SOFTWARE DESIGN DOCUMENT

for

< Cascade - Contactless Cafeteria Ordering System >

Version 1.0 approved.

Prepared by

19Z210 - Deepti Ravi Kumar

28.09.2021

TABLE OF CONTENTS

1 INTRODUCTION

1.1 Purpose

The purpose of this document is to describe the software design requirements for release 1.0 of the Cafeteria Ordering System (COS). This document is intended to be used by the members of the project team that will implement and verify the correct functioning of the system. Unless otherwise noted, all requirements specified here are high priority and committed for release 1.0.

The document contains the functional behavior and non-functional requirements of the system project. The document also contains the guidelines for system engineers and programmers to start working and accomplish the project on a given time frame.

1.2 Scope

The software product is named Cafeteria Ordering System. Keeping track of the cafeteria menu, attendance, and consumption is a difficult task. Manual and paper-based procedures are cumbersome and prone to errors, resulting in inaccuracy and waste of time and resources. Therefore, an automated canteen management system is essential.

This project attempts to build a canteen management system that analyzes food consumption and waste by item, offers customers a convenient mobile ordering system, and includes an AI-based attendance system. Meals for breakfast, lunch, and dinner, as well as special days and events, may all be planned. This project enables menu item monitoring, quick transactions, and the avoidance of accounting errors. Customers can also provide feedback using the feedback function, which allows the canteen manager to analyze and make necessary adjustments to the system.

1.3 Overview

The Cafeteria Ordering System assists the canteen manager in managing the canteen more efficiently and effectively by automating meal ordering, billing, and inventory control. The AI-based attendance and food collection tracking can help to ensure effective food distribution, as well as penalty and refund decisions.

The application provides the users with a clear and descriptive guideline for an appealing user experience. A wide range of hover texts are available to help navigate the user through the application. The application will include an FAQ or similar page that describes the application's terms, its functionality, and any other information deemed relevant to users. In case of further queries, the user shall contact the support team.

1.4 Reference Material

i. *IEEE Std 1016-1998*

ii. *[IEEE] The applicable IEEE standards are published in "IEEE standards Collection" 2001 edition.*

1.5 Definitions and Acronyms

This part involves acronyms and abbreviations used in the SDD.

- *SRS: Software Requirements Specification*
- *AI: Artificial Intelligence*
- *DESC: Description*
- *ERD: Entity-relationship diagrams*
- *Graphic Depicting - small icons that correspond to the weather.*
- *CRUD - In computer programming, create, read, update, and delete (CRUD) are the four basic functions of persistent storage. Alternate words are sometimes used when defining the four basic functions of CRUD, such as retrieve instead of reading, modify instead of update, or destroy instead of deleting.*

2 SYSTEM OVERVIEWS

The main objective of this project is to manage canteen details like food, staff, and orders. This project is totally for administrative purposes, only administrators have the authority to access it.

The purpose of this project is to build an application and reduce manual work for managing activities. It tracks the details about orders, student details, and staff.

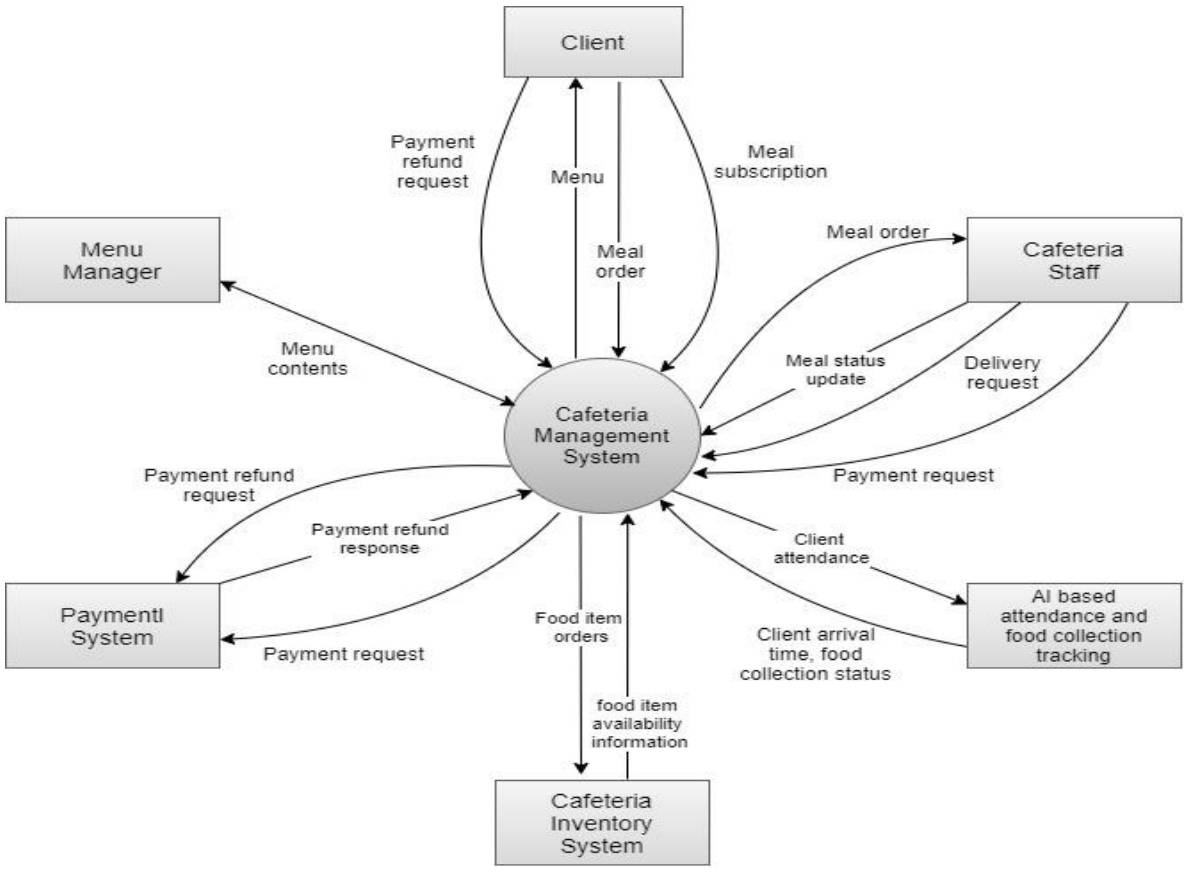
Once the order placing is done on the webpage and all the information is retrieved through the database then the task of an application has to accomplish.

