating " <i>Thanks for your feedback</i> ".
Exceptions:	3.0.E.1 No orders found COS displays an error message if the user's order history is nil.
Includes:	Post the meal delivery, Feedback option will be available to the user.
Frequency of Use:	Approximately 20 users a day, mainly depends on user's interest.
Assumptions:	The End user understands English
Priority	High

Use Case ID:	UC-8		
Use Case Name:	Share Cafeteria activity with Facebook		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	30/4/18
Actors:	End User		
Description:	This use case helps user to share his activity to his friends in Facebook.		
Trigger:	User wishes to express his happiness or his COS activities to make it noticed by his/her friends in Facebook		
Preconditions:	1. End user is registered with COS 2. End user holds an active Facebook account.		
Postconditions:	User successfully posts his COS activity as a status message in Facebook		
Normal Flow:	1.0 User opens the COS application. 2.0 COS displays <i>home screen</i> with <i>popularly rated recipes</i> , <i>Customize a meal</i> , <i>order history</i> , <i>account activity</i> and a <i>search bar</i> on the top of the screen. 3.0 User selects order history on the screen. 4.0 COS displays list of orders for the user including the current order. 5.0 User selects an option among those options. 6.0 COS displays the order details and displays share with – <i>Facebook</i> , <i>WhatsApp</i> , <i>Instagram</i> . 7.0 User selects Facebook option to share his selected order. 8.0 User clicks post in Facebook 9.0 COS displays an acknowledgement to the user intimating Activity posted on your Facebook profile.		
Alternative Flows:	5.1 COS requests the user to authorize his Facebook profile, if the user is not signed up through Facebook. 5.2 User selects WhatsApp 5.2.1 COS requests the user to authorize his WhatsApp profile		
Exceptions:	3.0.E.1 No orders found COS displays an error message if the user's order history is nil. 4.0.E.1 Invalid address or zip code. COS displays an error message if the user enters an incorrect address details.		

Includes:	The user has a valid email address or valid Facebook account.
Frequency of Use:	Approximately 20 users a day, mainly depends on user's interest.
Assumptions:	The End user understands English
Priority	High

Use Case ID:	UC-9		
Use Case Name:	Change Default cuisine		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	30/4/18
Actors:	End User		
Description:	This use case helps user to update his/her default cuisine		
Trigger:	User prefers to order his native or favorite cuisine rather than surfing the whole menu		
Preconditions:	1. End user is registered with COS		
Postconditions:	User successfully changes or updates his favorite cuisine		
Normal Flow:	1.0 User opens the COS application. 2.0 COS displays <i>home screen</i> with <i>popularly rated recipes</i> , <i>Customize a meal</i> , <i>order history</i> , <i>account activity</i> and a <i>search bar</i> on the top of the screen. 3.0 User selects <i>Account activity</i> on the screen 4.0 COS displays list of options for the user including order history, my account. 5.0 User selects my account option 6.0 User selects <i>Default cuisine</i> option. 7.0 COS displays list of available cuisine options for the user including the <i>current Default Cuisine</i> . 8.0 User selects his favorite cuisine among those cuisine options. 9.0 User clicks <i>save</i> . 10.0 COS generates an acknowledgement intimating <i>Changes saved successfully</i> .		
Includes:	The user has a valid email address or valid Facebook account.		
Frequency of Use:	Approximately 20 users a day, mainly depends on user's interest.		
Assumptions:	The End user understands English		
Priority	High		

Use Case ID:	UC-10		
Use Case Name:	Register with Cafeteria using Employee id (Only for Cafeteria Employees)		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	29/4/18
Actors:	Cafeteria Employees		
Description:	This use case helps Cafeteria Staff to login with Cafeteria application		
Trigger:	Cafeteria Employees shall login with Cafeteria application to accept order, generate COS activity report, create/delete menus etc.		
Preconditions:	1. Cafeteria employee shall own a personal smart mobile phone or a personal device that supports internet connectivity. 2. Cafeteria employee shall use their employee id to register with Cafeteria		

Postconditions:	Cafeteria staff is registered with Cafeteria application and that staff's employee id will be used as COS id as well.
Normal Flow:	1.0 Cafeteria staff installs Cafeteria ordering system application from google play store or from apple store 2.0 Cafeteria staff opens COS application which includes sign up with COS, employee login 3.0 Cafeteria staff selects employee login option 4.0 COS displays employee login screen 5.0 COS employee enters his employee id, phone number and clicks login 6.0 COS asks for employee role including menu manager, meal deliver, meal preparer, corporate manager payroll processor 8.0 Cafeteria employee selects his role 7.0 COS generates an acknowledgement intimating 'Employee registration successful'. 9.0 COS displays home screen with respect to the employee's selected role.
Alternative Flows:	1.1 Cafeteria employee visits www.myfoodcos.com in web browser.
Exceptions:	2E. <u>Invalid phone number</u> -2E1. COS displays an error message if the user's mobile number is invalid. 2E. <u>Invalid employee id</u> -2E2. COS displays an error message if the employee id is invalid.
Includes:	The user has a valid email address to access Google play store for downloading COS.
Frequency of Use:	Approximately 50 users a day and 500 users in a month.
Assumptions:	The End user understands English
Priority	High

Use Case ID:	UC-11		
Use Case Name:	Login with Cafeteria application using employee ID		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	07/5/18
Actors:	Cafeteria staff		
Description:	This use case helps Cafeteria staffs to login with Cafeteria application		
Trigger:	Cafeteria employees such as menu manager, meal preparer, meal deliverer shall be logging in to the cafeteria application every day to resume their duty.		
Preconditions:	1. Cafeteria employee shall own a personal smart mobile phone or a personal device that supports internet connectivity. 2. Cafeteria employee shall use their employee id to register with Cafeteria		
Postconditions:	Cafeteria employee is successfully logged in using employee id.		
Normal Flow:	1.0 Cafeteria staff opens COS application through mobile or Personal computer 2.0 Cafeteria staff enters his employee id and password for logging in with COS 3.0 COS displays a 'Welcome back' message to the user.		
Alternative Flows:	1.1 User visits www.myfoodcos.com in web browser.		
Exceptions:	2.0.E.1 <u>Invalid employee id</u> COS displays an error message if the employee id is invalid. 2.0.E.2 Incorrect password		

	COS displays an error message if the employee's password is incorrect.
Includes:	Cafeteria employee is registered with COS.
Frequency of Use:	Approximately 50 users a day and 500 users in a month.
Assumptions:	Cafeteria employee understands English
Priority	High

Use Case ID:	UC-12		
Use Case Name:	Create a menu		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	29/4/18
Actors:	Menu Manager		
Description:	This use case helps menu manager to create a menu		
Trigger:	Menu manager collects various menus from Cafeteria and other restaurants and creates an integrated menu list for the users.		
Preconditions:	Menu manager is registered with COS using his employee ID.		
Postconditions:	Menu Manager successfully creates a new menu		
Normal Flow:	1.0 Menu manager opens COS from his device. 2.0 COS displays <i>home screen</i> for menu manager including – <i>create menu, current menu, update ingredients list</i> 3.0 Menu manager selects his option from the home screen. 4.0 Menu manager provides menu name, menu category, various recipes in the menu and ingredients for the recipes. 5.0 Menu manager rechecks and clicks save. 6.0 COS generates an acknowledgement intimating <i>Menu created successfully</i> .		
Exception:	3.0.E.1 Menu name not provided 3.0.E.2 Menu category not provided		
Includes:	Menu manager collects various menus from Cafeteria and other restaurants.		
Frequency of Use:	5 – 10 times a day, depending on the need		
Assumptions:	The Menu manager understands English		
Priority	High		

Use Case ID:	UC-13		
Use Case Name:	Update a menu		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	29/4/18
Actors:	Menu Manager		
Description:	This use case helps menu manager to update a menu		
Trigger:	Menu manager collects various menus from Cafeteria and other restaurants and creates an integrated menu list for the users.		
Preconditions:	Menu manager is registered with COS using his employee ID.		
Postconditions:	Menu Manager successfully performs his activity with the menu list		
Normal Flow:	1.0 Menu manager opens COS from his device. 2.0 COS displays <i>home screen</i> for menu manager including – <i>create menu, current menu list, update ingredients list</i> 3.0 Menu manager selects <i>current menu</i> option from the home screen. 4.0 COS displays list of menus on the screen		

	5.0 Menu manager selects his choice 6.0 Menu manager updates any of the following options - menu name, menu category, various recipes in the menu and ingredients for the recipes, delete this menu. 7.0 Menu manager rechecks and clicks save. 8.0 COS generates an acknowledgement intimating <i>Changes saved successfully</i> .
Includes:	Menu manager collects various menus from Cafeteria and other restaurants.
Frequency of Use:	5 – 10 times a day, depending on the need
Assumptions:	The Menu manager understands English
Priority	High

Use Case ID:	UC-14		
Use Case Name:	delete a menu		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	29/4/18
Actors:	Menu Manager		
Description:	This use case helps menu manager to delete a menu		
Trigger:	Menu manager might feel to delete an old menu and to make a fresh new menu out of it, rather than editing the entire contents		
Preconditions:	Menu manager is registered with COS using his employee ID.		
Postconditions:	Menu Manager successfully deletes a menu from the menu list		
Normal Flow:	1.0 Menu manager opens COS from his device. 2.0 COS displays <i>home screen</i> for menu manager including – <i>create menu, current menu list, update ingredients list</i> 3.0 Menu manager selects <i>current menu</i> option from the home screen. 4.0 COS displays list of menus on the screen 5.0 Menu manager selects his choice 6.0 COS displays the following options for the selected menu - menu name, menu category, various recipes in the menu and ingredients for the recipe delete this menu. 7.0 Menu manager selects delete this menu option 8.0 COS requests for a reconfirmation for deleting the menu with 'OK'/'Cancel' prompt 9.0 Menu manager confirms by clicking 'OK' 8.0 COS generates an acknowledgement menu successfully deleted.		
Alternative Flow:	8.1 Menu manager selects 'Cancel' 8.2 COS cancels the deleting activity for the selected menu		
Includes:	Menu manager collects various menus from Cafeteria and other restaurants.		
Frequency of Use:	5 – 10 times a day, depending on the need		
Assumptions:	The Menu manager understands English		
Priority	High		

Use Case ID:	UC-15		
Use Case Name:	Recipes and Ingredients list management		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	29/4/18

Actors:	Menu Manager
Description:	This use case helps menu manager to delete a menu
Trigger:	Menu manager might feel to delete an old menu and to make a fresh new menu out of it, rather than editing the entire contents
Preconditions:	Menu manager is registered with COS using his employee ID.
Postconditions:	Menu Manager successfully deletes a menu from the menu list
Normal Flow:	1.0 Menu manager opens COS from his device. 2.0 COS displays <i>home screen</i> for menu manager including – <i>create menu, current menu list, update ingredients and recipes list</i> 3.0 Menu manager selects recipes and ingredients list 4.0 Cos displays recipes and ingredients screen including recipe list option, ingredients list option 5.0 Menu manager selects ingredients list option 6.0 COS displays ingredients list screen 7.0 Menu manager updates his desired ingredient count and clicks save 8.0 COS displays an acknowledgement intimating 'ingredients list updated'
Alternative Flow:	5.1 menu manager selects recipe list 6.1 COS displays recipes list screen 7.1 Menu manager updates the details of his desired recipe item and clicks save 8.1 COS displays an acknowledgement intimating 'recipe list updated'
Includes:	Menu manager collects various menus from Cafeteria and other restaurants.
Frequency of Use:	5 – 10 times a day, depending on the need
Assumptions:	The Menu manager understands English
Priority	High

Use Case ID:	UC-16		
Use Case Name:	Accept an order		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	29/4/18
Actors:	Meal preparer		
Description:	This use case helps meal preparer to accept an incoming meal order		
Trigger:	Meal preparer accepts the meal requests from the patrons and prepares them.		
Preconditions:	Meal preparer is registered with COS using his employee ID.		
Postconditions:	Meal preparer successfully accepts an order.		
Normal Flow:	1.0 Meal preparer opens COS from his device. 2.0 COS displays <i>home screen</i> for meal preparer including – <i>Accept Order, Current Orders, Completed orders</i> . 3.0 Meal preparer selects <i>accept meal order</i> from the home screen. 4.0 COS generates an acknowledgement on the screen intimating “ <i>Order Accepted</i> ”.		
Alternative Flow:	4.1 COS automatically updates the Order status as “ <i>Meal Processing in Progress</i> ” 4.2 COS also adds the accepted order to the current order’s list.		
Frequency of Use:	50 – 60 times a day for a meal preparer.		
Assumptions:	The Meal processor understands English		
Priority	High		

Use Case ID:	UC-17		
Use Case Name:	Request Delivery		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	29/4/18
Actors:	Meal preparer		
Description:	This use case helps meal preparer to request meal delivery for the completed order		
Trigger:	Meal preparer completes the meal order and request for delivery to the meal deliverer to deliver it to the patron.		
Preconditions:	1. Meal preparer is registered with COS using his employee ID. 2. Meal preparer has accepted the order.		
Postconditions:	Meal preparer successfully requests delivery for the completed meal.		
Normal Flow:	1.0 Meal preparer opens COS from his device. 2.0 COS displays <i>home screen</i> for meal preparer including – <i>Accept Order, Current Orders, Completed orders</i> . 3.0 Meal preparer selects <i>Current orders</i> from the home screen. 4.0 COS displays list of current orders on the screen. 5.0 Meal preparer selects his desired current order. 6.0 COS displays the details of the current order including <i>request delivery</i> option. 7.0 Meal preparer selects request delivery on the screen. 8.0 COS generates an acknowledgement on the screen intimating “ <i>Request delivery sent</i> ”.		
Alternative Flows:	8.1 COS automatically updates the Order status as “ <i>Requested for Delivery</i> ” 6.0 COS also displays the order to the meal deliverer as “ <i>Incoming delivery request</i> ”.		
Frequency of Use:	50 – 60 times a day for a meal preparer.		
Assumptions:	The Meal processor understands English		
Priority	High		

Use Case ID:	UC-18		
Use Case Name:	Accept delivery		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	29/4/18
Actors:	Meal deliverer		
Description:	This use case helps meal deliverer to accept an incoming meal delivery request		
Trigger:	Meal deliverer accepts meal delivery to complete the meal order requested by patron		
Preconditions:	Meal deliverer is registered with COS using his employee ID.		
Postconditions:	Meal deliverer accepts the meal requests from the meal preparer and follows instructions from the COS to pick-up the meal from the restaurant.		
Normal Flow:	1.0 Meal deliverer opens COS from his device. 2.0 COS displays <i>home screen</i> for meal deliverer including – <i>Accept Meal delivery, Current Deliveries, Completed Deliveries</i> 3.0 Meal deliverer selects <i>accept meal delivery</i> from the home screen. 4.0 COS generates an acknowledgement on the screen intimating “ <i>Meal</i>		

	<i>delivery accepted</i> ".
Alternative Flow:	4.1 COS automatically updates the Order status as <i>"Meal Delivery in progress – Waiting for meal pickup"</i> 4.2 COS also adds the accepted meal delivery to the current delivery list. 4.3 COS provides the instructions to the deliverer for picking up the meal from the restaurant.
Frequency of Use:	50 – 60 times a day for a meal deliverer.
Assumptions:	The Meal deliverer understands English
Priority	High

Use Case ID:	UC-19		
Use Case Name:	Meal pickup		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	29/4/18
Actors:	Meal deliverer		
Description:	This use case helps meal deliverer to accept an incoming meal delivery request and pick-up the delivery from the restaurant		
Trigger:	Meal deliverer accepts meal delivery to complete the meal order requested by patron		
Preconditions:	Meal deliverer is registered with COS using his employee ID.		
Postconditions:	Meal deliverer accepts the meal requests from the meal preparer and follows instructions from the COS to pick-up the meal from the restaurant.		
Normal Flow:	1.0 meal deliverer opens COS application from his device. 2.0 COS displays <i>home screen</i> for meal deliverer including – <i>Accept Meal delivery, Current Deliveries, Completed Deliveries</i> 3.0 Meal deliverer selects current deliveries on the home screen 4.0 COS displays the list of current deliveries 5.0 Meal deliverer selects the specific delivery he has picked up 6.0 COS displays the details for the delivery including meal pickup option. 7.0 Meal deliverer selects meal pickup option. 8.0 4.0 COS generates an acknowledgement on the screen intimating <i>"Meal order successfully picked-up"</i> .		
Alternative Flow:	8.1 COS automatically updates the Order status as <i>"Order out for delivery"</i> 8.2 COS provides the instructions to the deliverer for delivering the meal to the Patron.		
Frequency of Use:	50 – 60 times a day for a meal deliverer.		
Assumptions:	The Meal deliverer understands English		
Priority	High		

Use Case ID:	UC-20		
Use Case Name:	Deliver an order		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	29/4/18
Actors:	Meal deliverer		
Description:	This use case helps meal deliverer to accept an incoming meal delivery request and pick-up the delivery from the restaurant		
Trigger:	Meal deliverer accepts meal delivery to complete the meal order requested by patron		
Preconditions:	Meal deliverer is registered with COS using his employee ID.		
Postconditions:	Meal deliverer accepts the meal requests from the meal preparer and follows instructions from the COS to pick-up the meal from the restaurant.		
Normal Flow:	1.0 meal deliverer opens COS application from his device. 2.0 COS displays <i>home screen</i> for meal deliverer including – <i>Accept Meal delivery, Current Deliveries, Completed Deliveries</i> 3.0 Meal deliverer selects current deliveries on the home screen 4.0 COS displays the list of current deliveries 5.0 Meal deliverer selects the specific delivery he has delivered 6.0 COS displays the details for the delivery including <i>delivered</i> option. 7.0 Meal deliverer selects <i>delivered</i> option. 8.0 4.0 COS generates an acknowledgement on the screen intimating “ <i>Meal order successfully delivered</i> ”.		
Alternative Flow:	8.1 COS automatically updates the Order status as “ <i>Delivered</i> ”		
Frequency of Use:	50 – 60 times a day for a meal deliverer.		
Assumptions:	The Meal deliverer understands English		
Priority	High		

Use Case ID:	UC-21		
Use Case Name:	Generate Cafeteria activity reports		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	4/5/18
Actors:	Corporate Manager		
Description:	This use case helps corporate manager to generate cafeteria ordering system reports		
Trigger:	Corporate manager fetches activity reports from Cafeteria to know about the employee’s progress, number of orders taken, calculating the operating costs		
Preconditions:	Corporate manager is registered with COS using employee id		
Postconditions:	Corporate manager fetches COS reports		
Normal Flow:	1.0 User opens the COS application. 2.0 COS displays home screen including Order feedback, Generate report option 3.0 Manager selects generate reports option 4.0 COS displays report generation home screen with options – monthly, weekly, yearly report 5.0 Manager selects his choice and clicks generate.		

	6.0 COS automatically downloads reports in excel format.
Frequency of Use:	Approximately 2 times a week.
Assumptions:	The End user understands English
Priority	High

Use Case ID:	UC-22		
Use Case Name:	View Feedback		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	4/5/18
Actors:	Corporate Manager		
Description:	This use case helps corporate manager to view various feedback		
Trigger:	Corporate manager keeps a track of each and every order feedback made by the patron to understand the customer's pulse.		
Preconditions:	Corporate manager is registered with COS using employee id		
Postconditions:	Corporate manager views various records and marks a bookmark with few.		
Normal Flow:	1.0 Corporate manager opens the COS application. 2.0 COS displays home screen including Order feedback, bookmarked feedbacks Generate report option 3.0 Corporate Manager selects order feedback with date limit. 4.0 COS displays various list of order feedback by the users with the selected date limit 5.0 Corporate Manager selects an order feedback from the displayed list of orders 6.0 COS displays the feedback details 7.0 Corporate manager selects bookmark this feedback. 6.0 COS displays an acknowledgement intimating ' <i>bookmark successfully saved</i> '		
Frequency of Use:	Approximately 2 times a week.		
Assumptions:	The End user understands English		
Priority	High		

Use Case ID:	UC-23		
Use Case Name:	Deduce payroll		
Created By:	Deepak Mayanattanmy	Last Updated By:	Deepak Mayanattanmy
Date Created:	8/3/18	Last Revision Date:	4/5/18
Actors:	Payroll processor		
Description:	This use case helps the payroll processor to a deduce a payroll.		
Trigger:	Payroll processor deduces payroll while the user pays for the order		
Preconditions:	Payroll processor is registered with COS using employee id		
Postconditions:	Payroll processor deduces payroll successfully from the user		
Normal Flow:	1.0 User opens the COS application. 2.0 COS displays home screen including Accept payroll, payroll processed list 3.0 Manager selects accept payroll option 4.0 COS displays accept payroll screen 5.0 Payroll processor verifies the user data and accepts payment from user 6.0 COS displays an acknowledgement intimating ' <i>payroll processed.</i> '		
Frequency of Use:	Approximately 100-150 times a day		

Assumptions:	Payroll processor understands English
Priority	High

4. EXTERNAL INTERFACE REQUIREMENTS

4.1 USER INTERFACES

➤ WELCOME SCREEN:

- Welcome screen displays the cafeteria logo alongside with a green patch.
- This welcome screen will be displayed whenever a new user opens COS application

FIGURE 4.1 *Welcome Screen*



➤ **LOGIN/REGISTRATION SCREEN:**

Figure 4.2 Login/registration screen for both Patron and Employee

The image displays two side-by-side login/registration screens for a system called 'Cafeteria'. Both screens have a blue header with the 'Cafeteria' logo and title.

