Cafeteria Management System

SRS Group-3

Submitted by :

Sadikul Haque Sadi(bsse1003)

Mahdee Hasan(bsse1010)

Ahmed Sabbir(bsse1016)

Shezan Al Mahmud(bsse1023)

Arman Hossain(bsse1029)

Muntakim Hye Rashik(bsse1038)

Submitted to :

Prof. Dr. Kazi Muheymin-Us-Sakib

IIT
University of Dhaka

Table of Contents

1 INTRODUCTION 5

1.1 PURPOSE 5

1.2 INTENDED AUDIENCE 6

2 INCEPTION OF CAFETERIA MANAGEMENT SYSTEM 7

2.2 INCEPTION OF A CAFETERIA MANAGEMENT SYSTEM 8

2.2.1 IDENTIFY THE CLIENT OF OUR PROJECT 9

2.2.2 ICEBREAKING 9

2.2.3 IDENTIFYING THE STAKEHOLDERS OF THE CAFETERIA 9

3 ELICITATION OF CMS 12

3.1 INTRODUCTION 12

3.2 ELICITING REQUIREMENTS 12

3.2.1 COLLABORATIVE REQUIREMENTS GATHERING 13

3.2.2.1 NORMAL REQUIREMENTS 14

3.2.2.2 EXPECTED REQUIREMENTS 14

3.2.3 USAGE SCENARIO 15

4. SCENARIO BASED MODELING OF “Cafeteria Management System” 27

4.2 DEFINITION OF USE CASE 28

4.3 USE CASE DIAGRAMS 29

4.4 ACTIVITY DIAGRAM OF CMS 53

4.4.1 Cafeteria Management System 53

4.4.1.1 Authentication 55

4.4.1.1.1 Create account 56

4.4.1.2 Food management 57

4.4.1.3 Order 58

4.4.1.3.3 Service system 60

4.4.1.4.1 Initialize Booking 61

4.4.1.4.2 Modify Booking 63

4.4.1.5.1 Payment method 65

4.4.1.5.2 Payment type 67

4.4.1.6 Information management 69

4.4.1.6.4 Staff management 71

4.4.1.7 Activity Log 72

4.4.1.8 Feedback 74

4.5 SWIMLANE DIAGRAMS OF GMS 76

4.5.1 Cafeteria Management System 76

4.5.1.1 Authentication 78

4.5.1.1.1 Create account 79

4.5.1.2 Food management 81

4.5.1.3 Order 82

4.5.1.3.3 Service system 83

4.5.1.4.1 Initialize Booking 85

4.5.1.4.2 Modify Booking 87

4.5.1.5.1 Payment method 89

4.5.1.5.1 Payment type 90

4.5.1.6 Information management 92

4.5.1.6.4 Staff management 93

4.5.1.7 Activity Log 94

4.5.1.8 Feedback 95

5 DATA MODELING OF CMS 96

5.1 DATA MODELING CONCEPT 96

5.1.1.1 NOUN IDENTIFICATION 97

5.1.1.2 Potential Data Objects: 103

5.1.1.3 Analyzing and Finalizing DataObject: 105

5.1.1.4 Final Data Objects: 107

5.2 DATA OBJECT RELATIONSHIPS: 108

5.3 Schema Diagram 110

5.4 Entity Relation Diagram 117

Class based modeling of CMS 118

6.1 Class based modeling concept 118

6.2 GENERAL CLASSIFICATION 118

6.3 Selection Criteria 122

6.4 Class attributes and methods identification: 124

6.5 Finalizing Classes 129

6.6 Class Index Card 130

7 BEHAVIORAL MODELING OF GMS 141

7.1 STATE TRANSITION DIAGRAM: 142

7.1.1 EVENT AFTER ANALYSIS: 147

7.1.2 STATE TRANSITION DIAGRAM: 153

7.2 SEQUENCE DIAGRAM 162

7.3 DATA FLOW DIAGRAM: 163

1 INTRODUCTION

This section is a part of our software requirement specification for the project “Cafeteria Management System“. In this chapter, we will focus on the intended audience for this project.