3 SYSTEM ARCHITECTURE

3.1 Architectural Design

When the client visits a canteen application, they navigate through the ordering module, which displays a menu card with all available options and quantity data based on the customer's preferences. After selecting an item, the consumer must examine the details before checking out the form system.The application provides instant confirmation of the selected item for order, demonstrating the system's precision in completing processes in the shortest time possible. This system displays the overall load on the canteens once the entire process has been completed and item orders are automated. Upon receiving an order for an application, and obtaining all the necessary information from the database, the work of an application is completed.

Following the completion of orders, delivery information is entered in a very straightforward manner. This allows canteen workers to rapidly assess orders as they are placed for goods chosen by clients in the shortest amount of time possible.



3.2 Decomposition Description

3.2.1 DFD Level 0

The basic phase of application architecture includes the basic functioning of this phase:

i. ***Canteen management***
This phase includes the process of management functioning of canteen that how we can manage the items which are needed to be purchased from the canteen.

ii. ***Food Item Management***
This phase includes the process of food management that how we can manage the food items according to quantity and requirements of customers.

iii. ***Sales Management***
This phase shows the total number of sales of items that how much number of items are sold to customers.

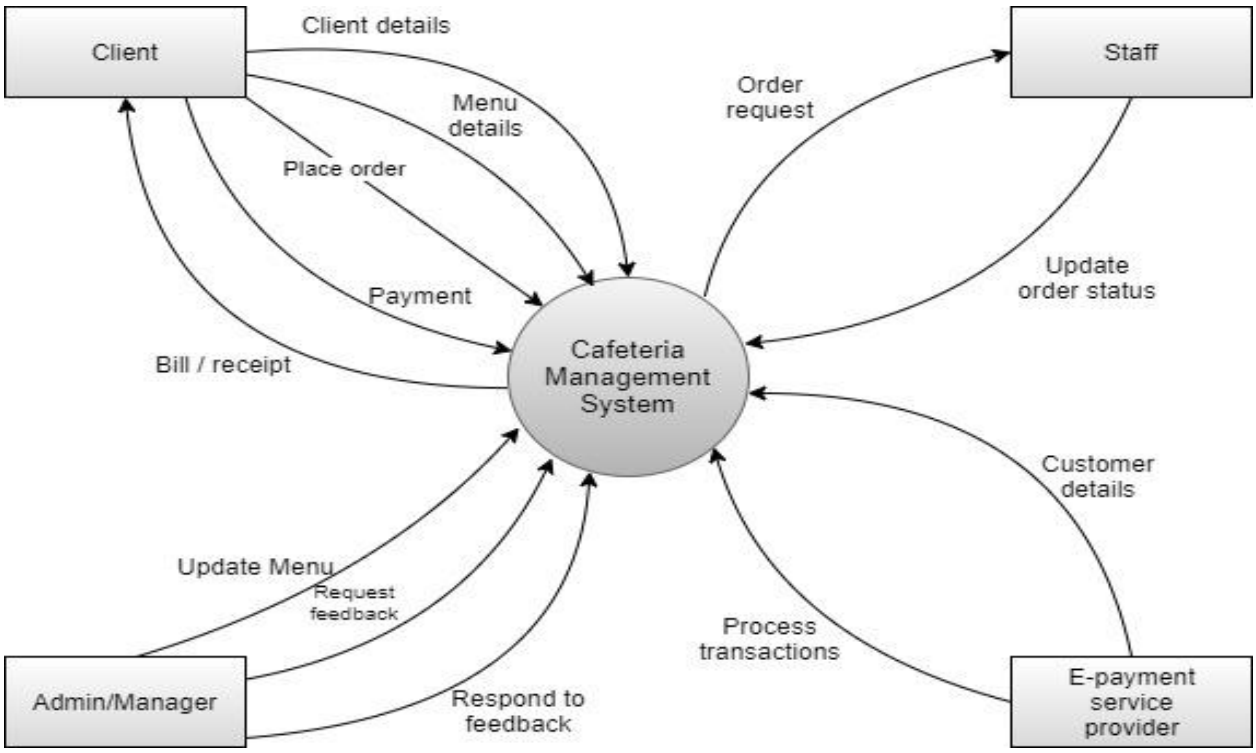
iv. ***Customer Management***
This phase shows the customer details and number of customers who have purchased the items with their total count.

v. ***Login Management***
This phase shows the user counts how many users are logged in to the system.

vi. ***System User Management***
This phase shows the interaction between system and user, how the user accesses this system according to his/her requirement.