Left Screen: SIGN IN/REGISTER

- Field: EMAIL/PHONE NUMBER/COS ID: (text input)
- Field: PASSWORD: (text input)
- Link: [Forgot Password?](#)
- Button: SIGN IN
- Text: Not registered yet?
- Button: SIGNUP/SIGNUP WITH 
- Text: OR
- Button: EMPLOYEE LOGIN

Right Screen: EMPLOYEE LOGIN/REGISTER

- Field: EMPLOYEE ID: (text input)
- Field: PASSWORD: (text input)
- Link: [Forgot Password?](#)
- Button: SIGN IN
- Text: OR
- Button: NEW EMPLOYEE REGISTRATION

There are two login/register screens,

1. Patron login/registration screen

- Patrons can sign up or sign in using Facebook
- Patrons can provide their phone number or email ID or COS ID for signing in with COS.
- Patrons can sign-in/sign-up using their Facebook profile.

2. Employee login/Registration screen

- Employees shall use employee login for logging in with COS.
- Employees shall provide their employee ID and password for signing in with COS.

➤ **FEEDBACK INTERFACE SCREEN FOR PATRON:**

Figure 4.3 Feedback Interface Screen



- Make a feedback interface will be available to the user only after the meal delivered to the Patron.
- Patron can make a feedback about the order by rating the food quality, rating the delivery service and rating the order experience through COS application.

➤ **SHARE AN ACTIVITY IN FACEBOOK INTERFACE:**

Figure 4.4 Share an activity Interface



- Patron can share an activity in Facebook by selecting an order under ‘Order History’ label.
- Patron shall authorize his Facebook profile to share the activity.

➤ **SELECTING DEFAULT CUISINE SCREEN:**

- Patrons can select their default cuisine by selecting an of the listed cuisine under ‘Account settings’ label.
- Based on the default cuisine preference, Patron’s menu will be displayed.

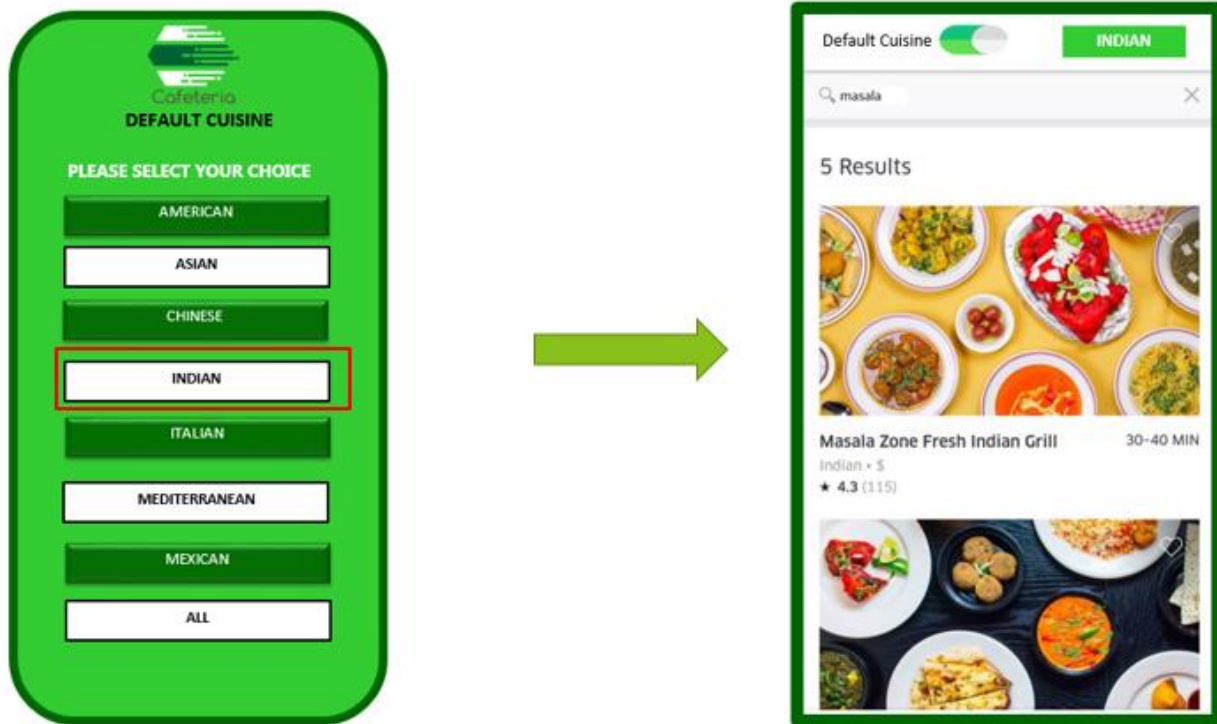
There are three different default cuisine interfaces, they are

1. Default cuisine screen with a selected cuisine preference:

- If a default cuisine is selected, say Indian cuisine, the user will be shown, his default cuisine is Indian which the user is surfing the menu screen for placing the meal order.
- In addition to this, user can de-select default cuisine at any time by clicking the default cuisine button at the menu screen while placing a meal order.

- The below image displays the default cuisine selection process.

Figure 4.5 Default cuisine selection process:



2. Default cuisine screen without cuisine preference:

- Patron can place a meal order with having a default cuisine option.
- If a patron has not selected any cuisine preference, the default cuisine will be set to 'ALL'
- 'ALL' cuisine intimates that, Patron is not having any cuisine preference and he will be shown will a complete list of menus.

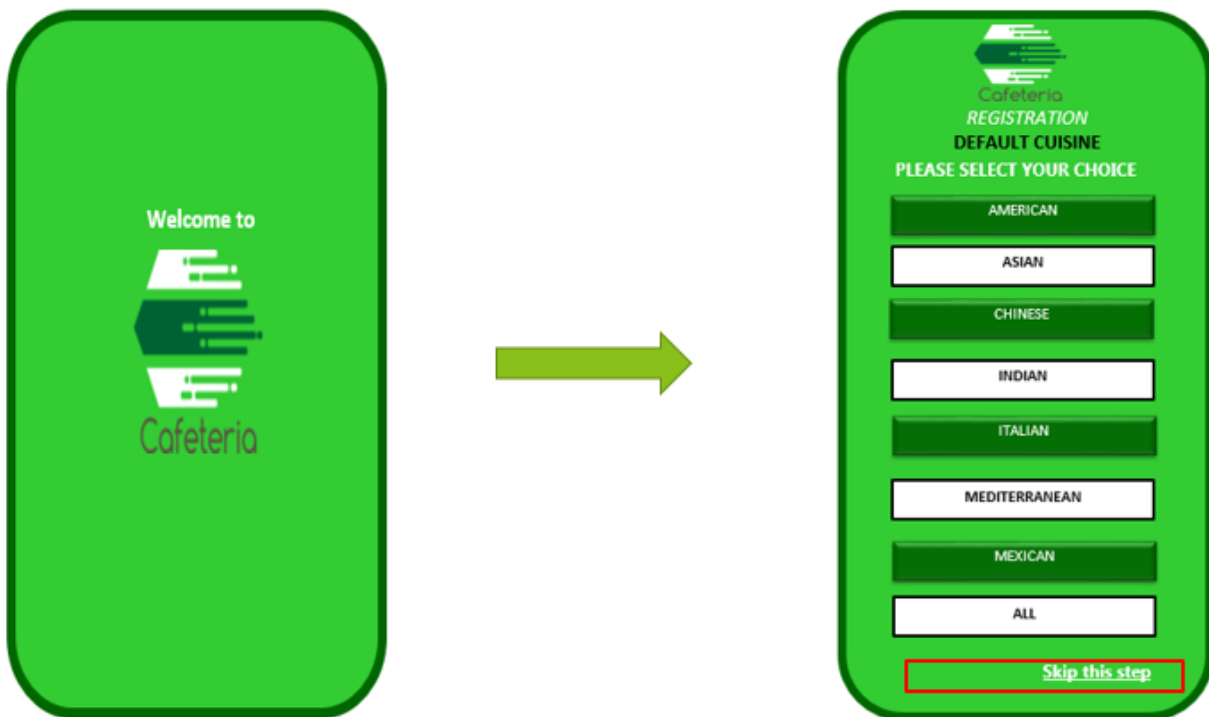
Figure 4.6 Default cuisine without any preference



3. Default cuisine Registration Screen:

- Patrons can select default cuisine at the time of registration.
- Patron can select default cuisine from the list of available options.
- If a patron selects 'skip this step' option, the default cuisine will be set to 'ALL'
- 'ALL' cuisine intimates that, Patron is not having any cuisine preference and he will be shown will a complete list of menus.
- Figure 4.7 displays the interface for default cuisine at the time of registration.

Figure 4.7 Default cuisine registration screen



➤ MEAL PREPARER INTERFACE:

Figure 4.8 Meal preparer Interface



Description:

- Meal preparer logs in through COS using his employee ID
- Meal preparer can accept an order, request for the delivery and can track his previous orders.

➤ MEAL DELIVERER'S INTERFACE

Figure 4.9 Meal deliverer's interface



Description:

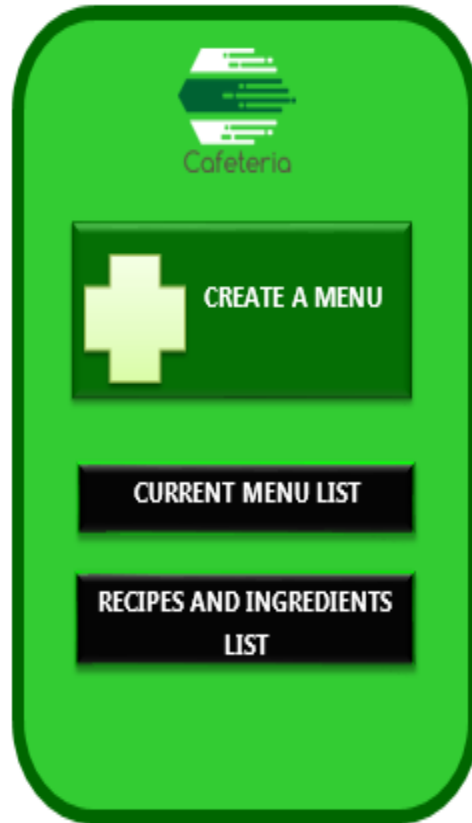
- Meal deliverer logs in through COS using his employee ID
- Meal deliverer can accept a new delivery, request for the delivery and can track his previous orders.

➤ MENU MANAGER INTERFACE:

Description:

- Menu Manager logs in through COS using his employee ID
- Meal manager has three main options, creating a menu, editing a current menu, and managing the recipes and ingredients list

Figure 4.10 Menu Manager's interface



➤ **CORPORATE MANAGER'S INTERFACE:**

The corporate manager mainly has two interfaces they are

1. Generate report interface

- Corporate Manager logs in with COS application using employee ID
- Corporate Manager selects from and to date for which he wishes to generate reports
- Post selecting the date, he downloads the report by selecting download PDF button.
- Figure 4.11 displays corporate manager's generate reports interface.

2. View Feedback Interface

- Corporate Manager selects from and to date for which he wishes to view the Feedbacks made by the Patrons.
- Post selecting the date, the corporate Manager can view the feedback by selecting 'View' option.
- Figure 4.12 displays corporate manager's view feedback interface.

Figure 4.11 Generate Report's Interface

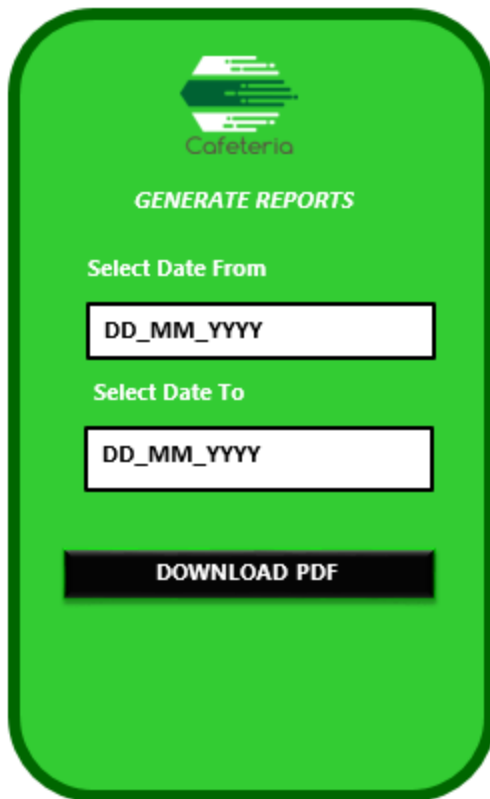
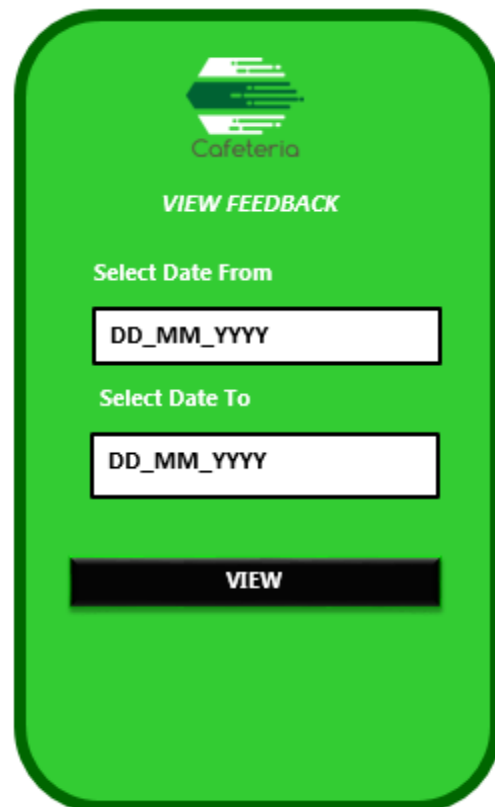


Figure 4.12 View Feedback Interface



4.2 HARDWARE INTERFACES

Our software need to know the distance between user and restaurant which he ordered, so our software should get geographical information about where the user is located by using the GPS inside mobile or computer and other than that we will not use any hardware interfaces.

4.3 SOFTWARE INTERFACES

Users can login into our system by using their existing Facebooks accounts. We use JavaScript SDK for web application and Android/iOS SDK for mobile application to enable people to sign into our application with Facebook login. When user clicks on “login with Facebook”, our system redirects to FB login page and submit user’s FB login credential. When user log into our application with Facebook they can grant permissions to our application, so we can retrieve information on their behalf.

4.4 COMMUNICATIONS INTERFACES

We use java programming language and SOA design pattern to build our system. The communication between the different parts of the system is finished by using HTTPS protocol. In addition, we use JSON as our system’s data-interchange format because JSON is easy for humans to read and write.

We should build a database server and separate reads and writes and scale our database by using master-slave replication. The master can handle both reads and writes while the slaves handle only reads. The slave then replicates any write statements finished on the master within 100ms.

5. OTHER NONFUNCTIONAL REQUIREMENTS

5.1 PERFORMANCE REQUIREMENTS

Since User Experience (UX) is critical to the success or failure of our system in the market and performance is UX, we should a strict requirement on our system's performance.

PER-1: The system should support more than 1000 user to checkout at the same time.

PER-2: The response time of HTTP interfaces should be less than 1 second.

PER-3: When the user request data by click on search button, searching result shall be presented on the screen within no more than 2 seconds.

5.2 SAFETY REQUIREMENTS

SAF-1: We should highlight spicy foods and high calorie foods in the menu in case users order the foods that they don't want.

5.3 SECURITY REQUIREMENTS

SEC-1: User's personally information like phone number and credit card information should be encrypted before storing in databases.

5.4 SOFTWARE QUALITY ATTRIBUTES

SQA-1: The system should be available 24/7.

SQA-2: The Android APK size should less than 50M.

5.5 MARKETABILITY REQUIREMENTS

MAK-1: The Cafeteria Ordering System shall enable patron to share Cafeteria activity with Facebook. Enabling the user to share the Cafeteria activity in Facebook builds the Cafeteria reputation and promotes marketability feature between various users.

6. OTHER REQUIREMENTS

OR-1: The user can set the mobile application to his/her preferred language. (English or Chinese)

OR-2: We should use cache to speed up our application.