1.1 PURPOSE

This document briefly describes the Software Requirement Analysis of Cafeteria Management System. It contains the normal, expected and exciting requirements and establishes a requirement’s baseline for the development of the system. The requirements contained in the SRS are independent, uniquely numbered and organized by topics. The SRS serves as an official means of communicating user requirements to the developer and provides a common reference point for both the developer team and the stakeholder community. The SRS will evolve over time as users and developers work together to validate, clarify and expand its contents.

1.2 INTENDED AUDIENCE

This SRS report is intended for teachers and officers of Dhaka University as well as the project managers, designers, developers, and testers of a software company.

- The customer will use this SRS to verify that the developer team has created a product that is acceptable to the customer.
- The project managers of the developer team will use this SRS to plan milestones and delivery date and ensure that the developing team is on track during the development of the system.
- The designers will use this SRS as a basis for creating the system's design. The designers will continually refer back to this SRS to ensure that the system they are designing will fulfill the customer's needs.
- The developers will use this SRS as a basis for developing the system's functionality. The developers will link the requirements defined in this SRS to the software they create to ensure that they have created a software that will fulfill all of the customer's documented requirements.
- The testers will use this SRS to derive test plans and test cases for each documented requirement. When portions of the software are complete, the testers will run their tests on that software to ensure that the software fulfills the requirements documented in this SRS. The

testers will again run their tests on the entire system when it is complete and ensure that all requirements documented in this SRS have been fulfilled.

1.3 CONCLUSION

This analysis of the audience helped us to focus on the users who will be using our analysis. This overall document will help each and every person related to this project to have a better idea about the project.

2 INCEPTION OF CAFETERIA MANAGEMENT SYSTEM

In this chapter, the Inception part of the SRS will be discussed briefly.

2.1 INTRODUCTION

The CARS cafeteria is a business organization which provides foods to the teachers and officers of Dhaka University only. There is a manager to runs this cafeteria with his staff. This document will act as a demand list of what the stakeholders of their respective wanted for their anticipated software to be. Like the various features and interfaces of the software which will help them to ease the way of their giving services

2.2 INCEPTION OF A CAFETERIA MANAGEMENT SYSTEM

At the beginning of our project, we entered the inception stage. This stage includes how the project will be started and its scope and limitation. The main goal of this phase is to identify the requirements, demand and establish some sort of mutual understanding between the software team and the stakeholders of the cafeteria. In order to make this phase effective we
took the following steps:

- Identifying the client of our project
- Icebreaking
- Identifying the stakeholders of the CARS cafeteria
- Identifying the multiple viewpoints of stakeholder

.

2.2.1 IDENTIFY THE CLIENT OF OUR PROJECT

At first, we identified the location from where we will start our expedition. We went to different departments and met with respected teachers and also officers of Dhaka University. We have analyzed our requirements with the consent of both of them.

2.2.2 ICEBREAKING

Icebreaking refers to the fact that to diminish the communication barrier between you and the other person. It is a crucial part since it denotes the acceptance of our proposal. We started this face by talking with them with context free languages. Their behavior, respond to our question or willing to make a change in the cafeterias service system solely depends on this phase.

2.2.3 IDENTIFYING THE STAKEHOLDERS OF THE CAFETERIA

Stakeholder refers to any person or group who will be affected directly or indirectly by the system. Stakeholders include end-users who interact with the system and everyone else in an organization who may be affected by its installation. The Cafeteria has only a limited number of stakeholders. Identification of the stakeholders was done from the information provided by the shopkeepers. Their names are given below:

- Teacher
- Officer
- CARS manager

2.2.4 IDENTIFYING THE MULTIPLE VIEWPOINTS OF THE STAKEHOLDER

Different stakeholders expect different benefits from the system as every person has his own point of view. So, we have to recognize the requirements of multiple viewpoints. Different viewpoints of the stakeholders about the expected software are given below:

Manager’s Viewpoint:

❖ First and foremost, a really friendly user interface

- Desktop-based software if affordable
- Provide signal when any food item is short
- Store information about people who are coming and working in the cafeteria
- Calculate the total amount of sell and the total amount of buying foods and the end of the month shows profit or loss using accounting software
- Keep information about salary deduction
- Data analytics
- Customer list
- Online ordering facility
- Provides notification to all the authorized customers of their payment, order, and booking
- Food menu will be shown
- Provide delivery system
- Providing multiple payment systems
- Must have a minimal cost