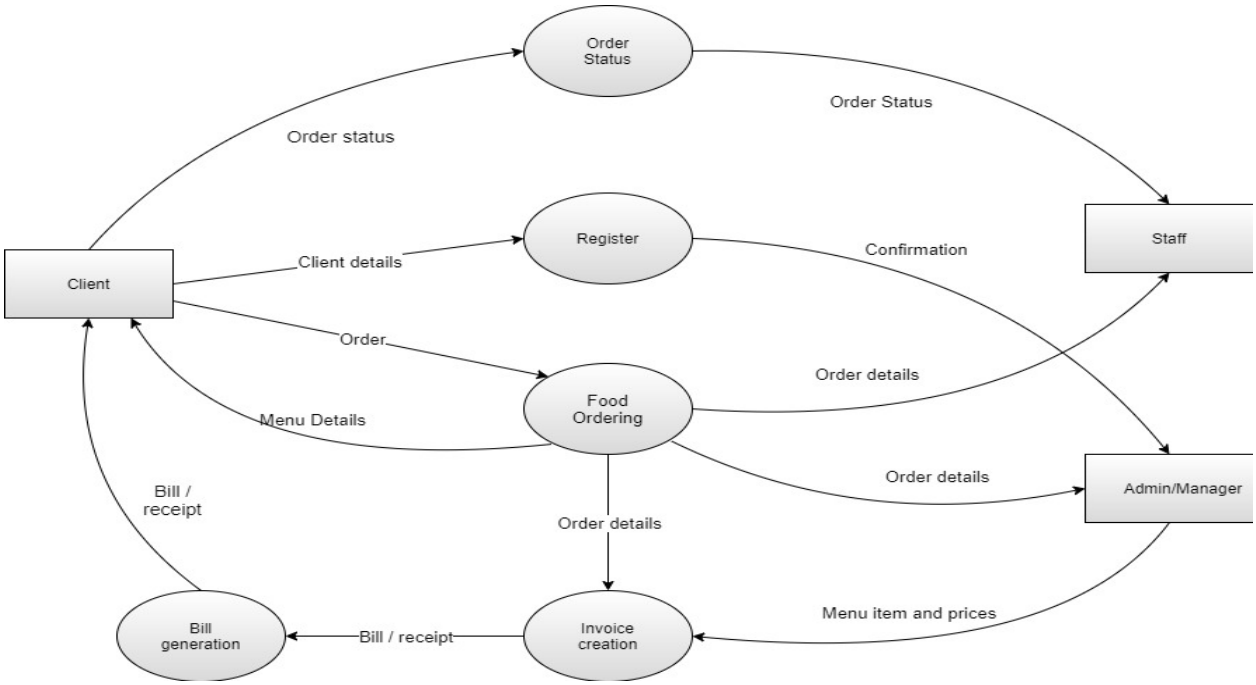
In this level of DFD, all of the functions will be performed in the order given below,

- *Food order via application*
- *Confirm order*
- *Online Payment*
- *Serve food*
- *Available goods*
- *Required goods*
- *Customer information*
- *Customer review*



3.2.2 DFD Level 1

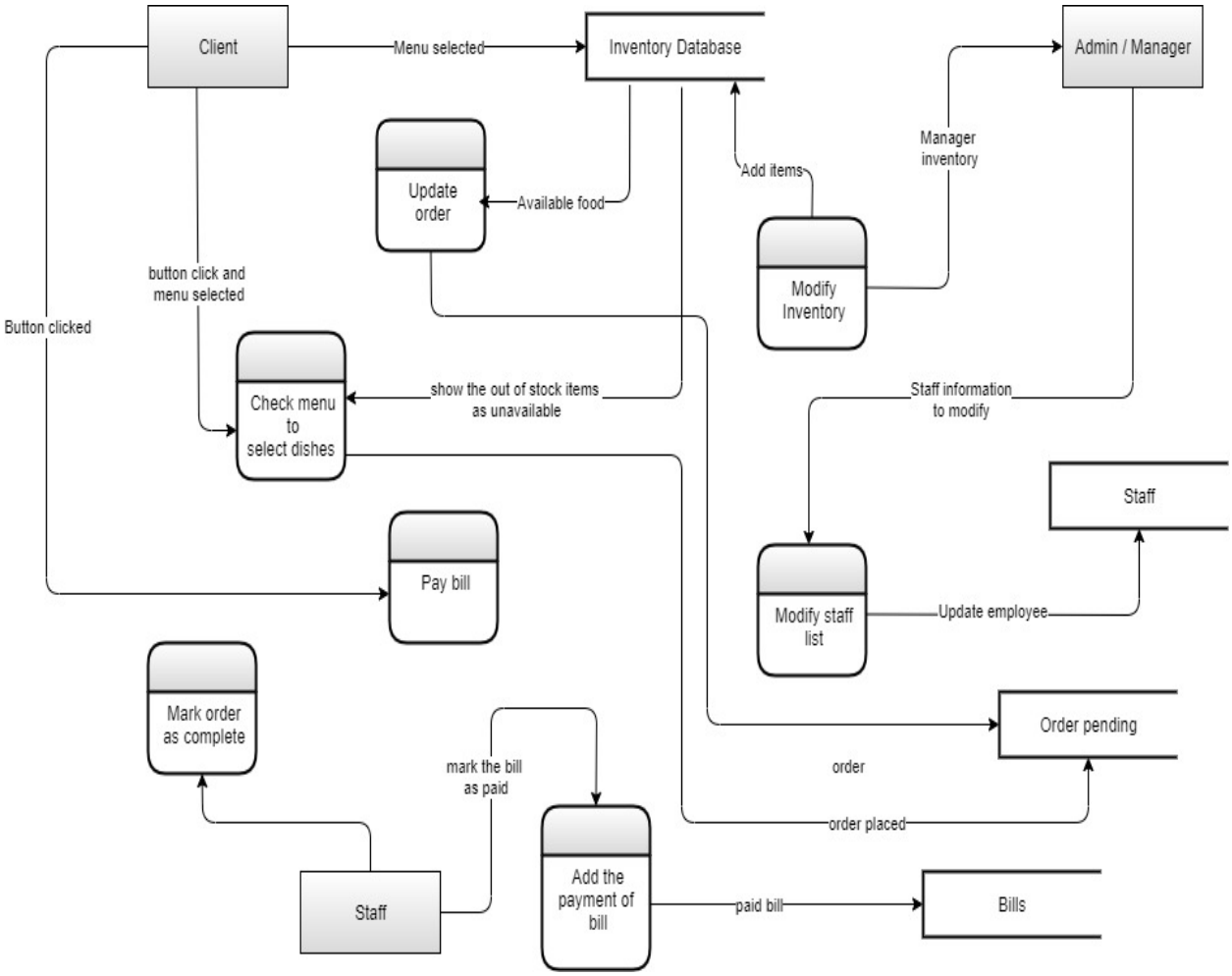
Furthermore, reports are generated for each and every process. Here, a detailed flow of data over various action elements is indicated.



3.2.3 DFD LEVEL 2

In this level of DFD illustration of the whole system in that login in that system is done then it will check all the details of the system after that accessibility check of system is done. After that, there are several modules that are managed such as:

i. *Manage system admins*
ii. *Manage roles of users*
iii. *Manage user permission*
iv. *Manage report*
v. *Manage food meal details*
vi. *Manage employee details*
vii. *Manage item details*
viii. *Manage sales details*
ix. *Manage customer details*



3.3 Design Rationale

The system will be portable and easily accessible (usable) and is highly reliable, efficient and the system is implemented by following all the legislative standards. It also ensures safety by following the organizational ethics. It is also interoperable.