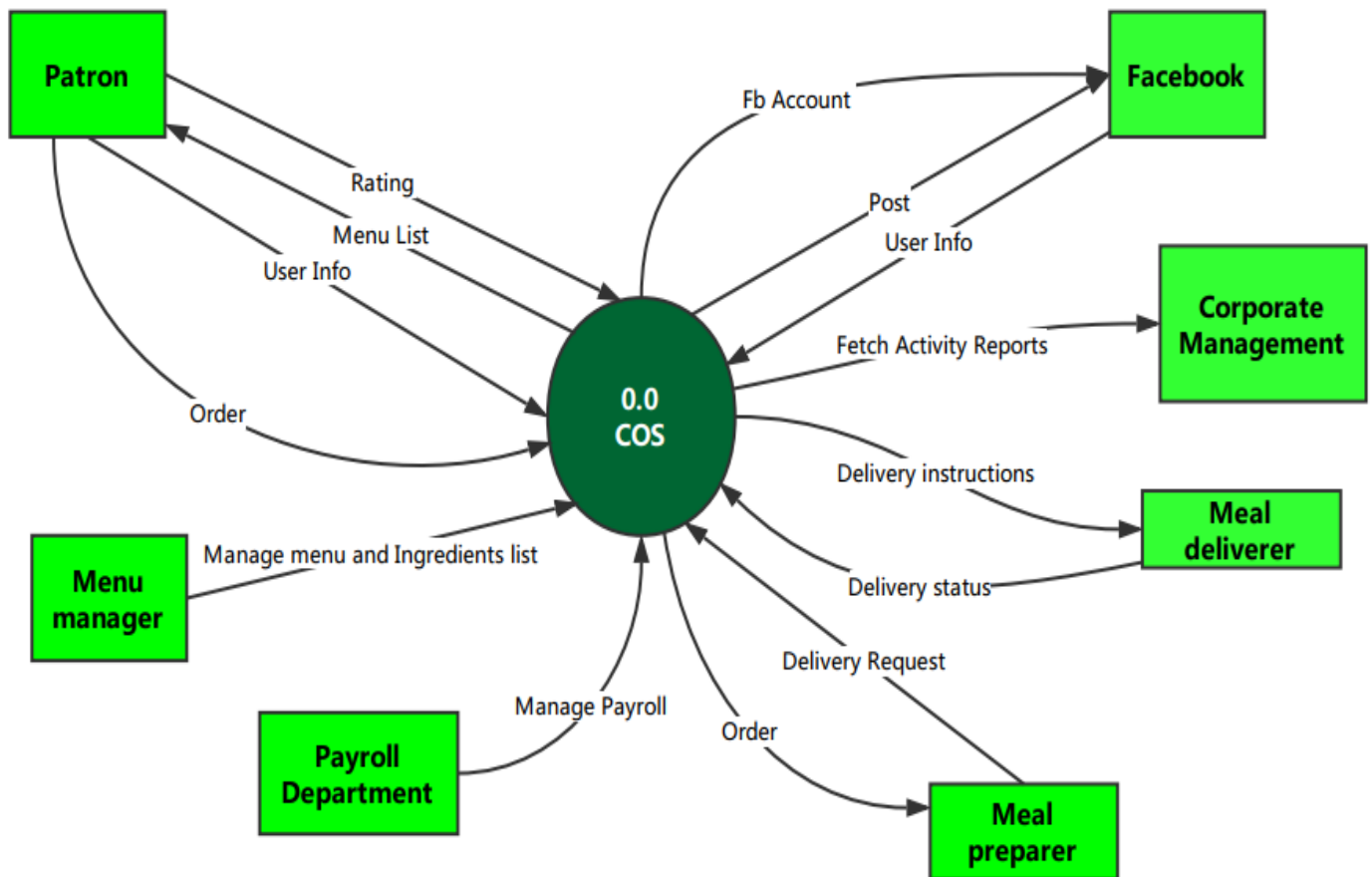
PART - II

7. DATA FLOW DIAGRAM

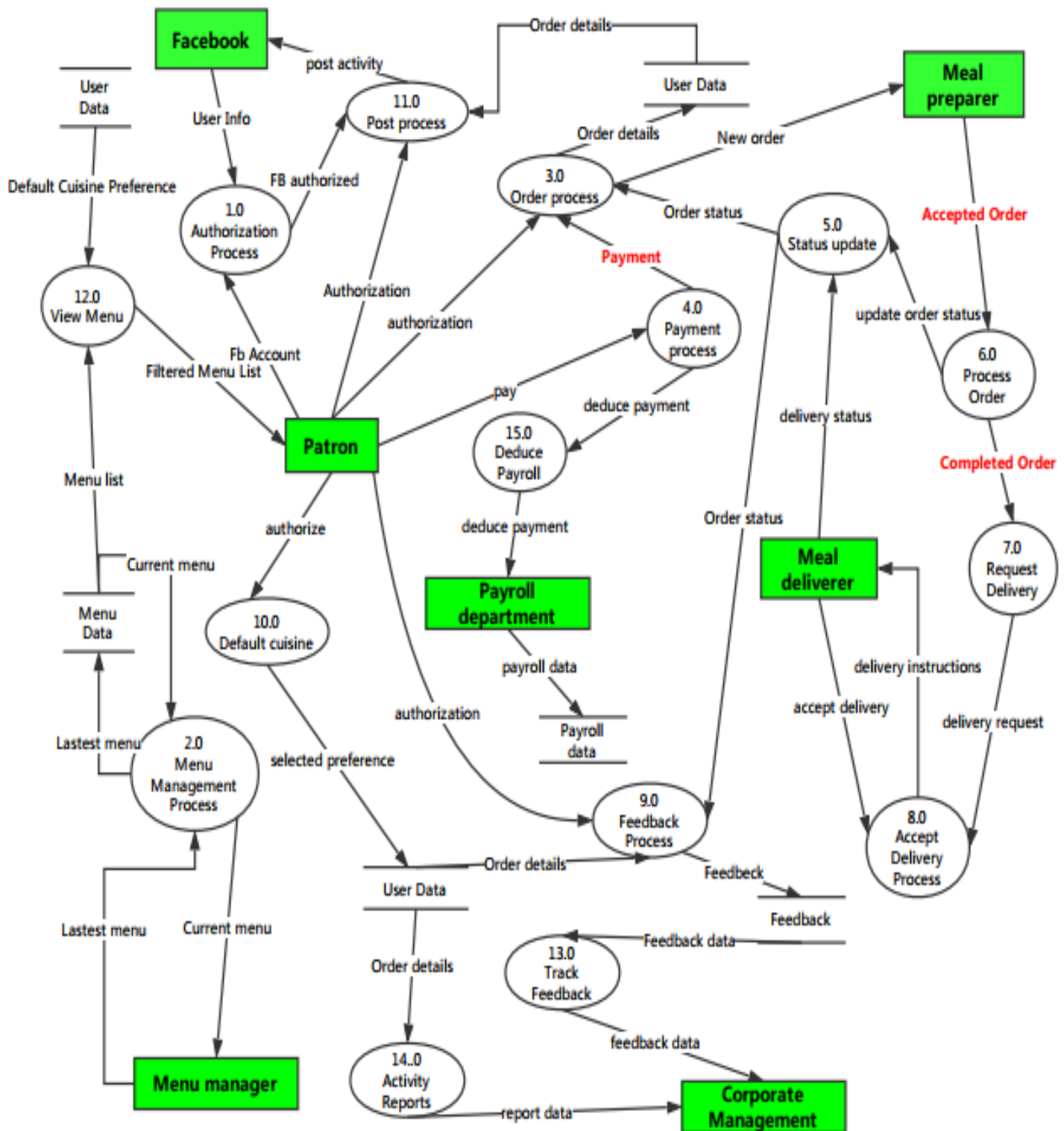
There are three levels of Data Flow Diagrams, they are as follows,

7.1 LEVEL 0 DATA FLOW DIAGRAM

The level 0 Data Flow diagram describes the bird eye view of Cafeteria Ordering System.



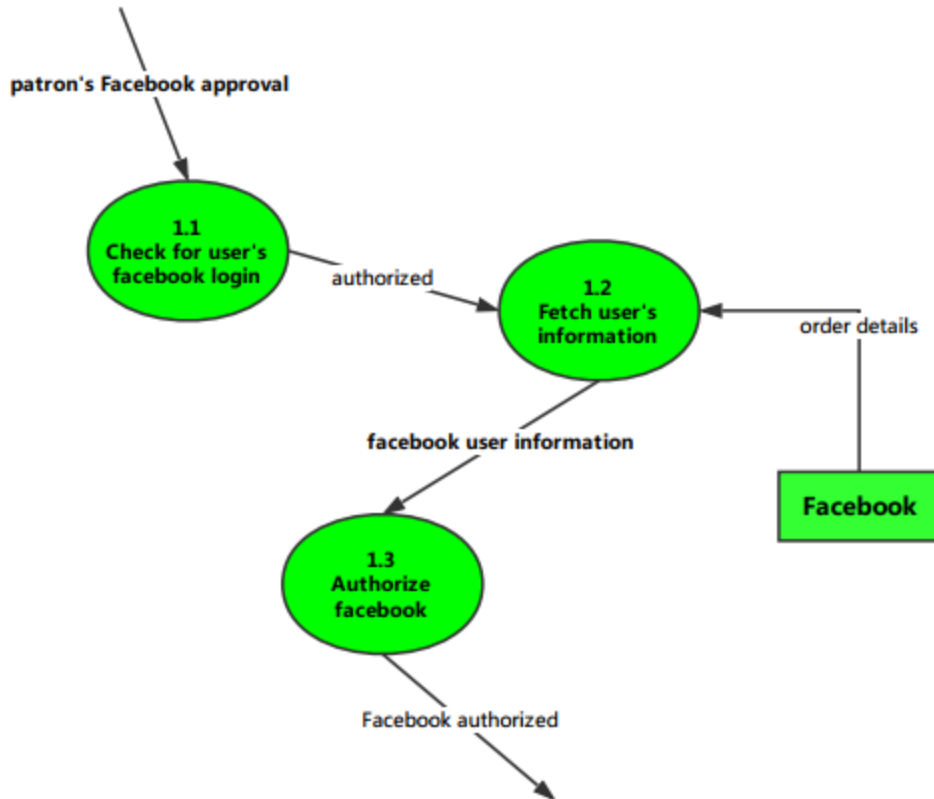
7.2 LEVEL 1 DATA FLOW DIAGRAM



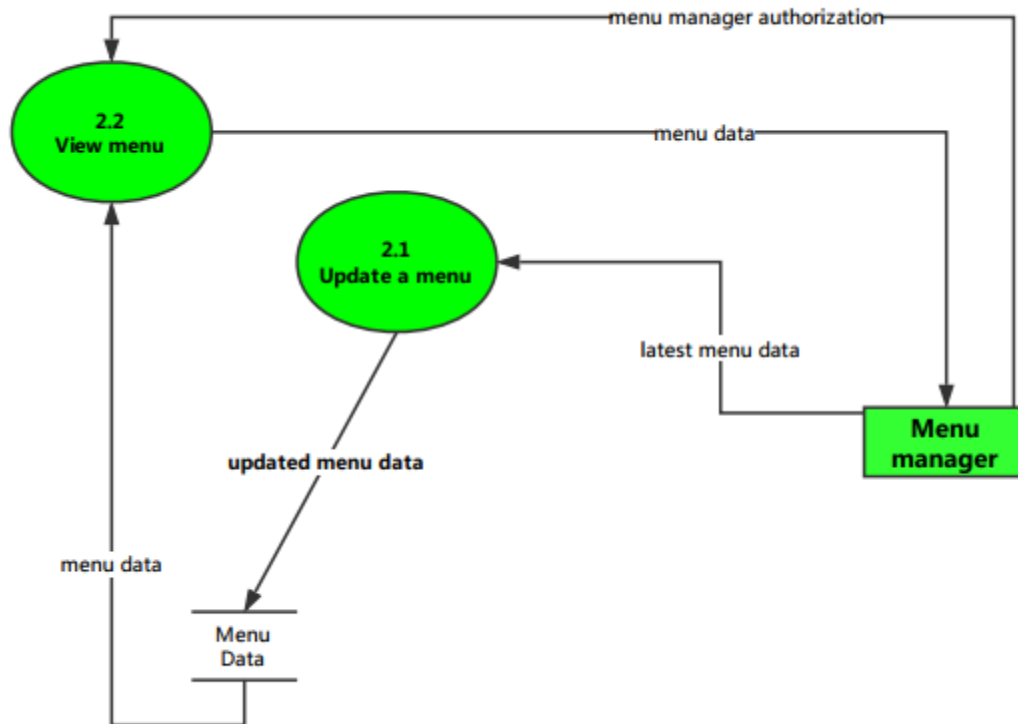
7.3 LEVEL 2 DATA FLOW DIAGRAMS:

Level 2 Data flow diagrams explains the various process in level 1 Data Flow diagrams in detail.