User’s Viewpoint:

- Ease of ordering food
- Minimum effort to use the software
- Reduce the time for ordering
- Viewing updated food menu
- Getting delivery system
- Mobile financial system for payment

2.3 Conclusion

The primary goal of this project is to model and design software for teachers and officers who receive services from the CARS cafeteria. For these reasons, the software will be designed in such a way that it won’t be a disaster for the client who will use it. The software will be as simple as a person who does not have any idea about software he/she can be able to maintain it without any annoyance. Otherwise, it will not be appreciable by the clients even it may create disturbance with their businesses. The software will be designed in such a way as it takes very little time to manage. To make this software project successful, collaboration with stakeholders was the main priority that what they want, how the software will work, how it can be more user friendly than the previous time, how it will save time and give services to more people at a time.

3 ELICITATION OF CMS

After discussing on the Inception phase, we need to focus on the Elicitation phase. So this chapter specifies the Elicitation phase.

3.1 INTRODUCTION

Requirements Elicitation is a part of requirements engineering that is the practice of gathering requirements from the teachers, manager, and officers. We have faced many difficulties, like understanding the problems, making questions for the stakeholders, limited communication with the stakeholders due to a short amount of time and volatility. Though it is not easy to gather requirements within a very short time, we have surpassed these problems in an organized and systematic manner.

3.2 ELICITING REQUIREMENTS

We have seen a Question and Answer (Q&A) approach in the previous chapter, where the inception phase of requirement engineering has been described. The main task of this phase is to combine the elements of problem solving, elaboration, negotiation and specification. The collaborative working

The approach of the stakeholders is required to elicit the requirements. We have finished the following tasks for eliciting requirements-

- Collaborative Requirements Gathering
- Quality Function Deployment
- Usage Scenarios
- Elicitation work products

3.2.1 COLLABORATIVE REQUIREMENTS GATHERING

We have met with many stakeholders in the Inception phase such as the teachers, officers, CARS staffs and manager. These meetings created an indecisive state for us to elicit the requirements. To solve this problem, we have met with the stakeholders (who are acting a vital role in the whole process) few times to elicit the requirements.

3.2.2 QUALITY FUNCTION DEPLOYMENT

Quality Function Deployment (QFD) is a technique that translates the needs of the customer into technical requirements for the software. Ultimately the goal of QFD is to translate subjective quality criteria into objective ones that can
be quantified and measured and which can then be used to design and manufacture the product. It is a methodology that concentrates on maximizing customer satisfaction from the software engineering

process. So, we have followed this methodology to identify the requirements for the project. The requirements, which are given below, are identified successfully
by the QFD.

3.2.2.1 NORMAL REQUIREMENTS

Normal requirements are generally the objectives and goals that are stated for a product or system during meetings with the customer. The presence of these requirements fulfills customers' satisfaction. These are the normal requirements for our project.

- First and foremost, a really friendly user interface
- Desktop-based software if affordable
- Store information about CARS staffs who are working in the cafeteria
- Calculate the total amount of sell and the total amount of buying foods and the end of the month or year shows profit or loss
- Keep information about the payment
- Must have a minimal cost
- Ease of calculation
- Minimum effort to use the software
- Reduce the time for ordering from the owner

3.2.2.2 EXPECTED REQUIREMENTS

These requirements are intrinsic to the product or system and may be so elementary that the customer does not explicitly state them. Their absence will be a cause for significant dissatisfaction. Below the expected requirements for our project are briefly described.

- Storing all cafeteria related information
- Interactive and attractive graphical user interface
- Authentication process
- Viewing activity log of authorized users

3.2.2.3 EXCITING REQUIREMENTS

These requirements are for features that go beyond the customer's expectations and prove to be very satisfying when present. The following are some exciting requirements of our project.

- Provide a reference-based signup system.
- Providing complaint hub system to make any complaint and give any suggestion.
- Providing a rating and commenting system for food items.
- Create an auto-generated financial report.