3.3.1 Balancing of loads

System is developed for admin purposes, loads need to manage in a proper manner so the login activity on the server can be done in a limited time period, there must be an appropriate time period for the server accessibility.

3.3.2 Easy to Access

Records are needed to be stored in the database without any loss in a proper manner so that all the details are managed properly according to information which is stored in the database, and it can be accessed easily during the requirement of it.

3.3.3 User Friendly System

A system needs to be user friendly so it can be easily accessed by users without any issues, some systems are very complex that users are barely able to access them.

3.3.4 Efficiency and reliability

Maintaining all details of records in the database on the server which will be easy to access according to the user requirement without any maintenance. Maintenance cost must be minimum and efficient as compare with storing all the customer data on the excel spreadsheet or in physical book records.

3.4.5 Easy Maintenance

Maintenance is fully dependent on the design work. If the system is designed in an easy way, not in a complex way then the maintenance will be eas

4 DATA DESIGN

4.1 Order management module

- *Requirements based according to users:*
  According to this module, customers are able to order their food as well as cancel it according to their preference.
- *Requirements based according to the system:*

4.1.1 Add order module

In this module information about the order is displayed, staff members need to enter those details into the system. Then the validation process got implemented and checking accuracy is done.
If there are some error occurrences, then a prompt message will occur that entered details are invalid.

4.1.2 Sequencing of orders

Newly orders are segmented in a queue, during the processing of orders system will dequeue according to first come first serve basis, the chef will cook food according to the order which served in front of him.

4.1.3 Deletion of orders

Staff can easily delete orders by just clicking on the delete button. Only those orders got removed which are canceled by customers.

4.1.4 Total calculation of orders

In total calculations, all the order details are retrieved from Database. Then each will be multiplied from the total quantity of orders then this process will continue till it covers all the orders. Then the total amount of orders will be displayed on the screen.

4.1.5 Add payment module

In the add payment module canteen inputs the payment information which will be validated in the system.
If the information is incorrect then an error message will occur and ask the user to re-input, the details. In addition, there is a dialog box that shows a summary of the actions.

4.2 Maintenance module based on customers

In this module details of customers are maintained such as an address, contact details, etc. Which can be used as a reference for contacting customers.

4.2.1 Add customer module

In this module details of customers are entered into the systems. A validation check will occur before the storing of data in the database, if there are some errors then a prompt message will during the validation process.

4.2.2 Edit customer module

In this module modification in customers can be done. We can add and delete details of customers according to the validation process, rectification of mistakes can be done easily through making changes in fields.

4.2.3 Delete customer module

The desired record will be searched first, by just clicking on the delete button we can remove the record which is useless after deletion of records message will display that the record is deleted.

4.2.4 Order maintenance module

In this module, we are going to keep track of orders changing and cancellation of orders will be also done.

5 COMPONENT DESIGN

The Program Features for Admin:

- *Add new Customer*
- *View all Items*
- *View sold Items*
- *Add New Item*
- *Edit Item*
- *Delete Item*

The program has a database connection at the beginning of the program.

The application has the following fundamental functions:

a. ***Function addNewItem ()***
- *Take name and quantity input from user*
- *If insertion query work successfully, then success message will be shown if not then error message will be shown in this Canteen Management System program.*

b. ***Function addNewSale()***
- *Pass two parameters, name and quantity of the item stored in canteen*
- *After adding new item to sold table, an update query will run to update the quantity of item in items table in the application*

c. ***Function allItems()***
- *In this function, all items are fetched from the items table*
- *If query failed, then it will show the error message.*

d. ***Function soldItems ()***
- *In this function, all sold item names from the sold table are fetched*
- *If query failed, then it will show the error message*

e. ***Function searchById ()***
- *In this function, item with one specific id is searched by passing id as parameter*
- *At last, it returns found variable*

f. ***Function searchByName ()***
- *In this function, item with one specific name is searched by passing name as parameter*
- *At last, it returns found bool variable.*

g. ***Function checkQuantity ()***
- *When user select item from the whole list, then it checks if the quantity is available or not with the specific name passed name and quantity*
- *previousQuantity will be a global variable. It is used to update items after user purchase any item*
- *It will return the found variable*

h. ***Function editItem()***
- *In this function, it first shows all items available in the store*
- *Then prompt for user input to enter an ID to edit*
- *Then it checks if the id is valid or not by searchById() function*
- *If id is found, then it will select the specific id from the items table and show in the console*
- *After then it asks the user to input the new item name and the quantity*
- *After then an update query will be created with new name and quantity*

i. Function deleteItem ()

- *At first, it shows all the items from the item table*
- *Then ask the user to enter a valid to Id which he/she want to delete*
- *Then it checks if the is valid or not*
- *If valid then run a delete query to delete the item*

5 HUMAN INTERFACE DESIGN

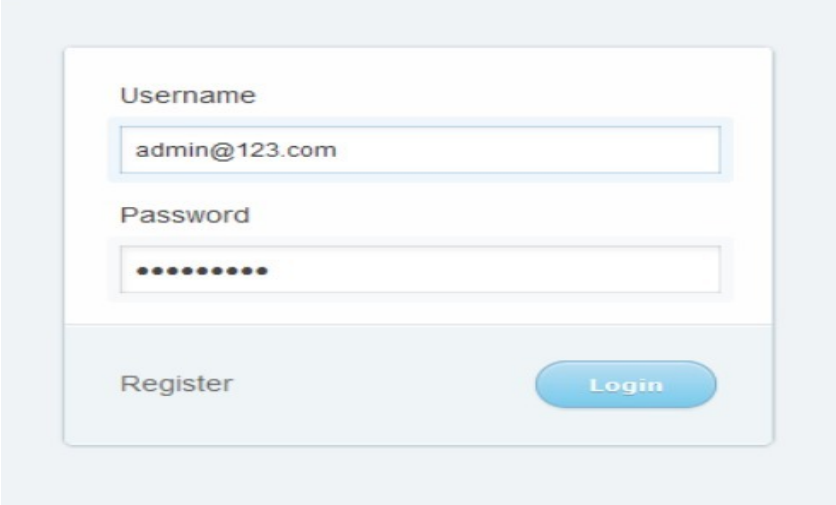
6.1 Overview of User Interface

The customer interface will contain three screens: Place order, Edit/Cancel order, feedback. Each dish will have an image, its price per serving. After confirming the order, the user will be shown a timer screen. In this screen customer will be shown "Edit Order" and "Cancel Order" buttons and a timer which shows the completion time of the order. There will also be a button to request for bill. In feedback screen, a button for "Request Bill" will be shown. Beneath this button we will display a form which will have different multiple-choice questions and a submit feedback button.