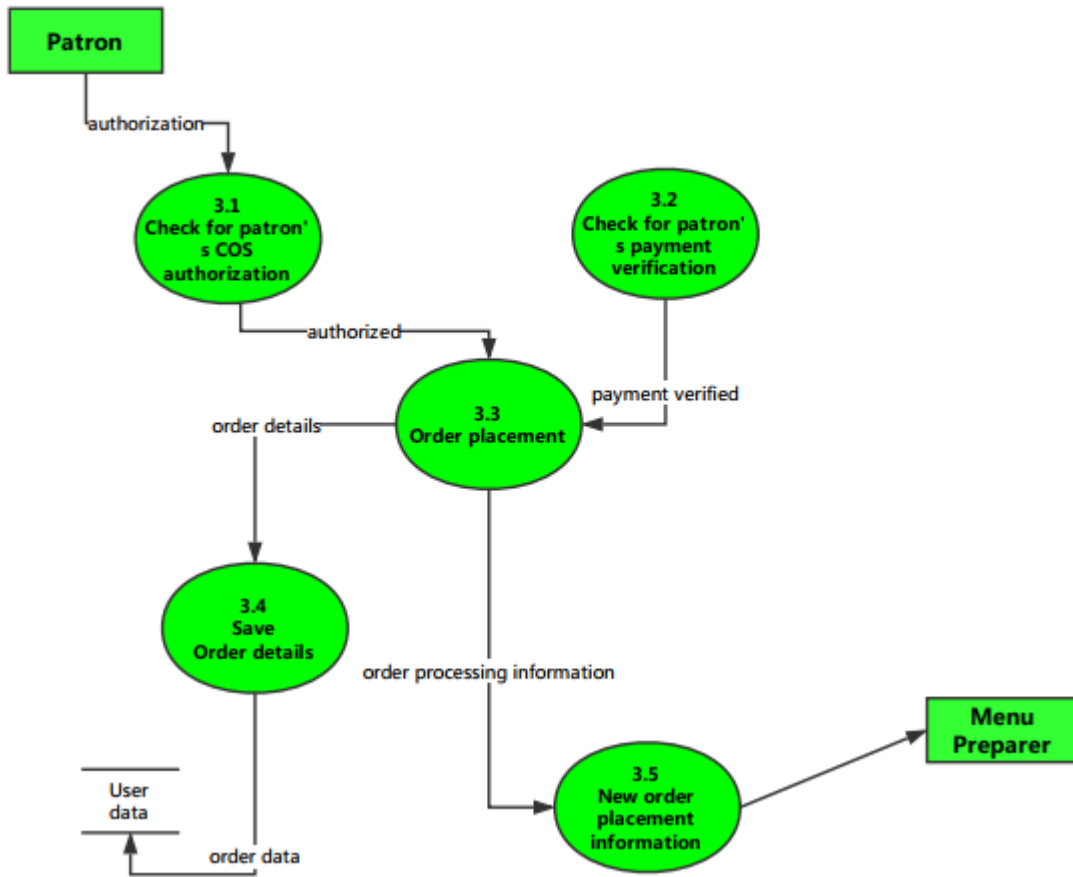
1.0 Facebook authorization process



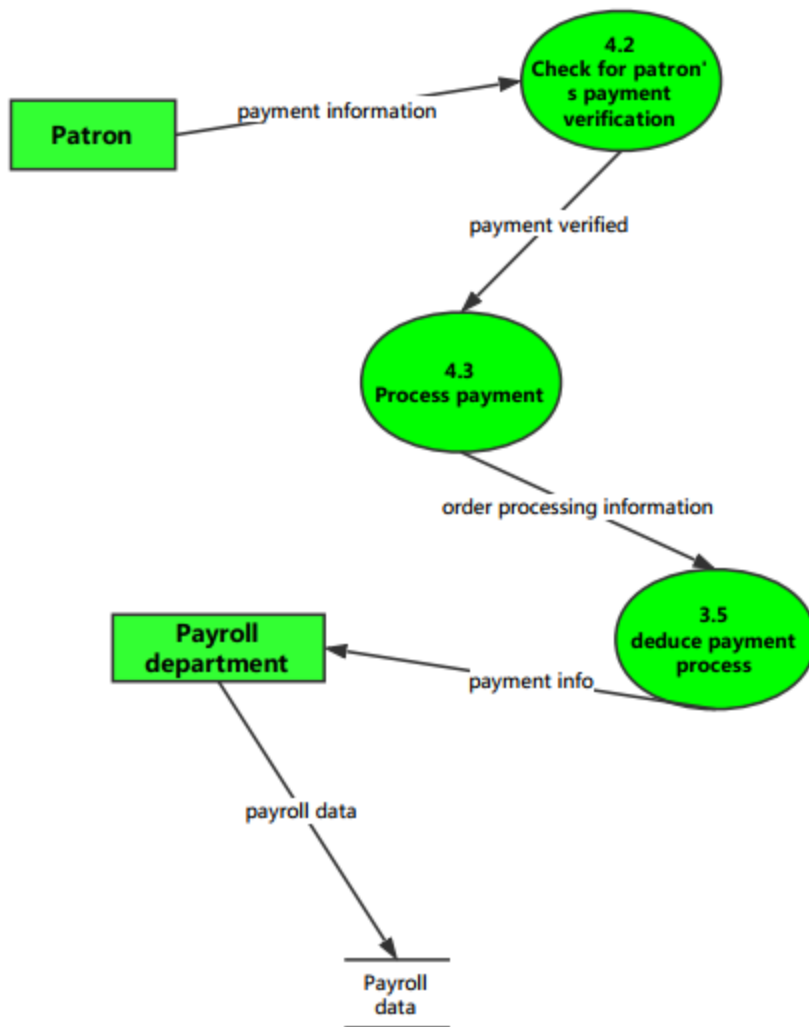
2.0 Menu management process



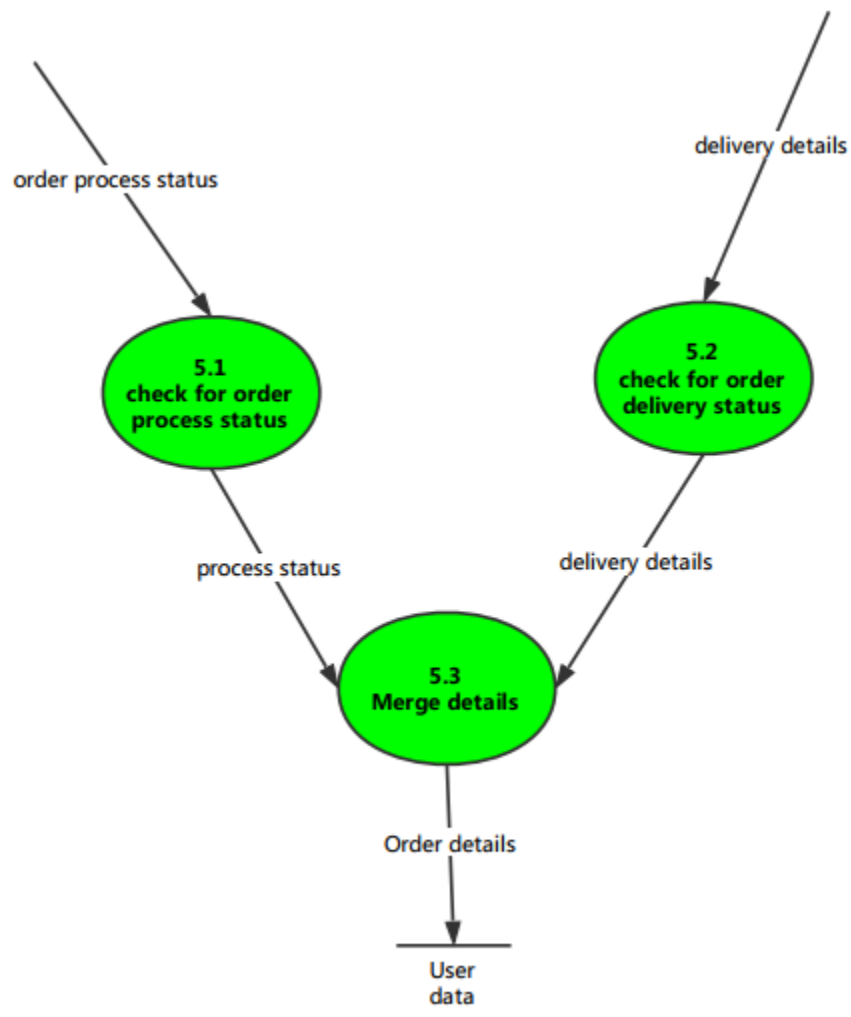
3.0 Order process



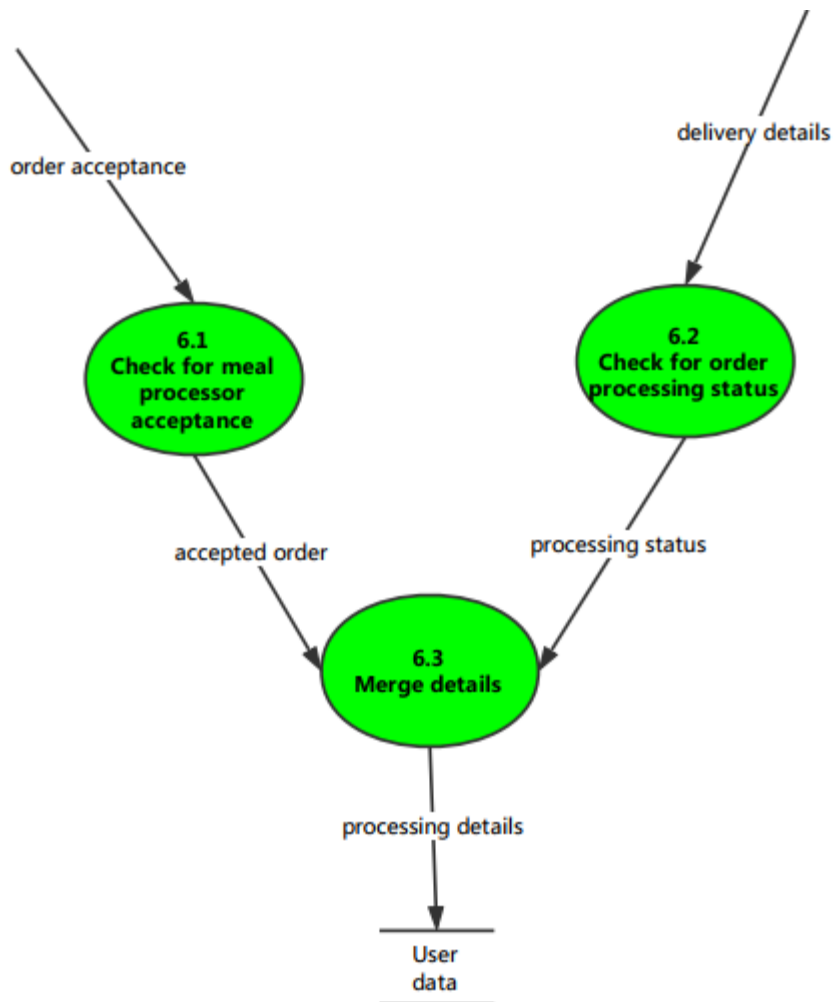
4.0 Payment process



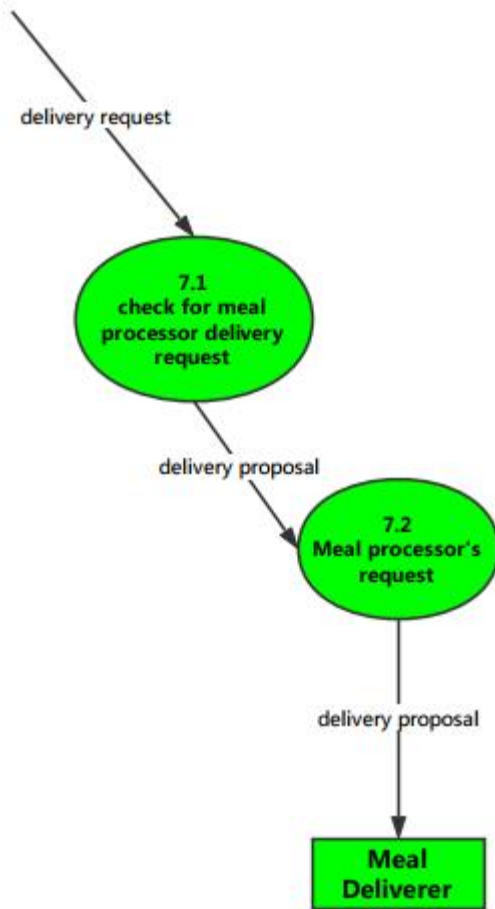
5.0 Order status process



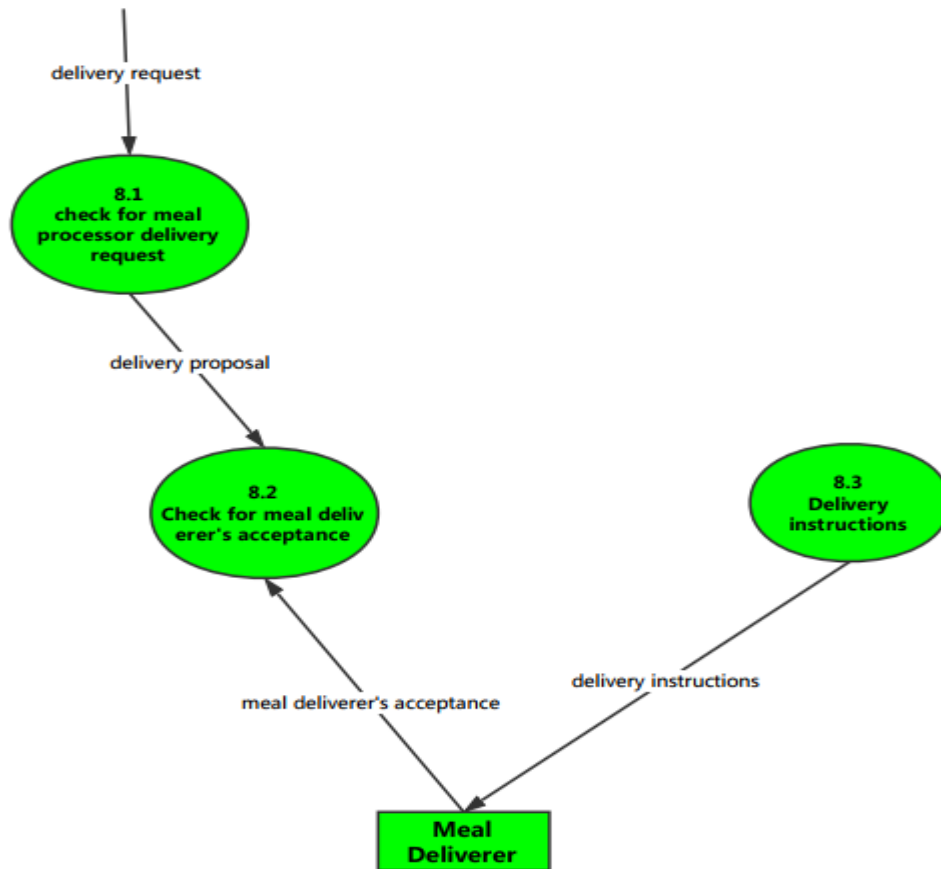
6.0 Process order diagram



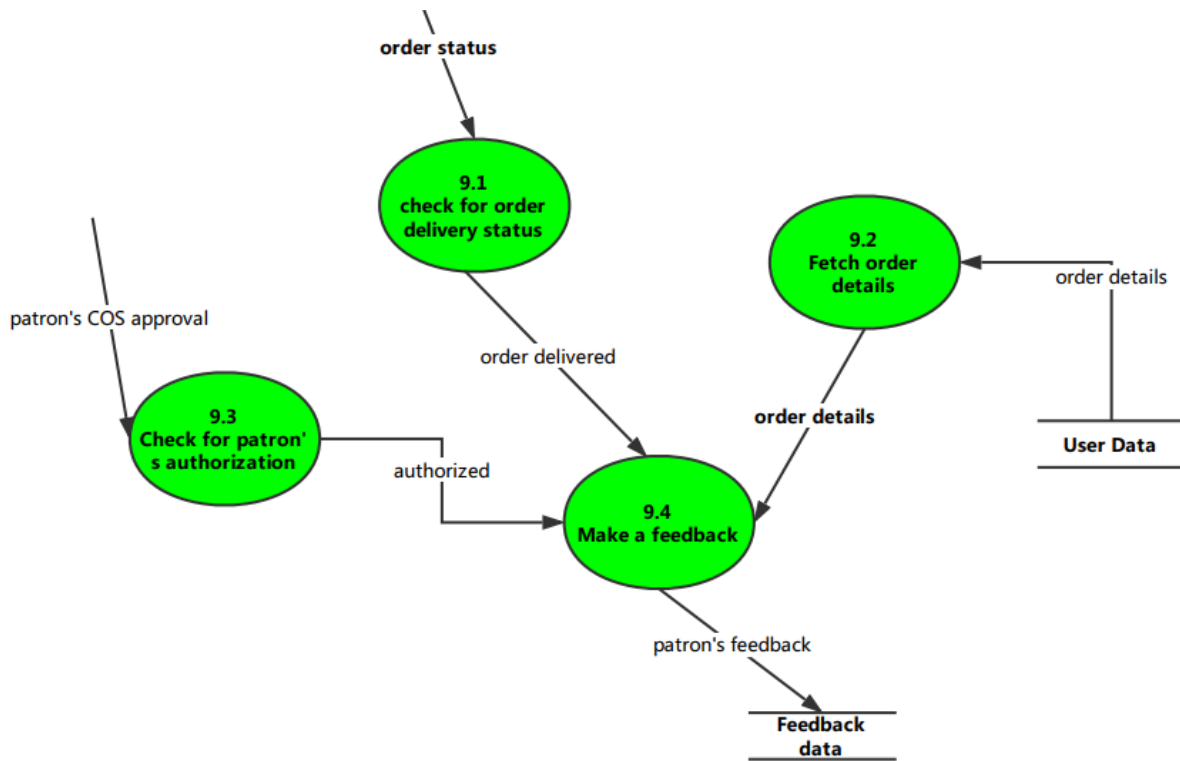
7.0 Delivery proposal



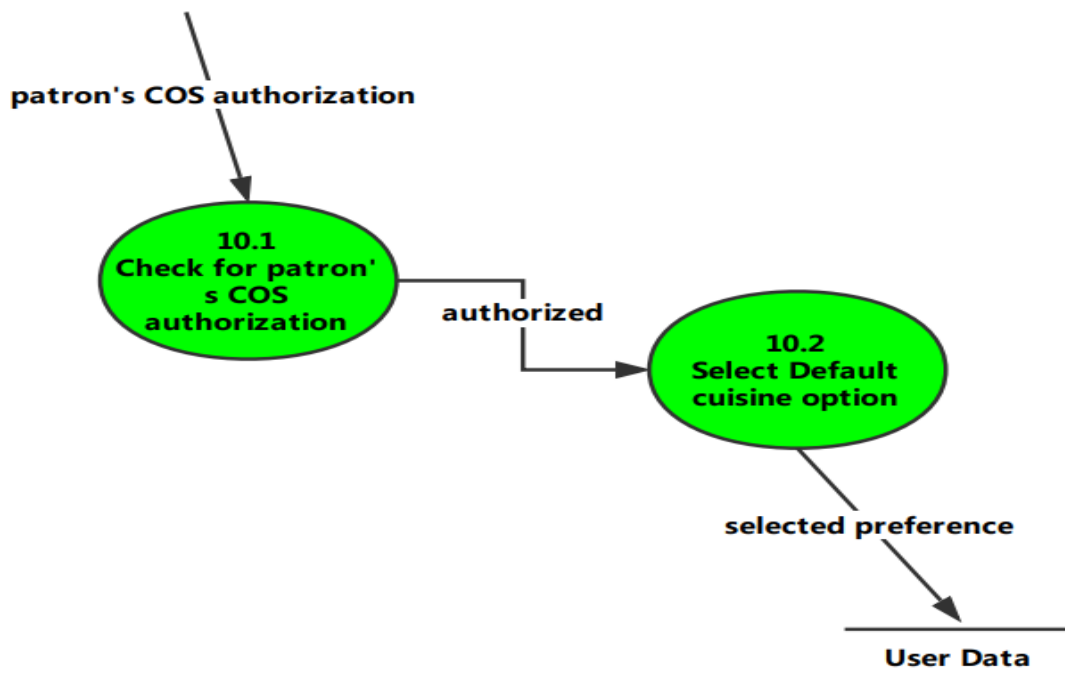
8.0 Meal Acceptance process



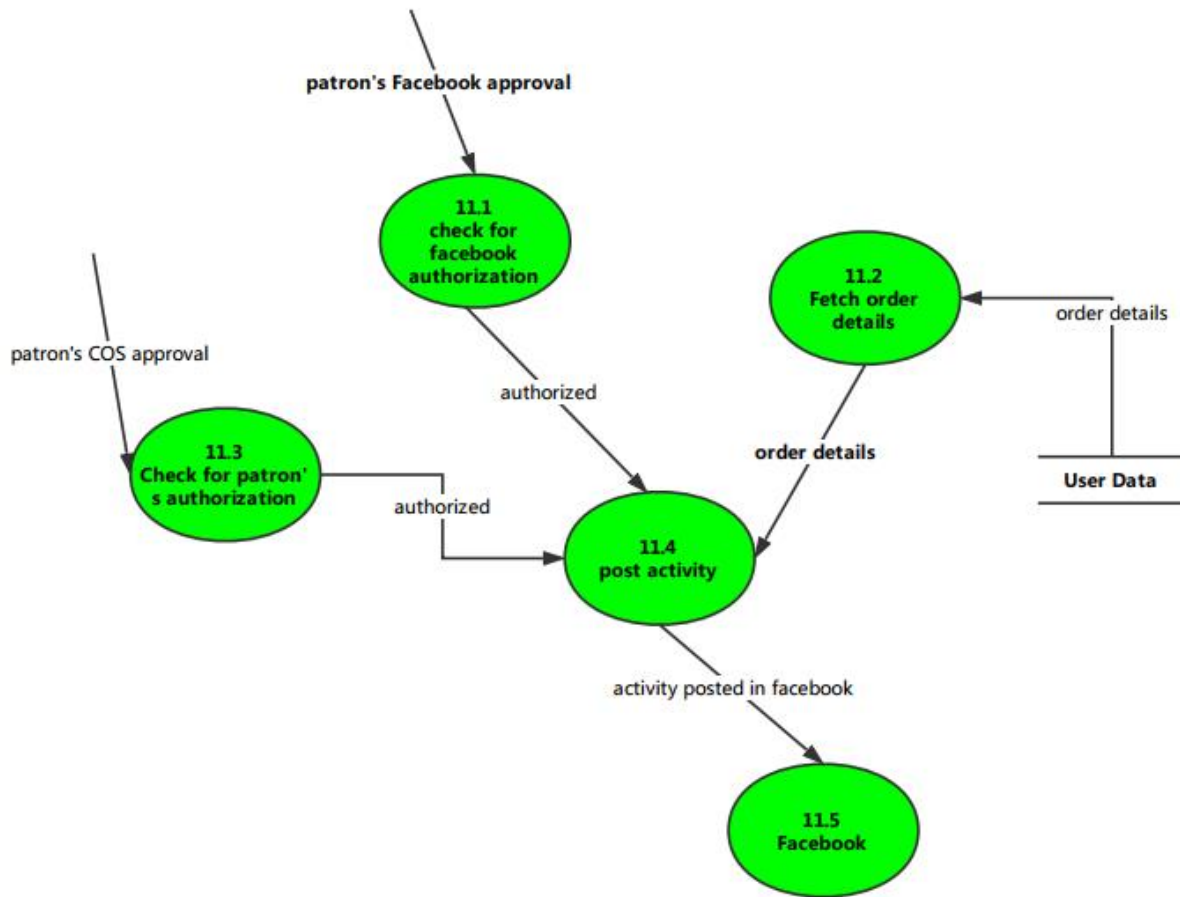
9.0 Make a feedback



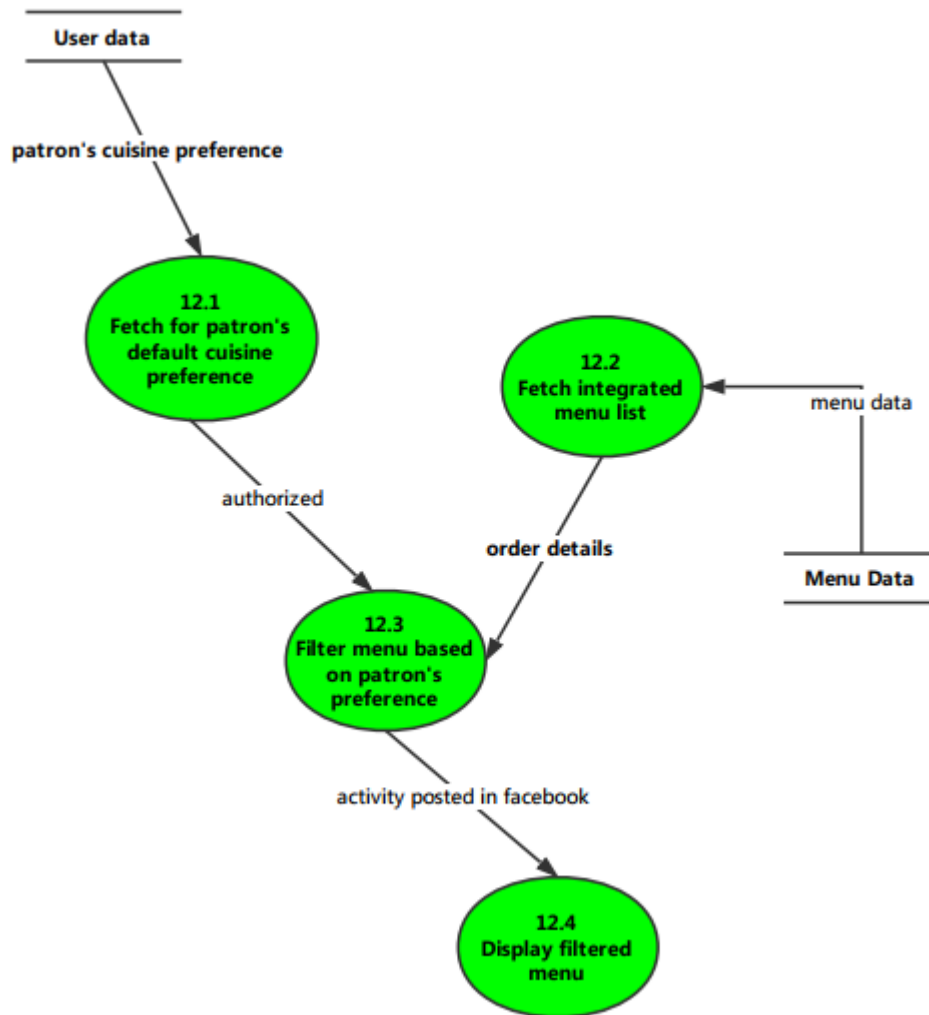
10.0 Setting default cuisine



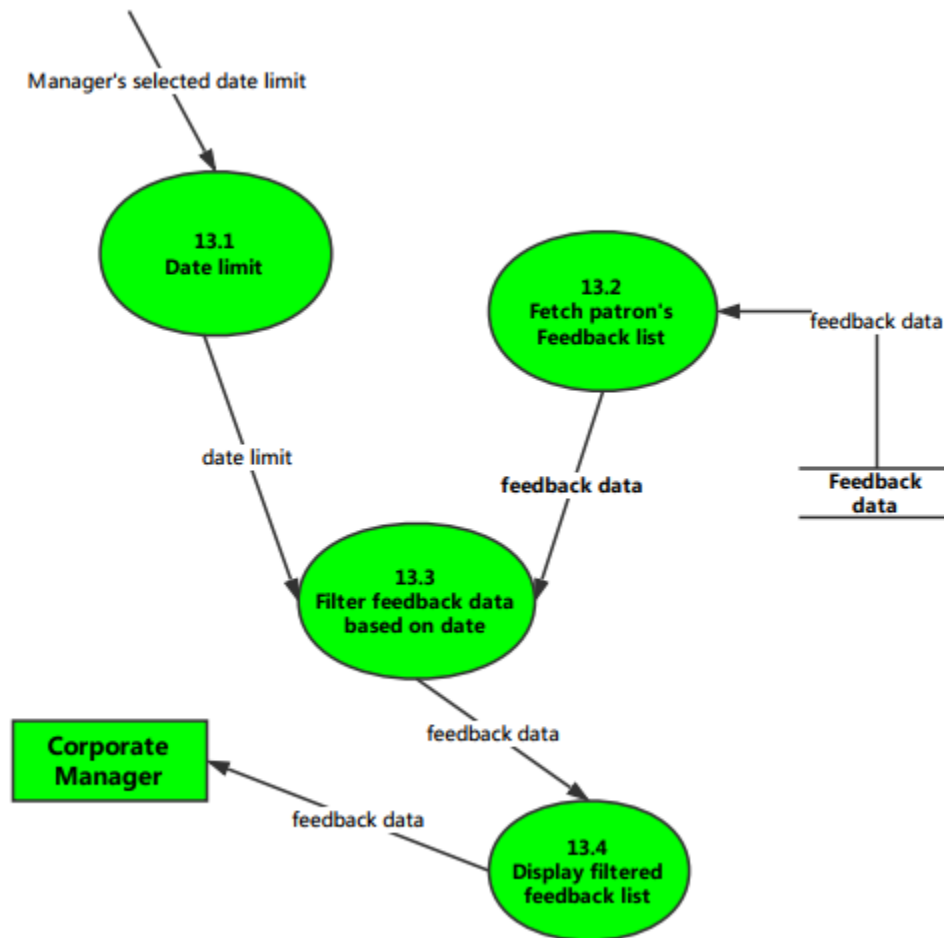
11.0 Share an activity in Facebook



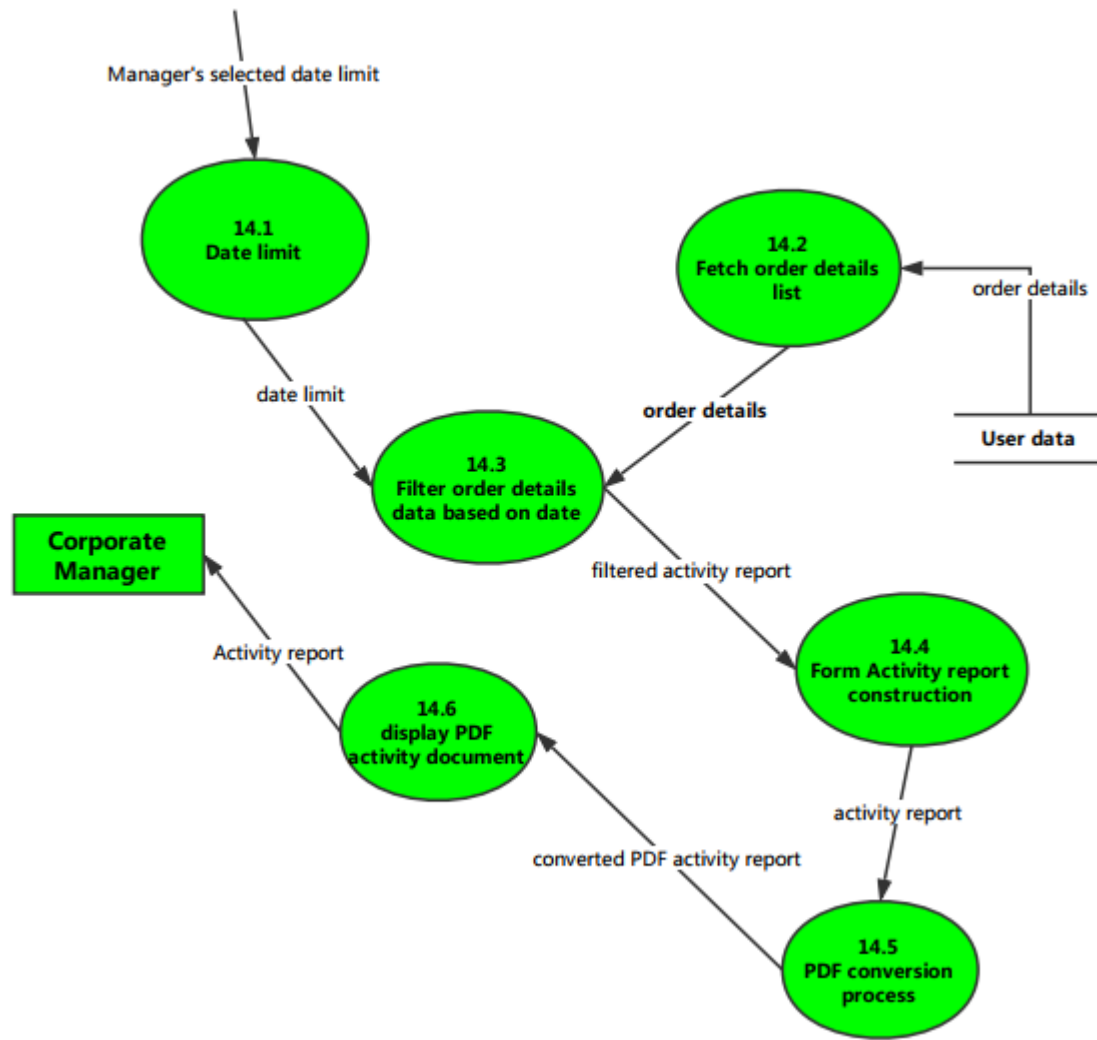
12.0 View Menu process



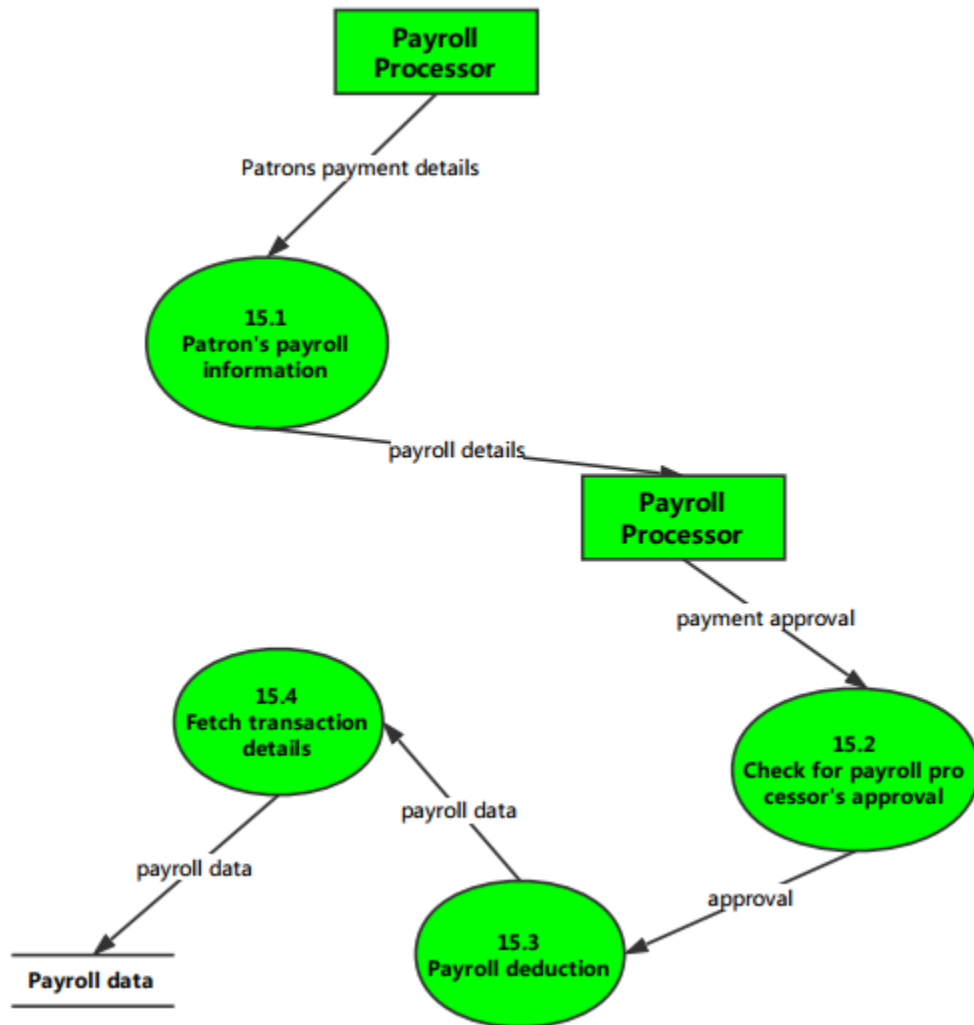
13.0 View Feedback process



14.0 Generate activity reports



15.0 Payroll process



8. FUNCTIONAL REQUIREMENTS

8.1 <Functional Requirement #1> - Order meals from the cafeteria menu to be picked up or delivered

8.1.1 Introduction

This Functional Requirement allows the user to place a meal order from cafeteria menu and to select his meal delivery preference.

8.1.2 Inputs

User should be registered with COS. User selects his favorite meal and the meal quantity from the menu

8.1.3 Processing

User pays the money through the registered card details and the payments will be processed by the customer's bank organization.

8.1.4 Outputs

E-receipt and SMS receipt will be generated as a token of order confirmation.

8.1.5 Error Handling

COS displays an error message if the user's card details are incorrect or expired and cancels the transaction.

8.2 <Functional Requirement #2> - Order meals from other local restaurants to be delivered

8.2.1 Introduction

This Functional Requirement allows the user to place a meal order from other restaurants and to select his meal delivery preference.

8.2.2 Inputs

User should be registered with COS. User selects his favorite meal and the meal quantity from other restaurants in the menu

8.2.3 Processing

User pays the money through the registered card details and the payments will be processed by the customer's bank organization.

8.2.4 Outputs

E-receipt and SMS receipt will be generated as a token of order confirmation.

8.2.5 Error Handling

COS displays an error message if the user's card details are incorrect or expired and cancels the transaction.

8.3 <Functional Requirement #3> - Create, view, modify, and delete meal service subscriptions

8.3.1 Introduction

This Functional Requirement allows the user to create, view, modify, and delete meal service subscriptions

8.3.2 Inputs

User should be registered with COS.

8.3.3 Processing

User might change the meal subscription by modifying or deleting the current meal service. If user wishes to enroll with a new meal service plan, he can create a new meal service subscription.

8.3.4 Outputs

COS generates an 'Acknowledgement message' intimating the user's activity regarding the meal service plan.

8.3.5 Error Handling

No error handling for this case.

8.4 <Functional Requirement #4> - Register for meal payment

8.4.1 Introduction

This Functional Requirement provides a platform for the users to add his card details for paying the order. Registering Patron's card details in advance comforts the meal ordering process for the patron.

8.4.2 Inputs

Patrons shall have registered with COS

8.4.3 Processing

Patrons can change or add a new card detail under 'Account Settings' label.

8.4.4 Outputs

Account settings label, including Change card details option.

8.4.5 Error Handling

COS shall display an error message if the Patron's card details are invalid.

8.5 <Functional Requirement #5> - Request meal delivery

8.5.1 Introduction

This Functional Requirement allows the user to request for a meal delivery

8.5.2 Inputs

User should be registered with COS

8.5.3 Processing

User may update his delivery options after placing an order. User should provide updated delivery location before the meal has been picked by the meal deliverer.

8.5.4 Outputs

COS displays a 'Delivery location updated successfully' message to the user.

8.5.5 Error Handling

If the order is out for the delivery, COS displays an error message 'Delivery location cannot be updated as the meal is out for the delivery'.

8.6 <Functional Requirement #6> - Create, view, modify, and delete cafeteria menus

8.6.1 Introduction

This Functional Requirement allows the menu manger to create, view, modify, and delete cafeteria menus

8.6.2 Inputs

Menu manager should have been registered with COS using his employee id.

8.6.3 Processing

Menu manager has the authority to create, view, modify, and delete an item from cafeteria menus. Menu manager can display the 'Deal of the day' in the menu screen.

8.6.4 Outputs

COS displays a 'Menu updated successfully' message to the menu manager.

8.6.5 Error Handling

No error handling for this case.

8.7 <Functional Requirement #7> - Order custom meals that aren't on the cafeteria menu

8.7.1 Introduction

This Functional Requirement allows the user order custom meals that aren't on the cafeteria menu.

8.7.2 Inputs

User should be registered with COS.

8.7.3 Processing

User order a new meal that is not available in cafeteria menu. User should provide the style of the food and the ingredients to customize the order.

8.7.4 Outputs

COS displays a 'Customized order successfully placed' message to the user.

8.7.5 Error Handling

COS displays an error message if the user's customized order dint match with cafeteria ingredients.

8.8 <Functional Requirement #8> - Produce recipes and ingredient lists for custom meals from cafeteria

8.8.1 Introduction

This Functional Requirement allows the menu manager to produce recipes and ingredient lists for custom meals from cafeteria

8.8.2 Inputs

Menu manager should have been registered with COS using his employee id.

8.8.3 Processing

Menu manager should maintain the recipes and ingredients list available in cafeteria.

Menu manager tracks the available ingredients and updates it frequently in COS.

8.8.4 Outputs

COS displays a 'Recipe and ingredient list updated successfully' message to the menu manager.

8.8.5 Error Handling

No error handling for this case.

8.9 <Functional Requirement #9> - Provide system access through corporate Intranet or through outside Internet access by authorized employees

8.9.1 Introduction

This Functional Requirement allows the cafeteria employees to access COS outside corporate internet.

8.9.2 Inputs

Cafeteria employees might have registered with COS using their employee id.

8.9.3 Processing

Cafeteria employees should enroll with COS using their employee id. Cafeteria employees should own a personal device such as mobile phone or personal computer to access COS outside their corporate environment.

8.9.4 Outputs

COS displays an 'Employee registration completed successfully' message to the cafeteria employees.

8.9.5 Error Handling

COS displays an error message if the employee id is invalid.

8.10 <Functional Requirement #10> - Login through Facebook

8.10.1 Introduction

This Functional Requirement allows the user to login his COS account through Facebook

8.10.2 Inputs

User should be registered with COS using Facebook profile.

8.10.3 Processing

If the user is accessing his COS account in web browser, the user may login using his COS id and password or through Facebook.

8.10.4 Outputs

COS displays 'Welcome back' message to the user.

8.10.5 Error Handling

COS displays an error message if the user id or password is incorrect.

8.11 <Functional Requirement #11> - Set default Meal preference

8.11.1 Introduction

This Functional Requirement allows the user to set his default meal preference.

8.11.2 Inputs

User must have installed COS application

8.11.3 Processing

Setting up default meal preference is a part of user's initial registration activity with COS. User might wish to order his favorite cuisine rather than surfing the entire menu. For this case, user might set his meal preference, making the application more user-friendly.

8.11.4 Outputs

COS displays a 'registration completed' message to the user.

8.11.5 Error Handling

If the user skips this step during registration, COS assumes the user has no default meal preference.

8. 12 <Functional Requirement #12> - Add my rating

8.12.1 Introduction

This Functional Requirement allows the user rate and review the meal service provided by COS.

8.12.2 Inputs

User should be registered with COS.

8.12.3 Processing

'Add my rating' feature is accessible to user, once the meal delivered to the user successfully. User can rate and review by taste, delivery service, and restaurant.

8.12.4 Outputs

Generate a message intimating 'Thanks for your feedback'.

8.12.5 Error Handling

No error handling for this case.

8. 13 <Functional Requirement #13> - Track order status

8.13.1 Introduction

This Functional Requirement allows the user to track his order status

8.13.2 Inputs

User should be registered with COS.

8.13.3 Processing