3.2.3 USAGE SCENARIO

Cafeteria management system (CMS) will be used by teachers and officers from the University of Dhaka. They must create an account to be an authorized user. Also, guest users are allowed to view food menu and order food with reference. Users will interact with the system through an interface. CMS provides the following functionalities -

1. Authentication: CMS can only be used by a valid user. A valid user is authenticated by the following methods -

(i) Create Account: There are two ways to create an account. They are the following:

➢ <u>With Reference:</u> Users must provide their name, employee id, designation, e-mail address, mobile number, department name,location, user id, password and a reference email id that already exists in our system(we have initiated some valid users from each department added by admin.Any valid users email id can be used as reference).Then a verification request will be sent to that reference user's mail. If the reference user verifies, a verification code consists of 6 digits that will be sent to the applicant's email. Once the applicant logs into the system via the verification code he/she will become a valid user.

➢ <u>Without Reference:</u> If the applicant does not provide any references then he/she must provide email id with the 'du.ac.bd' domain and then admin will match this information in Dhaka University's database. If the applicant is a teacher or officer then a verification code consists of 6 digits that will be sent to the applicant's email. Once the applicant logs into the system via the verification code he/she will become a valid user.

(ii) <u>Log in:</u> To get the services of CMS a user must log in to his/her account by providing username/email followed by a password. If the username and password are correct he/she enters the system. Otherwise, he/she can try again.

(iii) Forgot Password: If a user forgets his password, he will choose the reset password option and provide his email id. Then a verification code will be sent to his email id. Using this code he has to authenticate first and change his password.

(iv)change information: User can change his account information without any kind of verification process.

2. Food Management: Cafeteria provides lunch, snacks, and refreshment items.

(i) Food menu: Everyday's food menu are fixed with specific items.The admin initially update the quantity of the items according to that day's demand. Admin can also add, reduce and modify the food menu. After an order is confirmed, the quantity of that remaining food item will be automatically reduced. On the other hand, when an order is canceled, remaining food item is increased automatically. Anyone can see the food menu.

খাবারের মেনু এবং তালিকা		মূল্য
1	ভাত, সবজি, মুরগি, ডাল, সালাদ	৬৫
2	ভাত, সবজি, রুই মাছ, ডাল, সালাদ	৭৫
3	ভাত,সবজি, গরুর মাংস (হাড় ছাড়া) ডাল, সালাদ	১০০
4	রুটি (৪ পিছ), সবজি, ডাল, সালাদ, ডিম ভাজি	৫০
5	ভাত, সবজি, ডাল, সালাদ, ডিম ভাজি	৫০
6	ভাত, সবজি, ডাল , সালাদ	৫০
7	রুটি (৪ পিছ), মুরগি,	৬০
8	রুটি (৪ পিছ), গরুর মাংস (হাড় ছাড়া)	৮০

স্পেশাল মেনু		মূল্য
1	খিচুড়ি,বেগুন ভাজি,মুরগি,সালাদ	৮৫(রবি)
2	খিচুড়ি,বেগুন ভাজি,সালাদ,গরুর মাংস (হাড় ছাড়া)	১২০(রবি)
3	চাইনিজ খাবার(fried rice,fried chicken, veg)	১৫০(সোম)
4	খিচুড়ি,বেগুন ভাজি,মুরগি, সালাদ	৮৫(মঙ্গল)
5	খিচুড়ি,বেগুন ভাজি, সালাদ,গরুর মাং (হাড় ছাড়া	১২০(মঙ্গল)
6	চাইনিজ খাবার(fried rice,fried chicken, veg)	১৫০(বুধ)
7	পোলাও,বেগুন ভাজি,মুরগি,সালাদ	৮৫(বৃহস্পতি
8	পোলাও,বেগুন ভাজি,গরুর মাংস (হাড় ছাড়া),সালাদ	১২০(বৃহস্পতি

এক্সট্রা		মূল্য
1	ভাত/খিচুড়ি/পোলাও/f ed rice	৫/২০/২০/৩০
2	সবজি / চাইনিজ সবজি	১০/২৫
3	ডাল	১০
4	মুরগি	৩৫
5	গরু	৬০
6	মাছ	৪৫
7	BOX	১০

(ii) Stock management: Only admin can access stock management system. He can view how much food is stored in storage and he can calculate how much food to buy. He also can update and modify the quantity of foods and food items.