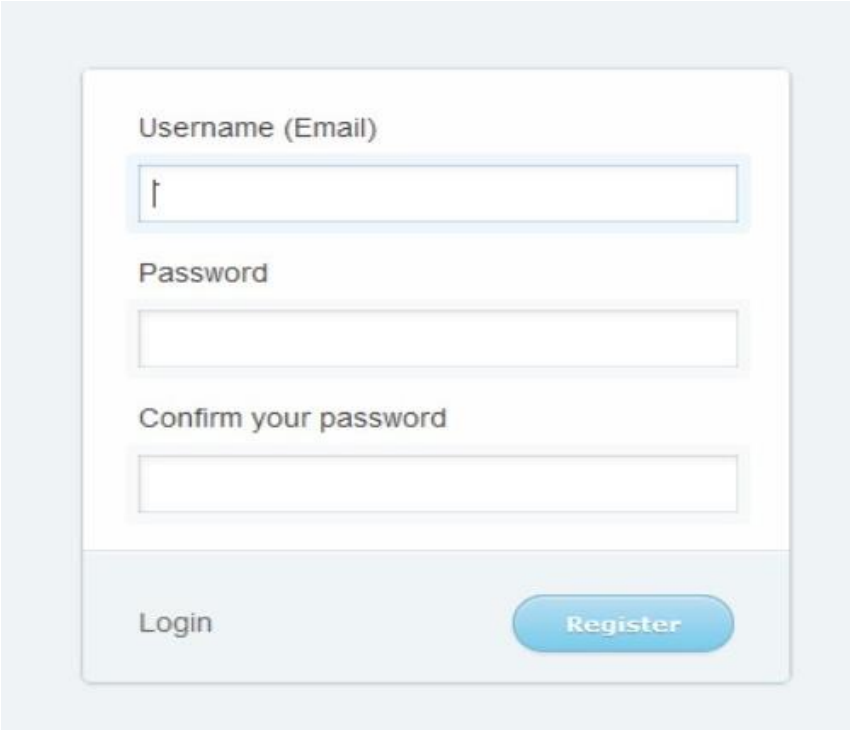
In head chef interface, system will show all the current orders in detail i.e., all the dishes of a particular order. In each order, there is a button which will be used to mark that dish cooked. Moreover, when customer wants to remove a dish from his order, system will show head chef a notification to approve the removal of the dish.

6.2 Screen Images

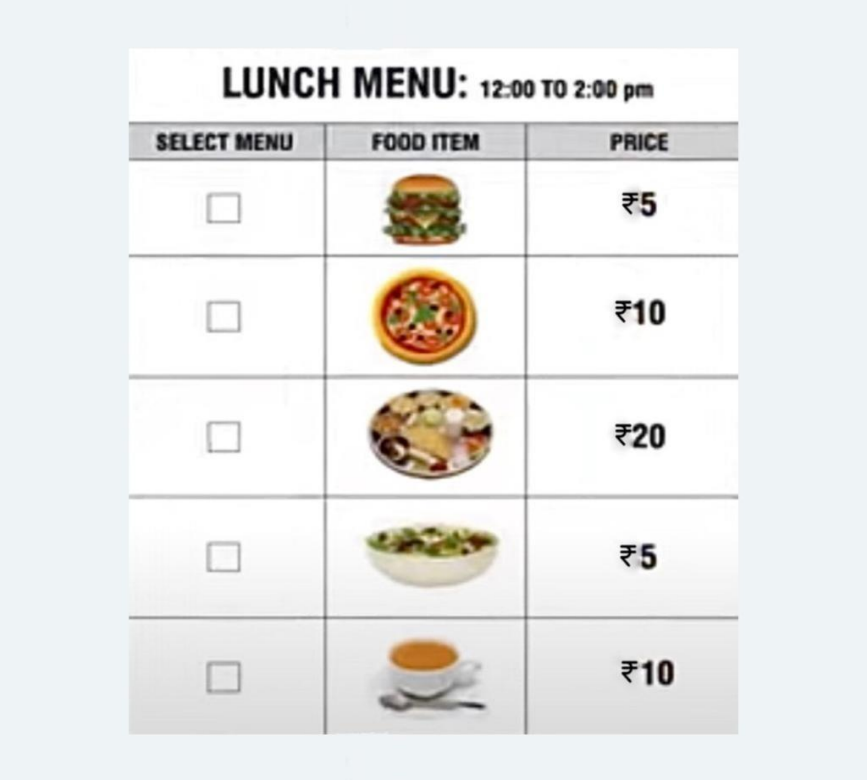
i. Login Page



ii. Registration Page



iii. Menu Page



iv. Food order tracking page

6.3 Screen Objects and Actions

The product includes 3 major action phases are as follows:

6.3.1 Administrator Phase

- *Order Incoming*
- *Printing of bills*
- *Add and Remove options are available*
- *Offers are added*
- *Updating Inventories*
- *Count of sales place in week and month*
- *Count of each individual item sold every day*
- *Count of Total earnings*

6.3.2 Staff Phase

- *Customer placed an order*
- *Total offers according to item*
- *Forecasting of Order*

6.3.3 User Phase

- *Selection of items from Menu*
- *Placing of orders*
- *Combo Selection of items*
- *Billing*