Tracking order status shall be enabled as the part of user's order history. User shall be able to track about his order.

8.13.4 Outputs

The various order statuses are order accepted, order under progress, order completed – pending with delivery, order out for delivery, order delivered.

8.13.5 Error Handling

No error handling for this case.

8.14 <Functional Requirement #14> - Register using employee ID

8.14.1 Introduction

This Functional Requirement allows the cafeteria employees to register with COS using employee ID

8.14.2 Inputs

Cafeteria employees shall provide their employee ID and password while logging in with COS

8.14.3 Processing

Cafeteria employees shall register COS application, and shall register with Employee ID with their desired password.

8.14.4 Outputs

COS displays 'Welcome back' message to the user.

8.14.5 Error Handling

COS displays an error message if the employee id or password is not valid

8.15 <Functional Requirement #15> - Login using employee ID

8.15.1 Introduction

This Functional Requirement allows the cafeteria employees to login his COS account using employee ID

8.15.2 Inputs

Cafeteria employees shall be registered with COS using employee ID.

8.15.3 Processing

Cafeteria employees such as menu manager, meal preparer, meal deliverer shall be logging in to the cafeteria application every day to resume their duty.

8.15.4 Outputs

COS displays 'Welcome back' message to the user.

8.15.5 Error Handling

COS displays an error message if the employee id or password is incorrect.

8.16 <Functional Requirement #16> - Maintain recipes and ingredients list

8.16.1 Introduction

This Functional Requirement menu manager to maintain recipes and ingredients list for allowing the user to place a customized meal order

8.16.2 Inputs

Menu manager shall be registered with COS using employee ID.

8.16.3 Processing

Menu manager tracks various ingredients in the cafeteria and updates them in the recipes and ingredients list.

8.16.4 Outputs

COS displays an acknowledgement to the menu manager, 'Recipes and ingredients list updated'

8.16.5 Error Handling

No error handling for this case

8.17 <Functional Requirement #17> - Share an activity in Facebook

8.17.1 Introduction

This Functional Requirement allows the user to share an activity in Facebook

8.17.2 Inputs

User should be registered with COS using Facebook profile or end user shall be registered with Facebook.

8.17.3 Processing

User shares an activity in Facebook, which builds the reputation for Cafeteria restaurant and promotes marketing from user's end.

8.17.4 Outputs

COS shall display 'Activity successfully posted in Facebook.'

8.17.5 Error Handling

COS displays an error message if the user has not authorized his Facebook, intimating 'Activity cannot be posted in Facebook.'

8.18 <Functional Requirement #18> - Provide Delivery instructions

8.18.1 Introduction

This Functional Requirement provides delivery instructions to the meal deliverer post accepting the delivery.

8.18.2 Inputs

Meal deliverer shall be registered with COS using employee ID.

8.18.3 Processing

COS shall provide delivery instructions to the meal deliver such as restaurant location for picking up the meal and the Patron's delivery details and the time constraints for delivering the meal

8.18.4 Outputs

COS shall display the Meal pickup details, Patron's address delivery, time constraints to the accepted meal delivery.

8.18.5 Error Handling

No error handling for this case.

8.19 <Functional Requirement #19> - Generate activity reports

8.19.1 Introduction

This Functional Requirement allows the corporate manager to generate the activity reports

8.19.2 Inputs

Corporate Manager shall be registered with COS using employee ID.

8.19.3 Processing

Corporate manager fetches activity reports from Cafeteria to know about the employee's progress, number of orders taken, calculating the operating costs

8.19.4 Outputs

COS shall display the various activities performed through Cafeteria ordering system

8.19.5 Error Handling

No error handling for this case.

8.20 <Functional Requirement #20> - Payroll Approval

8.20.1 Introduction

This Functional Requirement allows the payroll processor to approve the payroll

8.20.2 Inputs

Payroll processor shall be registered with COS using employee ID.

8.20.3 Processing

Payroll processor deduces the payment made by the user while placing the order by verifying the Patron's payment details and approving it.

8.20.4 Outputs

COS shall display an acknowledgement intimating 'Payment approved' and shall record the transaction details in a separate database for future reference.

8.20.5 Error Handling

No error handling for this case.

8.21 <Functional Requirement #21> - View Feedback

8.21.1 Introduction

This Functional Requirement allows the corporate manager to view various Feedback made by the user for the orders placed through Cafeteria application

8.21.2 Inputs

Corporate Manager shall be registered with COS using employee ID.

8.21.3 Processing

COS shall displays various Feedback made by the user for the orders placed through Cafeteria application. Corporate manager records various positive feedbacks and negative feedbacks for improving the Cafeteria restaurant's growth.

8.21.4 Outputs

COS shall display the various Feedbacks made by the user

8.21.5 Error Handling

No error handling for this case.

8.22 <Functional Requirement #22> - Accept Order

8.22.1 Introduction

This Functional Requirement allows the meal preparer to accept the incoming meal order from the patron

8.22.2 Inputs

Meal preparer shall be registered with COS using employee ID.

8.22.3 Processing

Patron places a new meal order, which reaches at meal preparer's inbox to accept the order. It is noted that, the order will be processed post the acceptance from meal preparer's end.

8.22.4 Outputs

COS shall display an acknowledgement intimating, 'Order accepted'.

8.22.5 Error Handling

No error handling for this case.

8.23 <Functional Requirement #23> - Accept delivery

8.23.1 Introduction

This Functional Requirement allows the meal deliverer to accept the incoming meal delivery from the meal preparer

8.23.2 Inputs

Meal deliverer shall be registered with COS using employee ID.

8.23.3 Processing

Meal preparer places new meal delivery request post completing the meal order. The delivery request reaches to the meal deliverer's inbox. It is noted that, the meal delivery will be processed post the acceptance from meal deliverer's end.

8.23.4 Outputs

COS shall display an acknowledgement intimating, 'Meal delivery accepted'.

COS also displays the delivery instructions to the meal deliverer for collecting the meal from meal preparer and delivering the same to the patron.

8.23.5 Error Handling

Cancel meal delivery – COS provides an option to the meal deliverer to cancel or denying the meal delivery, if the meal deliverer is distant from the restaurant's location.

8.24 <Functional Requirement #24> - Pickup meal for the delivery

8.24.1 Introduction

This Functional Requirement allows the meal deliverer to pick up the meal at the restaurant's location.

8.24.2 Inputs

Meal deliverer shall be registered with COS using employee ID.

Meal deliverer accepted the meal delivery.

8.24.3 Processing

Meal deliverer picks up the meal from the restaurant for delivering the same to the Patron. Meal deliverer follows the instructions provided by COS for picking up and delivering the meal.

8.24.4 Outputs

COS shall display an acknowledgement intimating, 'Meal has been picked up at the restaurant'. COS also displays the delivery instructions to the meal deliverer for delivering the meal to the patron.

8.24.5 Error Handling

No error handling for this case.

8.25 <Functional Requirement #25> - Display most rated meal at Patron's home screen

8.25.1 Introduction

This Functional Requirement requests that, users shall view the most rated meal at their home screen

8.25.2 Inputs

Patrons shall be registered with COS

8.25.3 Processing

Various recipes that has been rated as favorite by majority of Patrons, shall be collected and displayed at the home screen when Patron opens the COS

8.25.4 Outputs

COS displays most rated recipes at Patron's home screen

8.25.5 Error Handling

No error handling for this case.

8.26 <Functional Requirement #26> - Search Bar for Patron's input for surfing the meal

8.26.1 Introduction

This Functional Requirement allows the user to enter his desired keyword for ordering a meal

8.26.2 Inputs

Patrons shall be registered with COS

8.26.3 Processing

Patron shall be given preference to enter the restaurant's keyword or the meal's keyword for ordering the meal. COS shall fetch the results for the keywords entered by the Patron.

8.26.4 Outputs

COS displays the keyword results provided by the Patron at the search bar

8.26.5 Error Handling

COS displays an error message if there are no results for the Patron's input.

8.27 <Functional Requirement #27> - User interface shall be easy to use

8.27.1 Introduction

This Functional Requirement request the COS developers to design the user interface which makes all the user classes to feel them easy to use the application.

8.27.2 Inputs

Various User classes (Patron, meal deliverer, meal preparer, menu manager, corporate manager, payroll processor) shall have registered with COS

8.27.3 Processing

User interface shall be catchy for various users and shall make them feel easy to use it.

8.27.4 Outputs

User interface

8.27.5 Error Handling

No error handling for the case.

8.28 <Functional Requirement #28> - Change/Update user details

8.28.1 Introduction

This Functional Requirement provides a platform for the users to maintain their personal data with COS.

8.28.2 Inputs

Patrons shall have registered with COS

8.28.3 Processing

Patrons shall change their address details, card details, default cuisine preference, change meal subscriptions under 'Account Settings' label.

8.28.4 Outputs

Account settings label, including address details, card details, default cuisine preference, change meal subscriptions options.

8.28.5 Error Handling

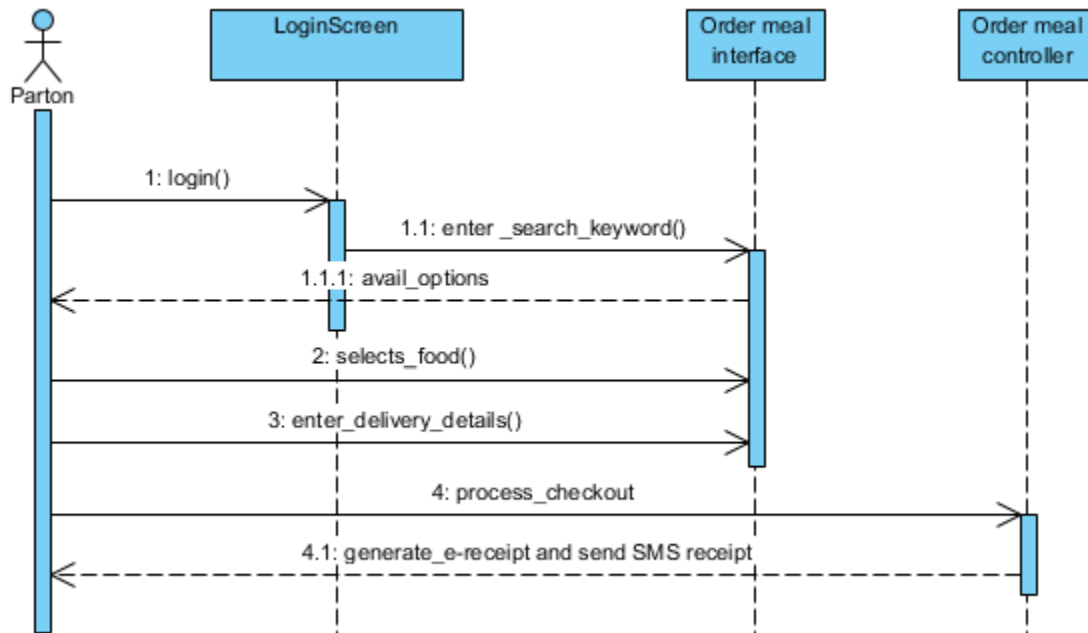
COS shall display an error message if the Patron's address is invalid.

COS shall display an error message if the Patron's card details are invalid.

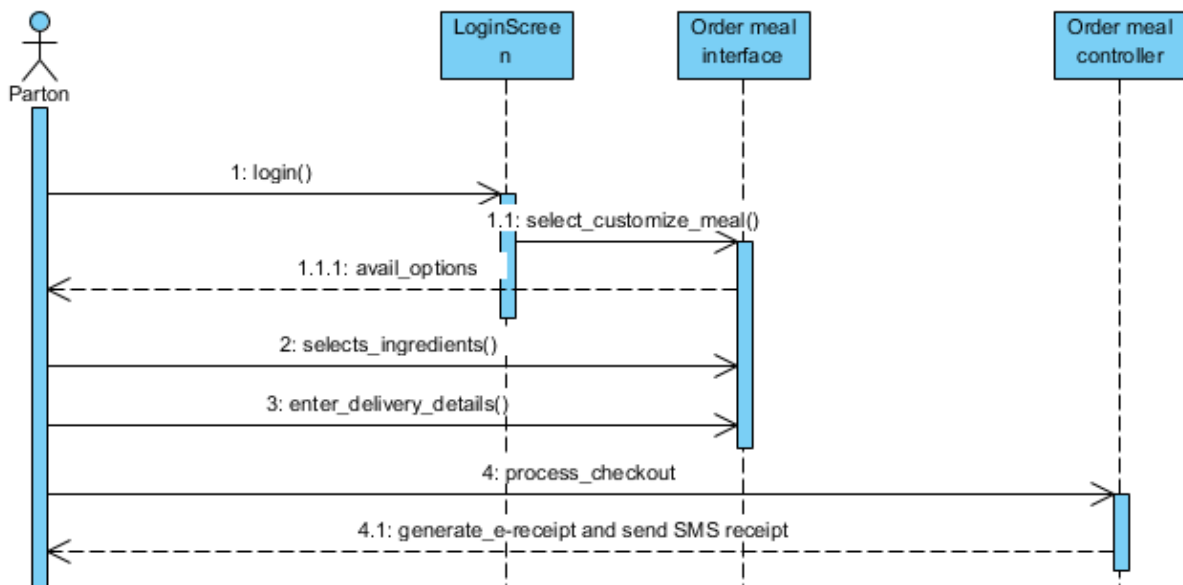
9. SEQUENCE DIAGRAM

The various sequence diagrams for sequence diagram are as follows,

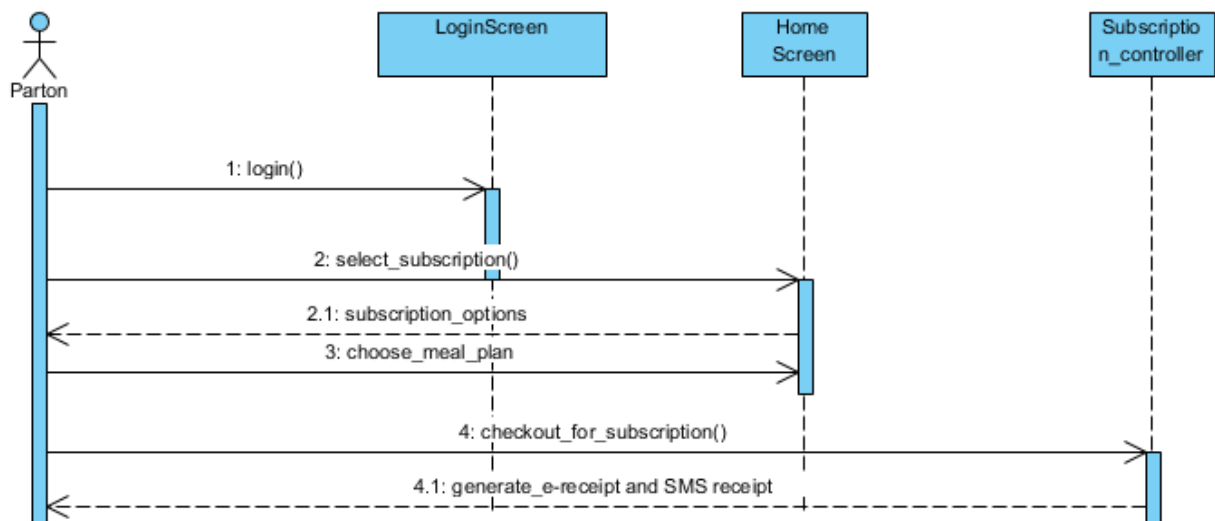
1. ORDER A MEAL



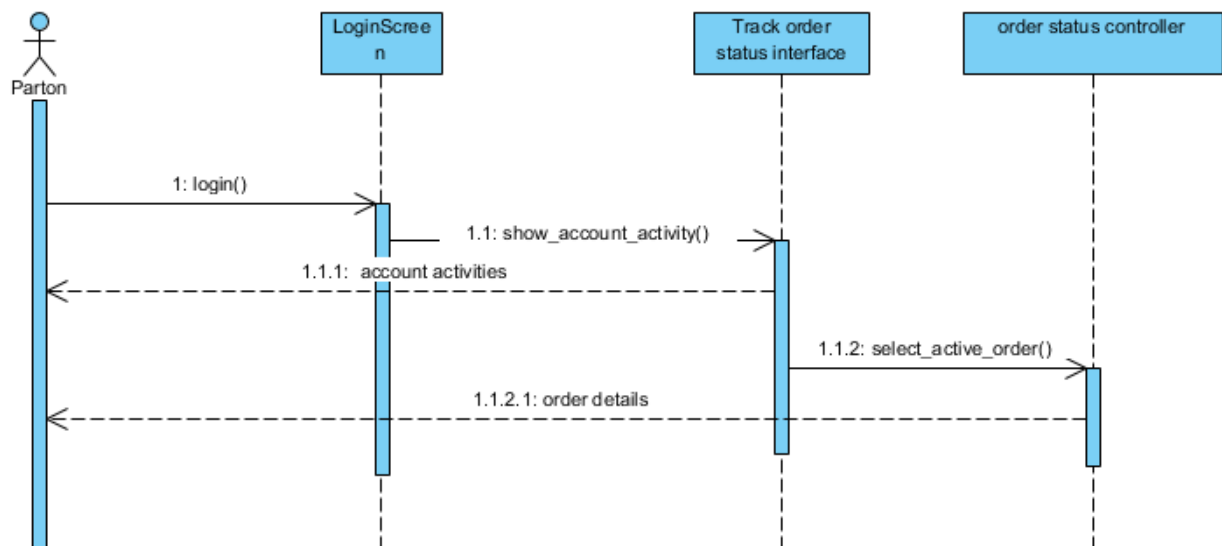
2. ORDER A CUSTOMIZED MEAL



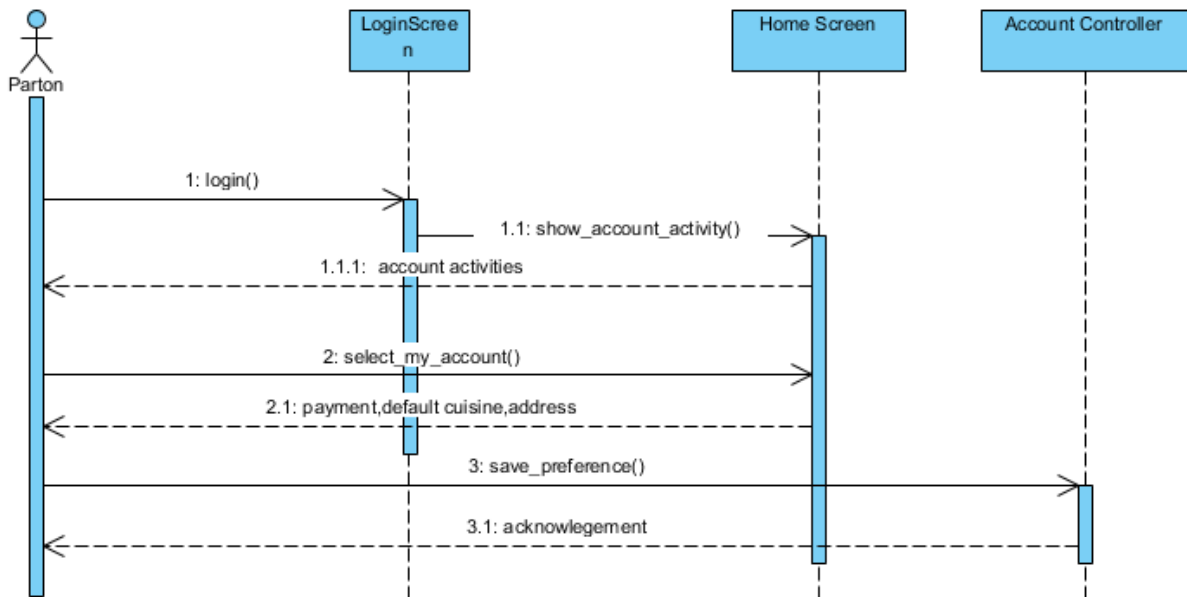
3. SUBSCRIBE A MEAL PLAN



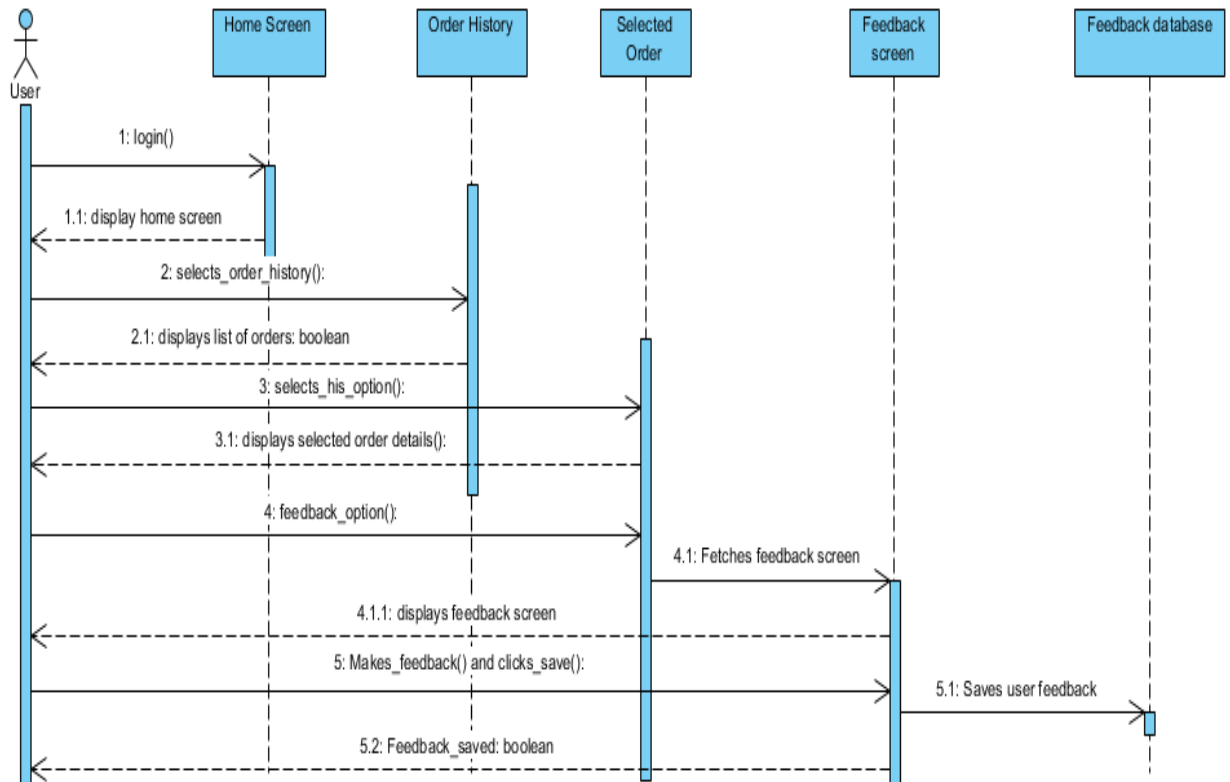
4. TRACK ORDER STATUS



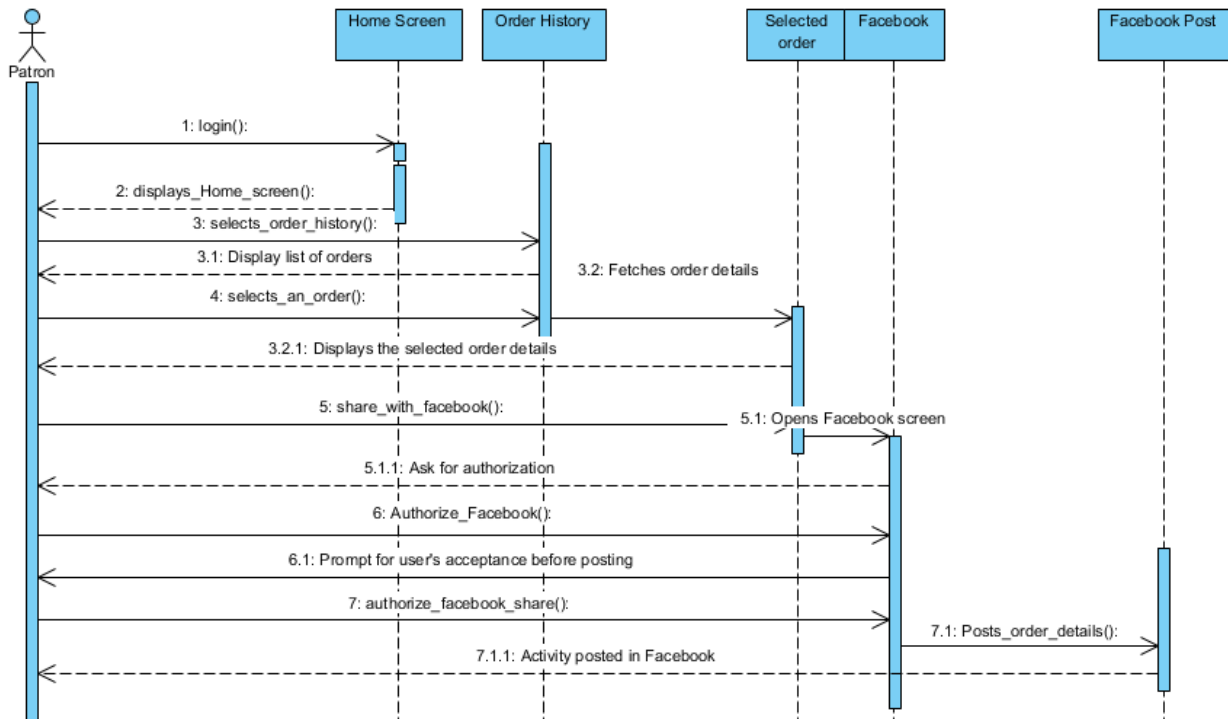
5. MANAGE MY ACCOUNT



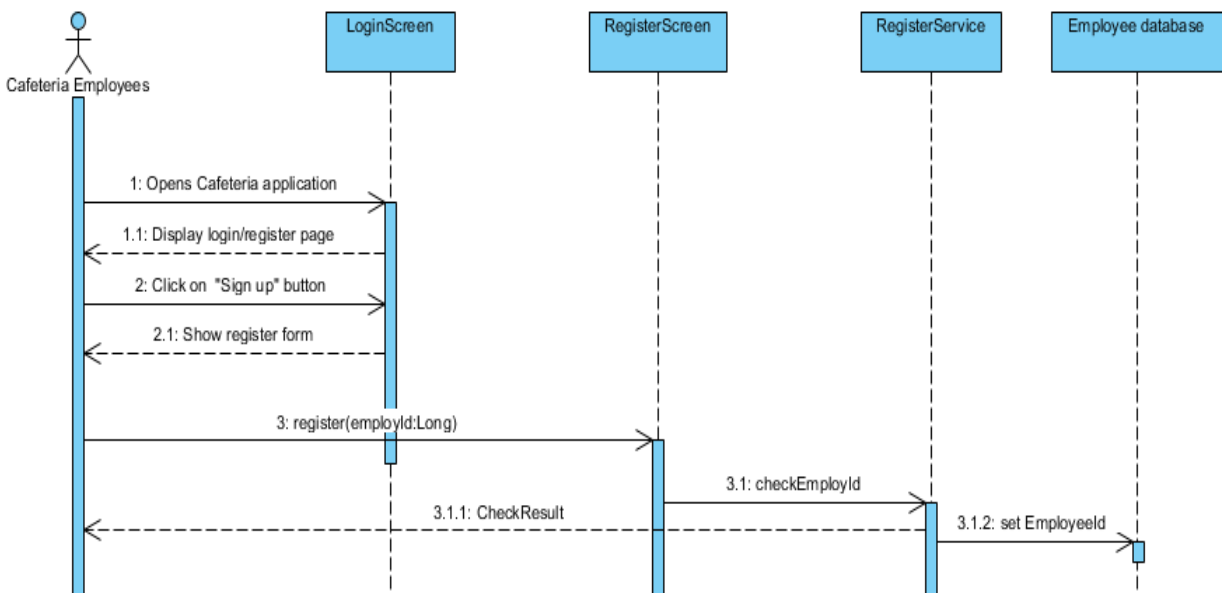
6. MAKE A FEEDBACK



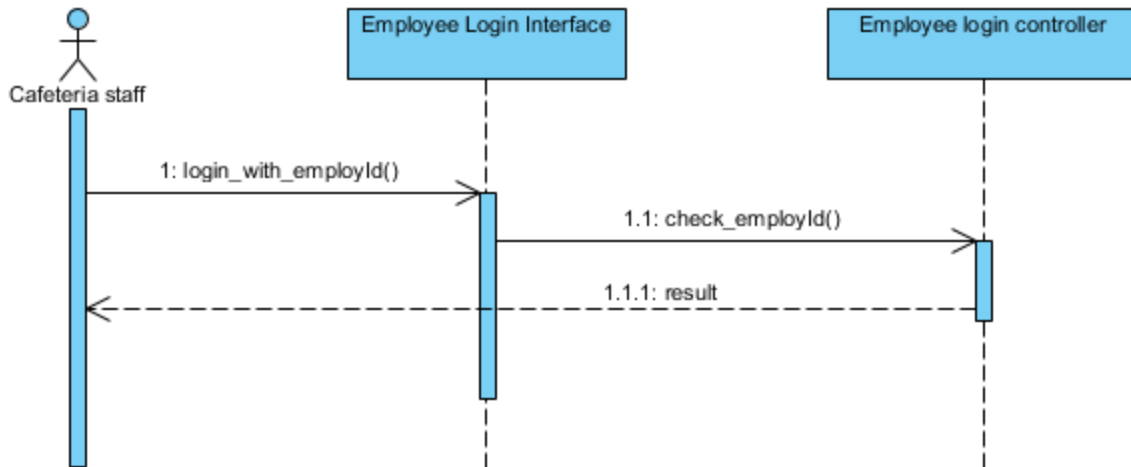
7. SHARE A CAFETERIA ACTIVITY WITH FACEBOOK