7 REQUIREMENTS MATRIX

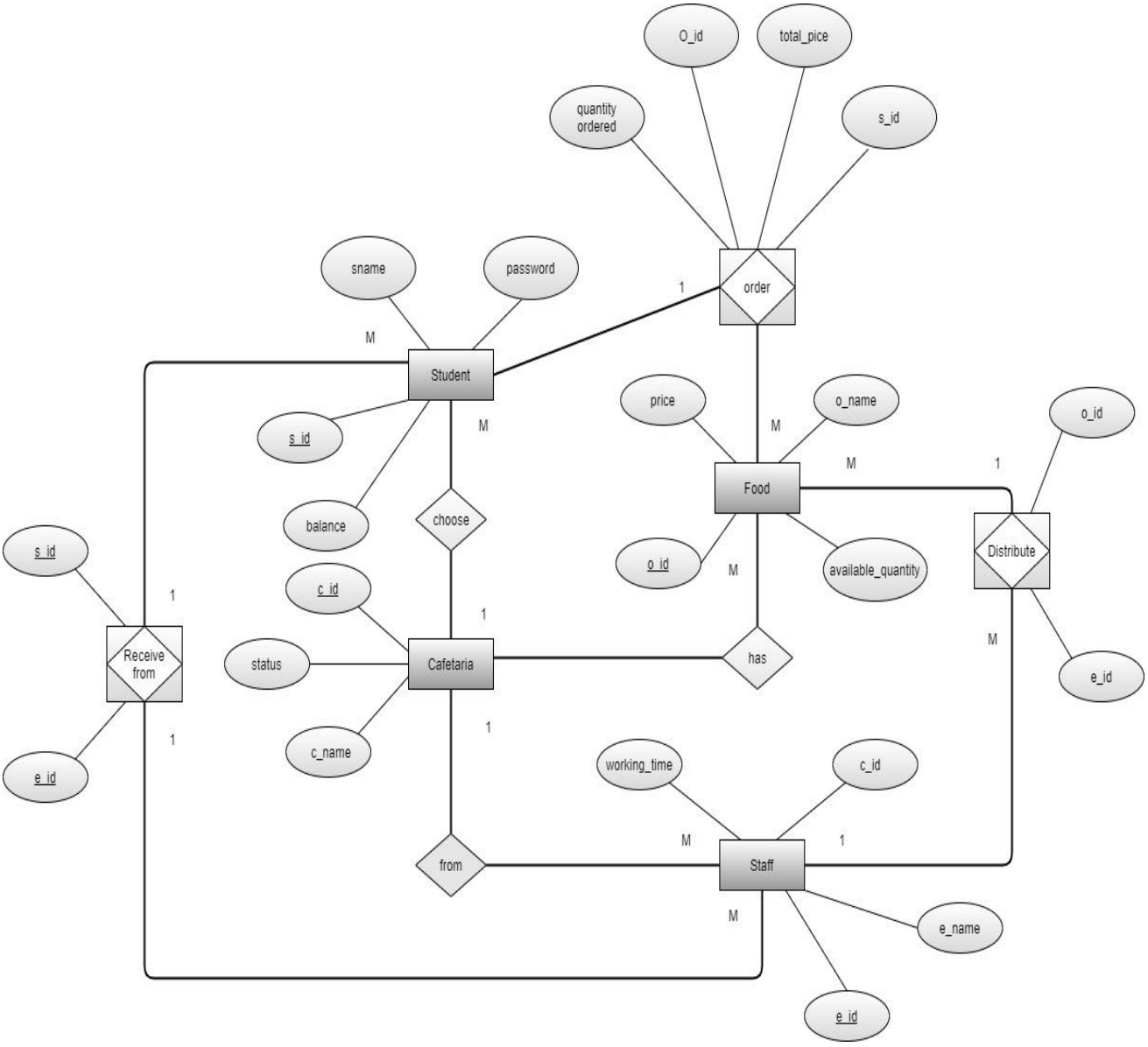
SNO.	*REQUIREMENT DESCRIPTION*	*DESIGN REFERENCE*	*MODULE NAME*
1.	***Add order*** *User will enter details into the system before it got stored in the database, a validation process will take place whether all the details are correct or not, if some details are incorrect, a dialog box will warn the user that some details are incorrect.*	*4.1.1*	*Order Management module:* • *Add Order Module*
2.	***Edit order*** *Users can easily edit order details, Modification of orders can be done in this module without any inconvenience, if modification is inappropriate then an alert message will occur.*		
3.	***Complete order*** *This module will then display the completion of orders according to the first-come and first serve process. Users can select complete status after finalization*	*4.1.2*	*Order Management module:* • *Sequencing of orders Module*

	of orders and the record got saved automatically.		
4.	***Cancel order*** *Customers can cancel orders according to their wish, if a record is not available then it can easily remove from the database and the status will also display that order is got canceled.*	*4.1.3*	*Order Management module:* • *Deletion of orders module*
5.	***Total calculation of orders*** *The system will calculate the total amount of orders based on algorithms that are implemented during this process it also includes total quantity and charges based on services.*	*4.1.4*	*Order Management module:* • *Total calculation of orders module*
6.	***Add payment*** *In this module, payment details have entered the system. Then the validation process begins, and it ensures accuracy if some details are invalid, they will be rectified through dialog, and it will grab the error and resolve it.*	*4.1.5*	*Order Management module:* • *Payment module*
7.	***Post payment*** *It will search the user payment details through the post, then the user will click on a post this payment, and a dialog will display that the user gets affirmation.*		

8 APPENDICES

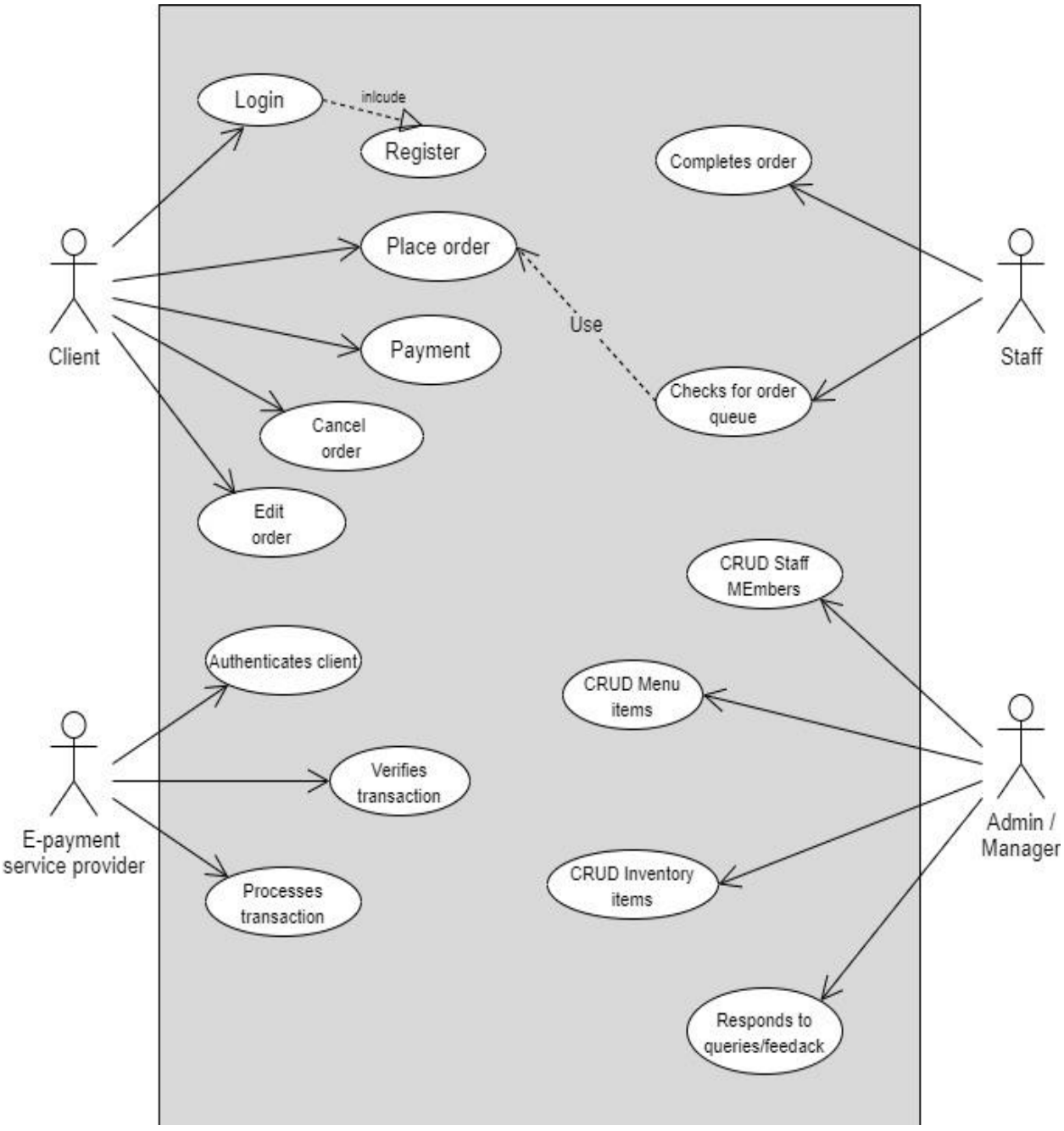
8.1 Analysis Models – ER diagram

The following is an ER diagram of the Mausam- Weather Forecasting Application:



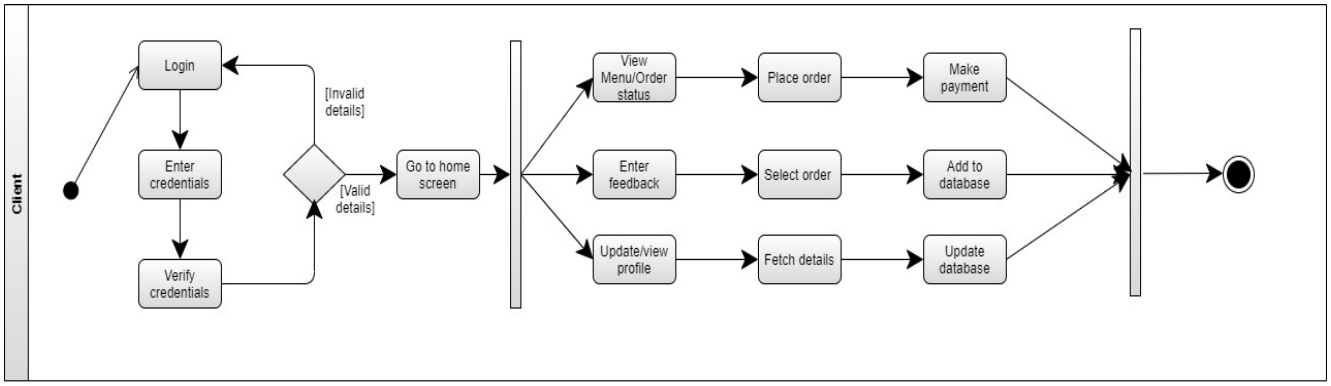
8.2 Analysis Models – Use Case model

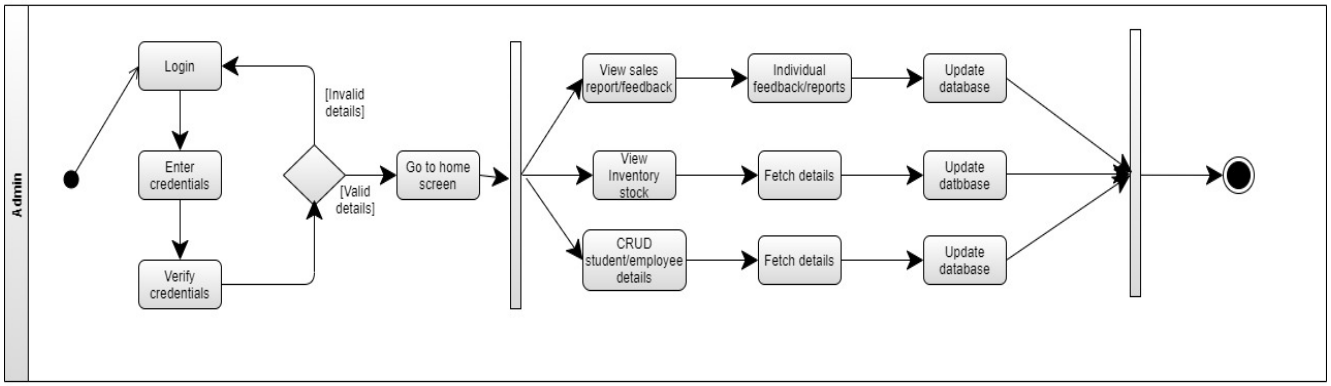
Following is the simple diagram that shows the major components of the overall system.