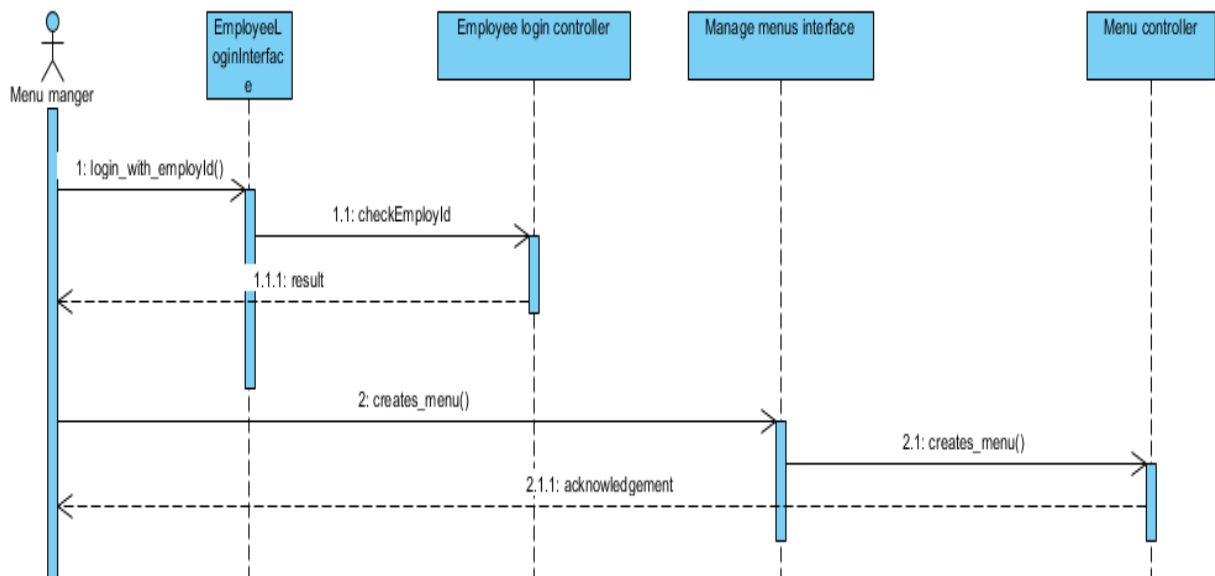
8. REGISTER WITH COS USING EMPLOYEE ID



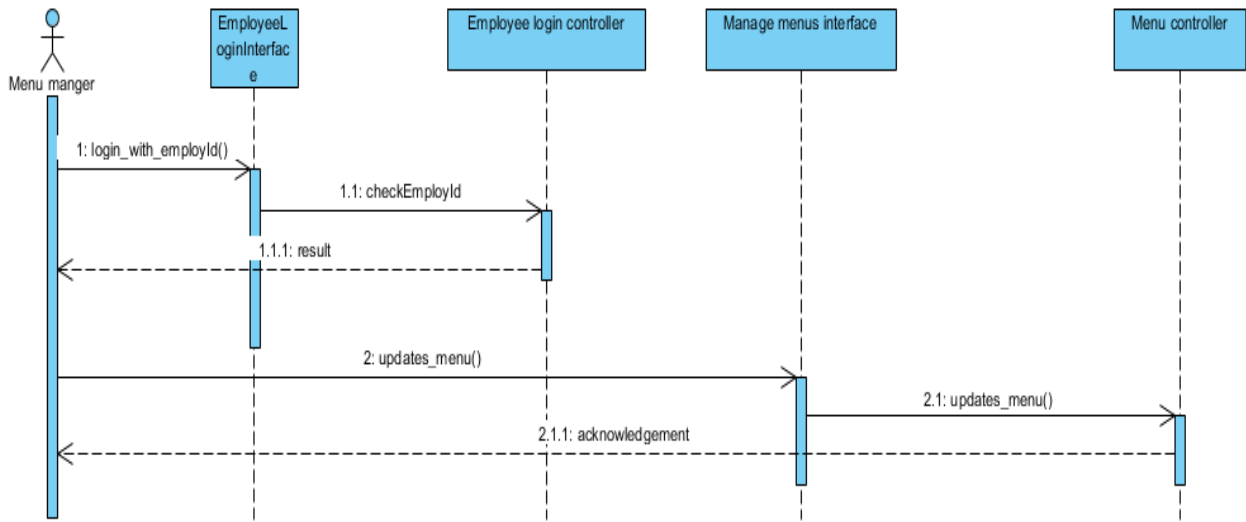
9. LOGIN WITH CAFETERIA USING EMPLOYEE ID



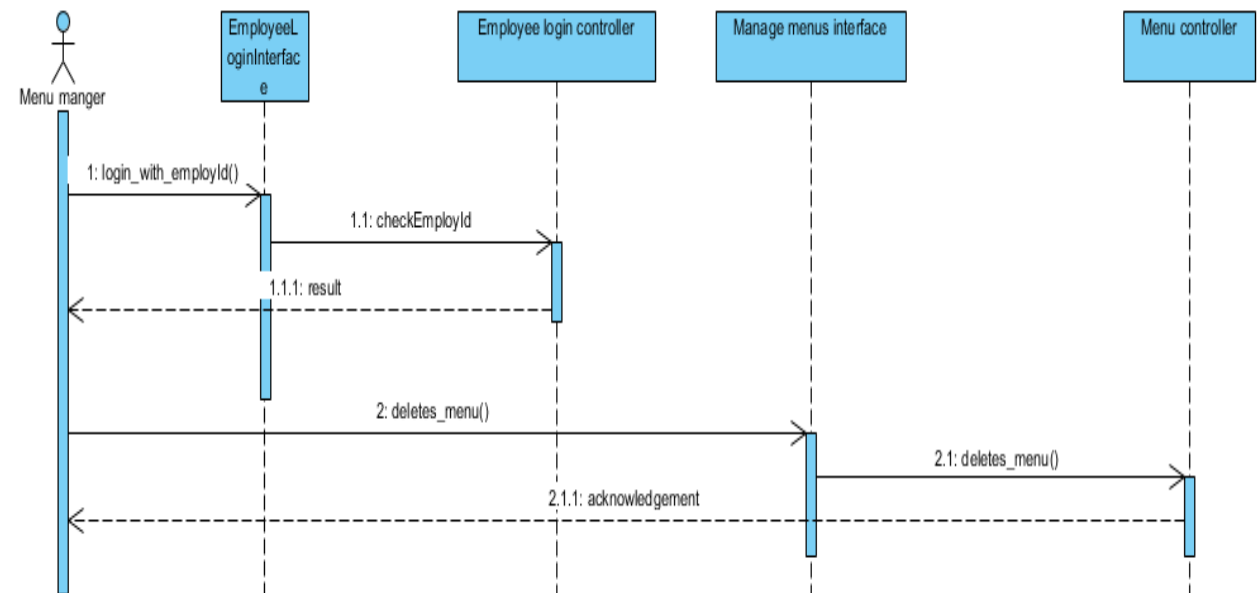
10. CREATE A MENU



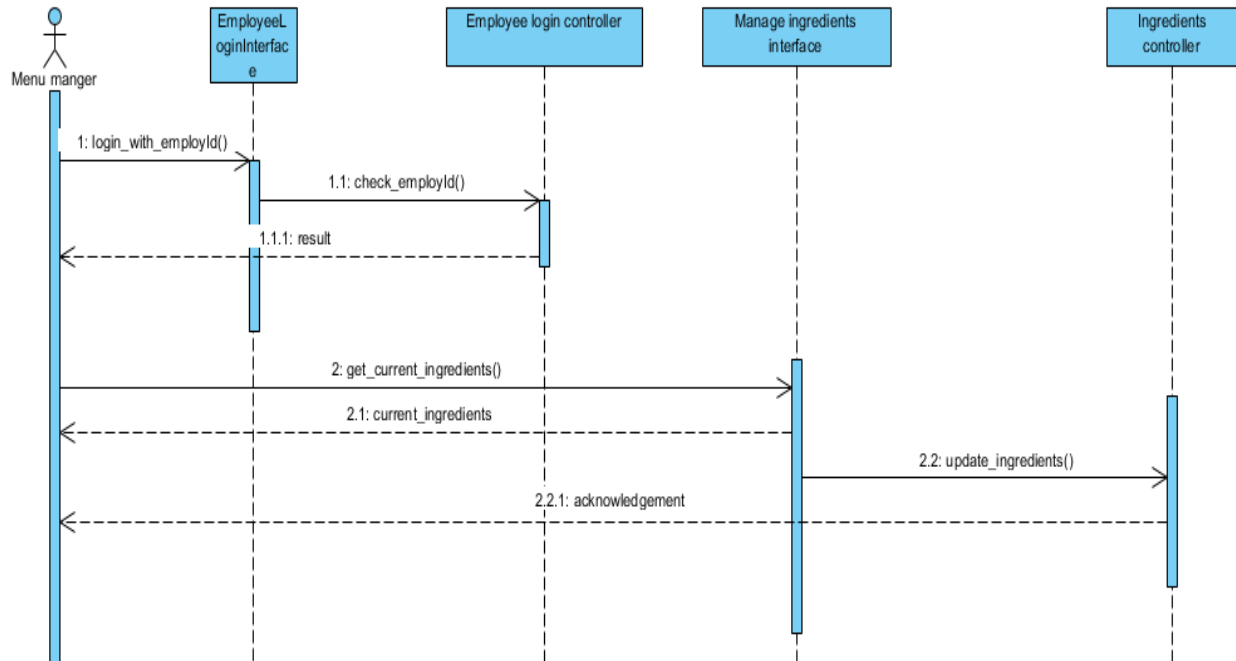
11. UPDATE A MENU



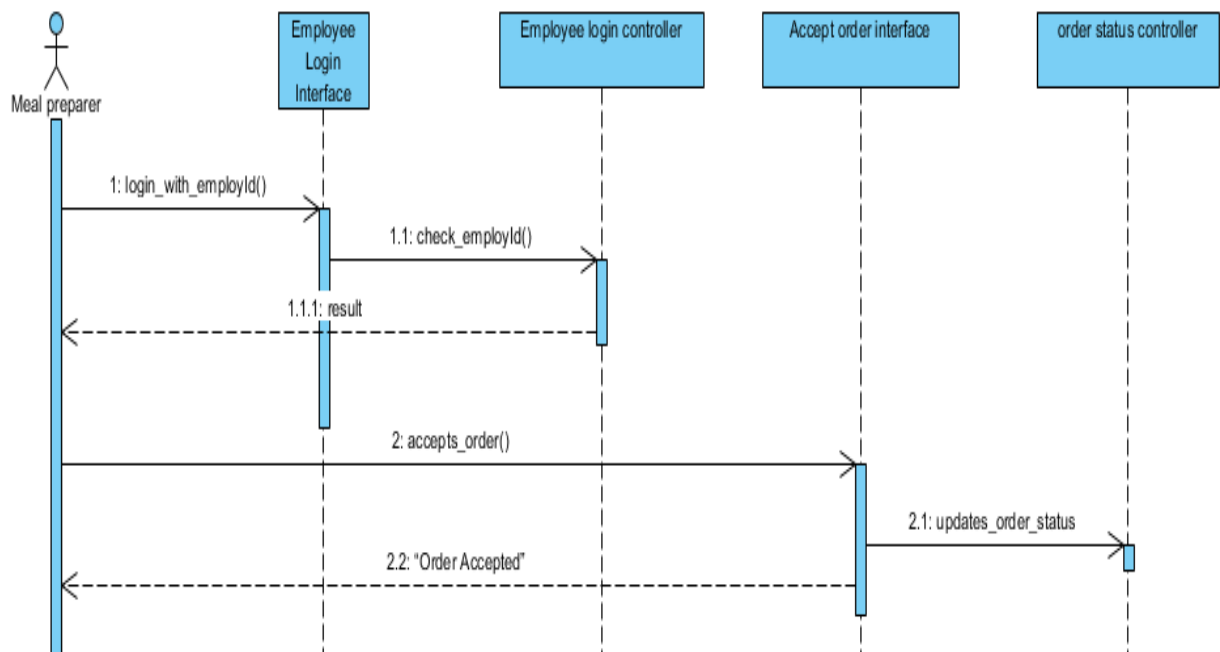
12. DELETE A MENU



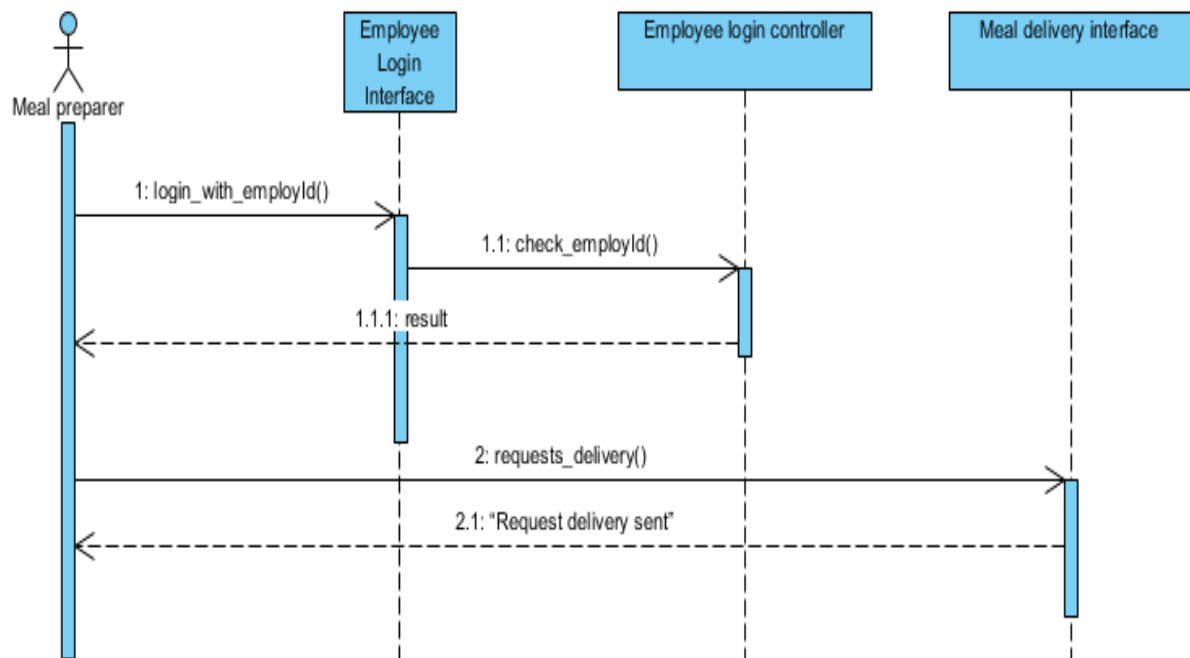
13. RECIPES AND INGREDIENTS LIST MANAGEMENT