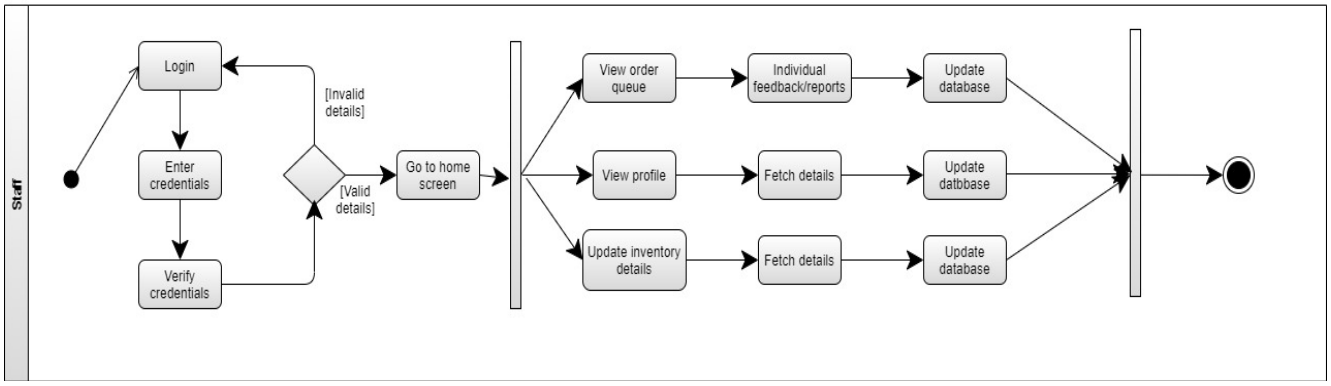


8.3 Analysis Models – Activity Diagram

Following is the activity diagram that shows the activities performed by the major components of the application..

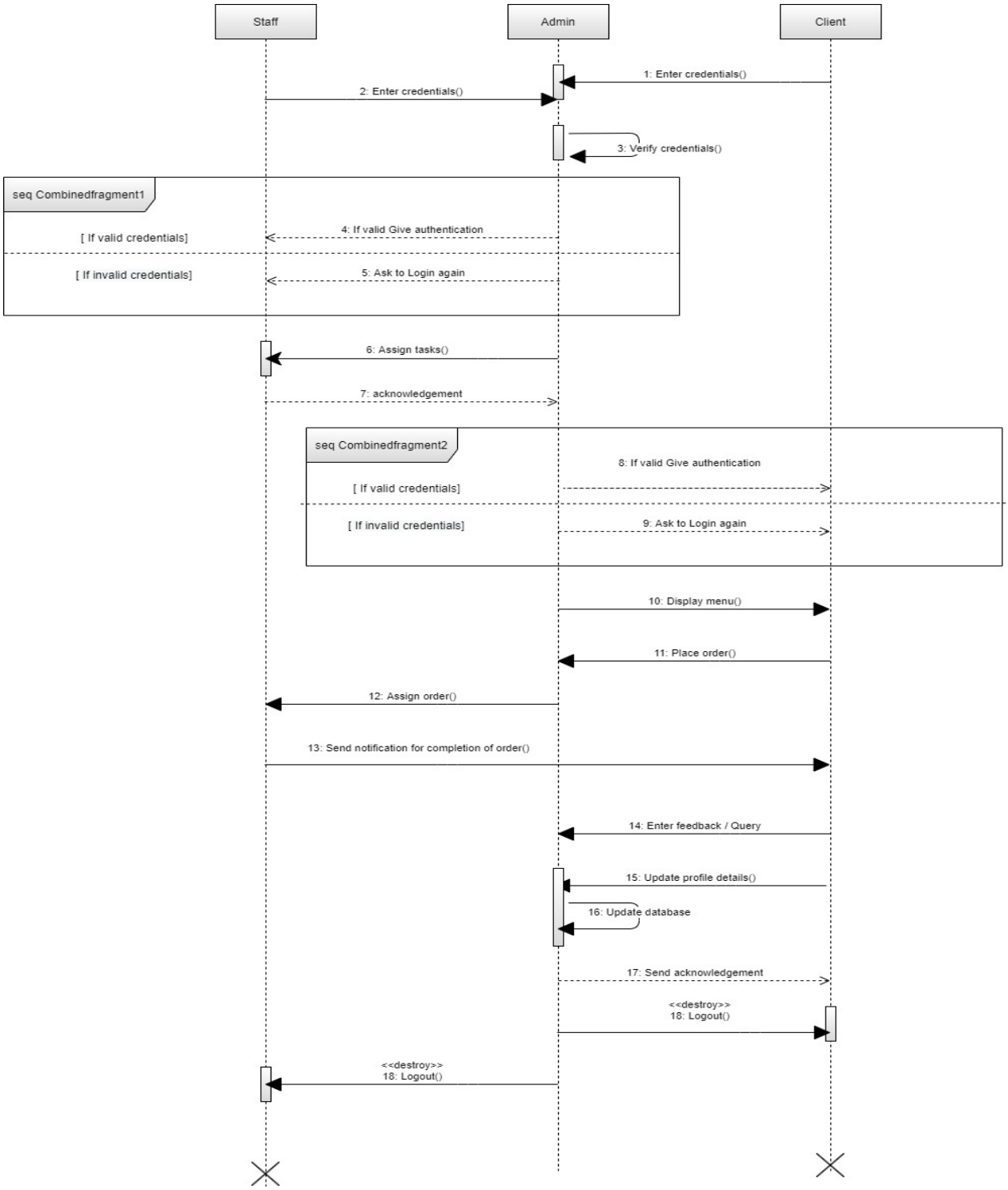






8.4 Analysis Models – Sequence Diagram

Following is the diagram that shows the overall sequence of events performed by the major components of the application.



8.5 Analysis Models – Class Diagram

Following is the diagram that shows the class split-up of the major components of the application.