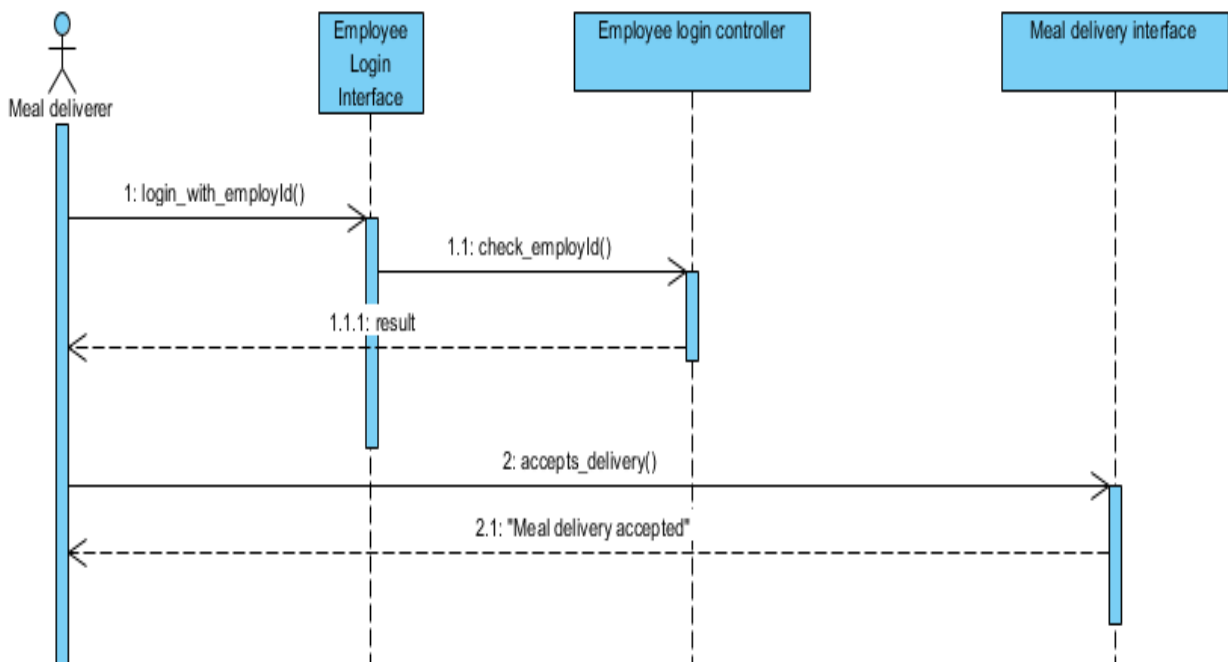
14. ACCEPT AN ORDER



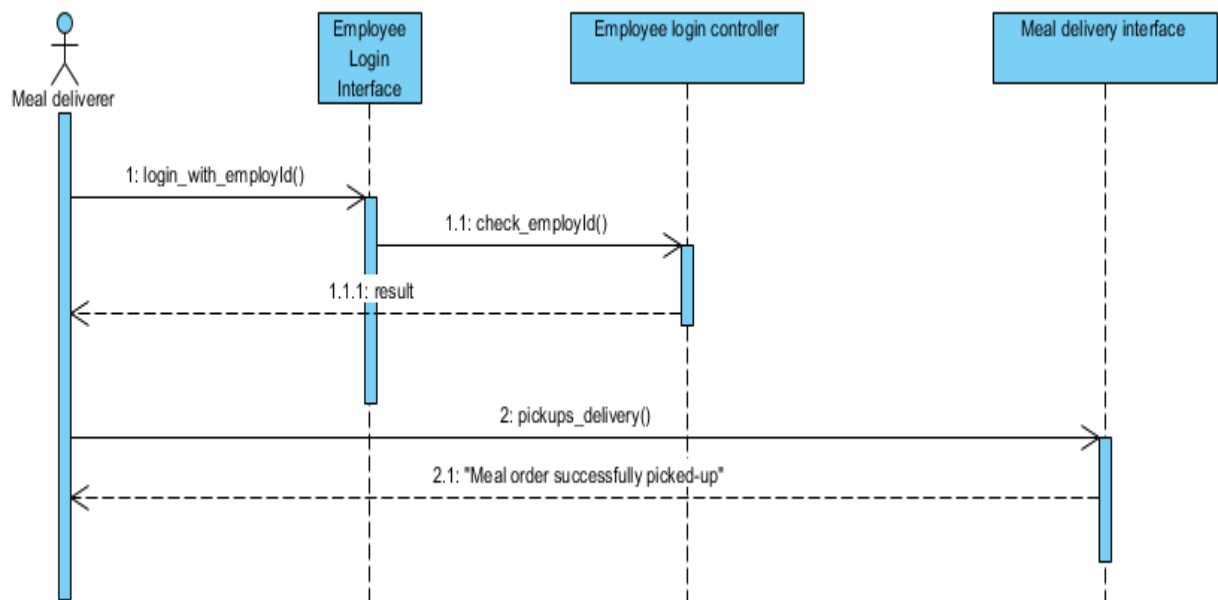
15. REQUEST DELIVERY



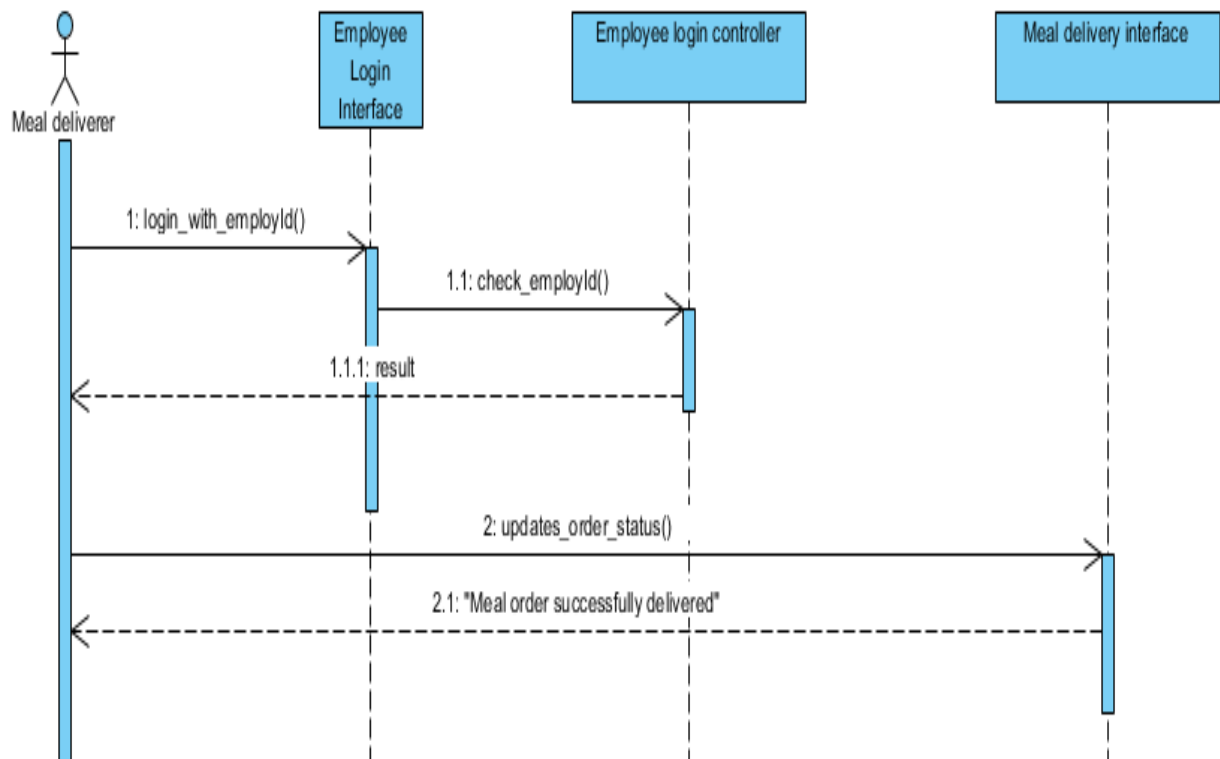
16. ACCEPT DELIVERY



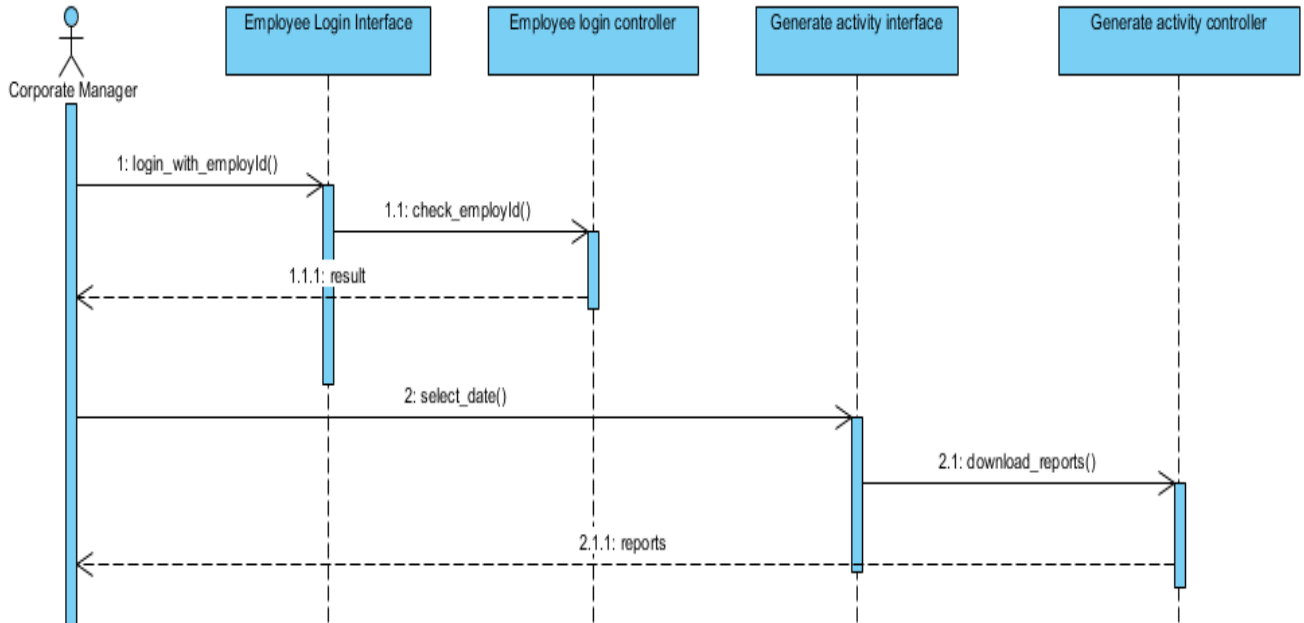
17. MEAL PICKUP



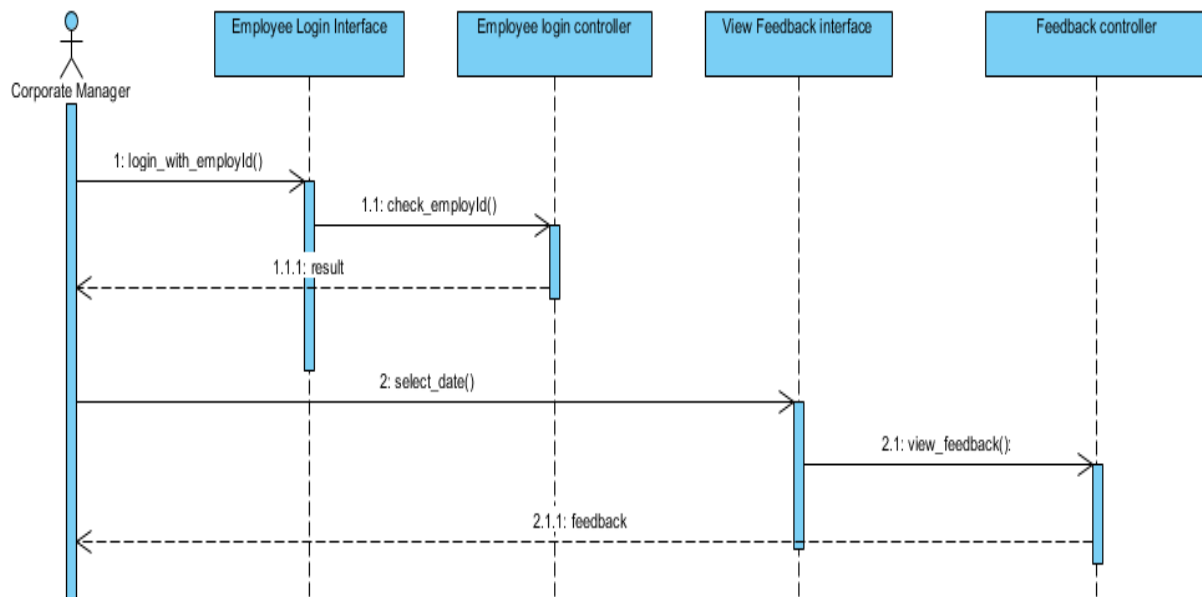
18. DELIVER AN ORDER



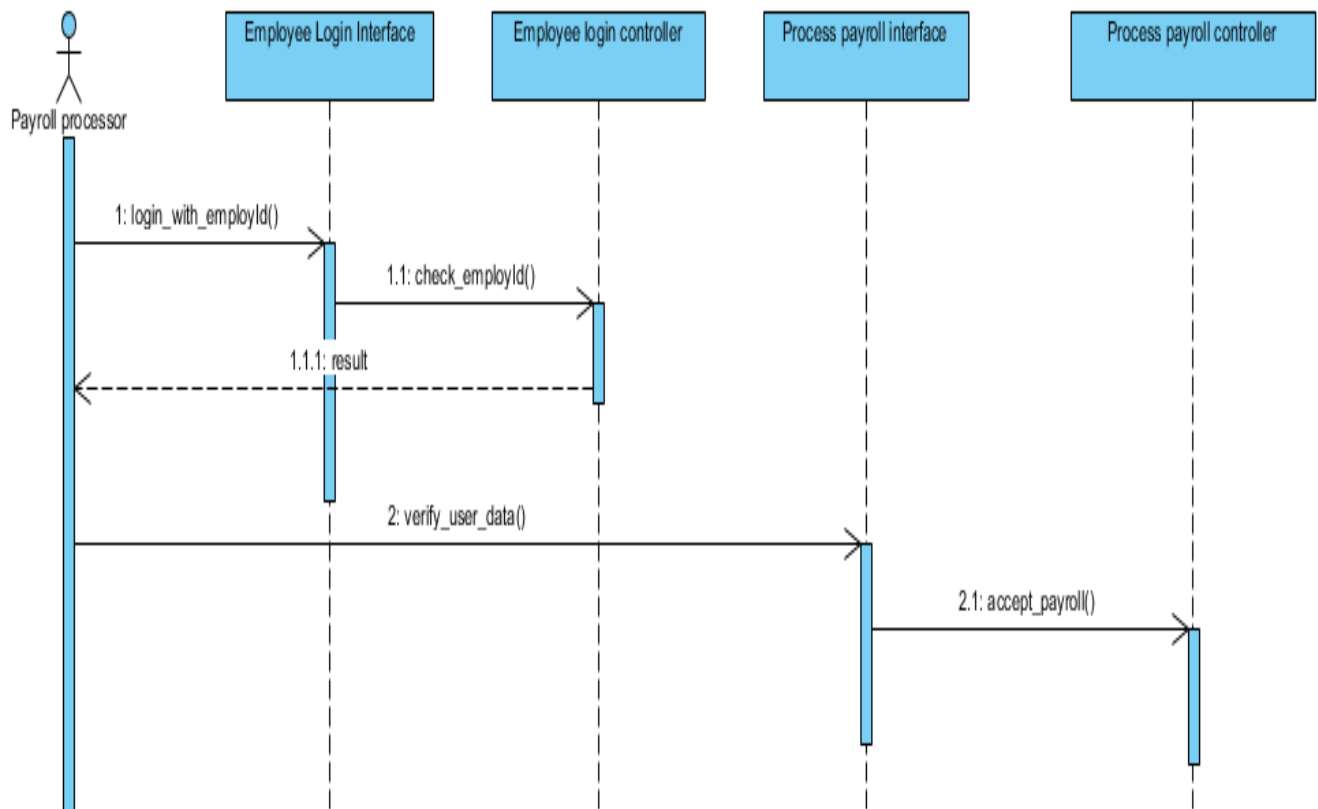
19. GENERATE CAFETERIA ACTIVITY REPORTS



20. VIEW FEEDBACK



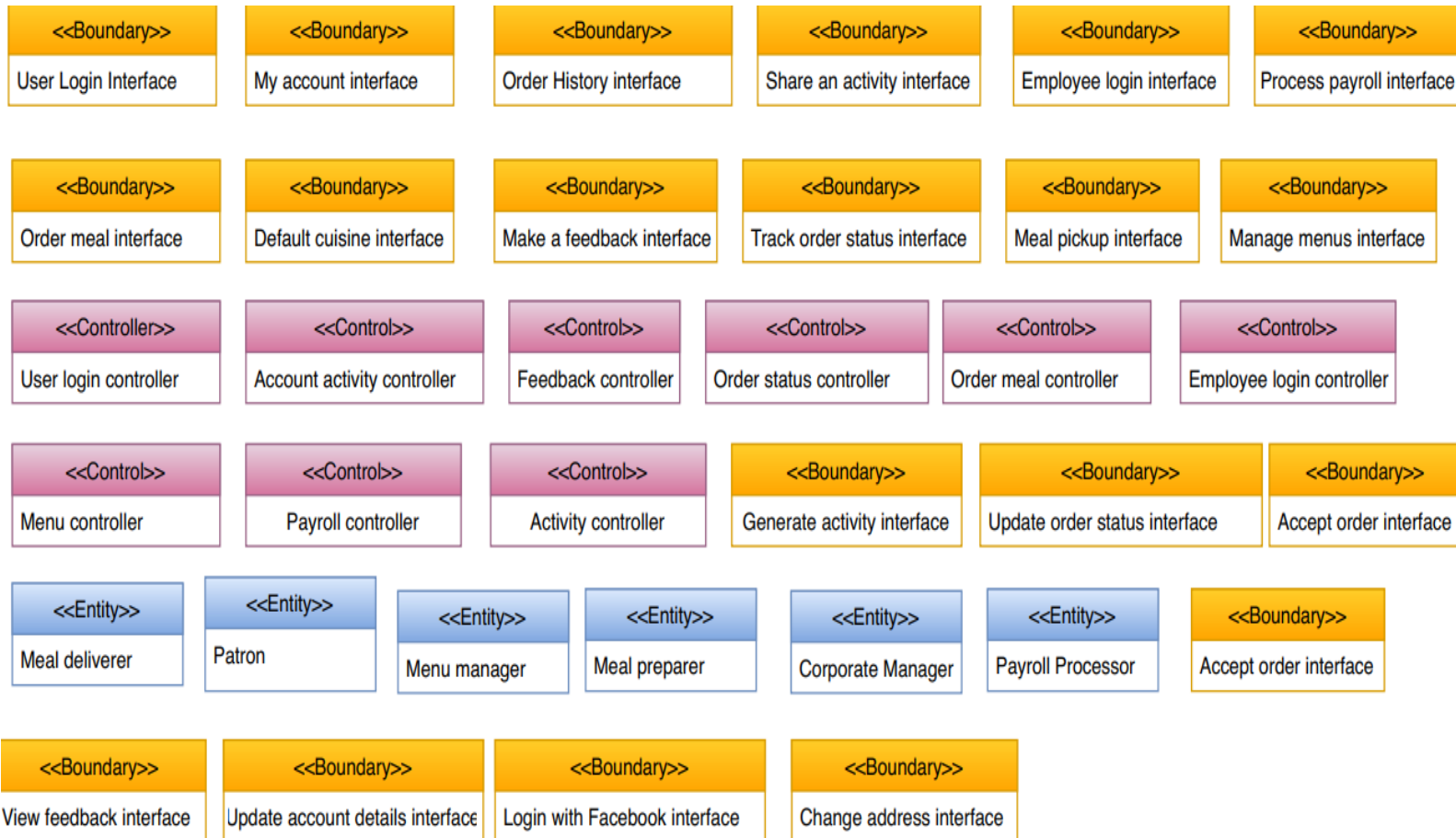
21. DEDUCE PAYROLL






10. CLASS DIAGRAMS

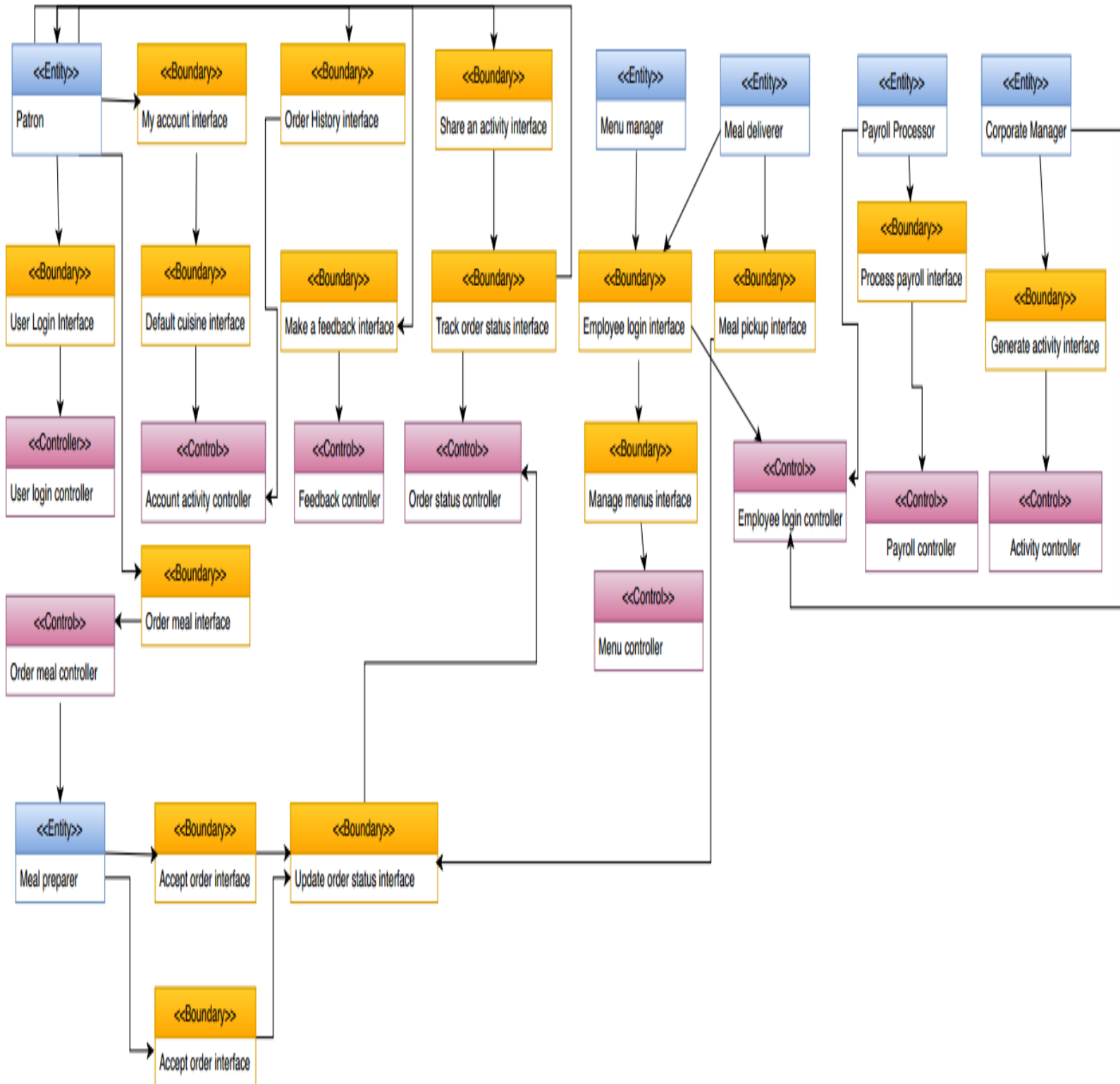
The various class diagrams for Cafeteria Ordering System are as follows,




10.1 INITIAL CLASS DIAGRAM



<i>KEY</i>	
<i>Color code</i>	<i>Class Name</i>
	<i>BOUNDARY CLASS</i>
	<i>CONTROL CLASS</i>
	<i>ENTITY CLASS</i>

10.2 MODIFIED CLASS DIAGRAM



<i>KEY</i>	
<i>Color code</i>	<i>Class Name</i>
	<i>BOUNDARY CLASS</i>
	<i>CONTROL CLASS</i>
	<i>ENTITY CLASS</i>

10.3 DETAILED CLASS DIAGRAM