3. Order: After a successful login, users will be able to order food items from the food menu. If users order food manually, admin will add this order info to the order list. From the food menu, a user chooses food items with corresponding quantity.

It has the following functionality :

(i)Service system:

When he orders, he will get three options-

❏ Dine-in
❏ Take-away
❏ Delivery

If he chooses any of the sections, the cost will be shown to him and he will be redirected to the payment system for confirmation.

(1) Dine-in: If users want to eat their food in the cafeteria, they have to choose this service system. Seat reservation system is available here. So, users may reserve available seats for a particular time duration.

(2) Take-away: If users want to take their food home carrying by them, they need to choose this system.

(3) Delivery: If users want their food to be delivered to their office, they need to choose this foodservice system. They have to provide a location for the delivery. The delivery charge will be shown to the users.

(ii)Modify order: After ordering the food, users will be provided with a time duration between which users can change their selected food items.

(iii)Cancel order: After payment, the order will be confirmed and users will get a notification via email/SMS. Users will be provided with a time duration between which users can cancel the order. After the time duration, users will not be able to cancel the order.

(iv) Guest Order: Guest can also order from order section. At first guest, user needs to provide a reference i.e any valid user's email id to proceed. When the referenced user will forward it, the guest will be allowed to order food items. Guest users must provide location within Dhaka University otherwise order will not be accepted.

4. **Booking:** Cafeteria services also include a booking system for lunch and dinner. The cafeteria can also be booked for party lunch and buffet. In case, any user wants to arrange any of these programs, he/she must use the booking system.

Users will choose a date and time for their programs. If the time slot isn't available, they will have to choose another date and time. Users will choose the food menu with the corresponding quantity for the program. Then the total cost for the arrangement at the program will be shown. After that, if users want to book, he needs CARS head's permission. So admin will redirect the request to the head and if he accepts then admin will send a notification to the applicant for payment. Then the user will be redirected to the payment section. Until the user pays and confirms booking the timestamp will be available for other users.

After the booking request is accepted, the user will get a notification via email/SMS with time duration (no other user can book on that request date and time). Within this time, users have to pay the full amount of cost. Otherwise, the booking will be canceled. Once users pay the full amount of cost, the booking will be confirmed.

5. Payment system: Every payment is recorded with transaction id and user id and later added to accounting software.

(i) Payment method: Each user will be able to pay the bill by the following procedures:

(1)Pre-paid system: Users can pay advance money through their accounts. When users consume or ordering food, a certain amount will be deducted from their accounts and a notification will be sent to them via email/SMS which includes remaining balance.

(2)Salary deduction: In this system, users can order food items without instant payment. It'll keep a record of the total cost of the orders of a month. After the end of the month, that amount of cost will be deducted from his salary.

(3)Mobile Financial Services(MFS): Here, users can pay their bills with their bKash, Rocket, iPay and Nagad mobile apps. He'll be notified via email/SMS after the payment is successful. If any new MFS system comes, the admin can add this to the system. Guest users can pay through MFS.

(4) Credit card: If users are interested in paying their bills through credit cards, they can pay through the merchant's credit card reader or e-commerce site. The system will send a request to the BluePay(a third-party agent) to route the transaction information. If payment is done, a notification will be sent to the user. Guest users can pay through credit card.

(5) Cash-On: When any user pays cash-on the receiver will add it manually to the accounting software and also to the CMS system with details. This time users will not be notified.

(6) combined method: Users are allowed to pay their dues via multiple methods. When a user wants to pay he/she can choose multiple payment methods and the corresponding amounts from the method. But the summation of the amount should be exact the due amount(cannot exceed or less).

(ii) Payment type:

(1) Payment from Order: After the user chooses the order, he will be redirected to the payment methods. Then, he has to pay via any payment method to confirm his order. Otherwise, order request will not be taken.

(2) Payment from Booking: After the user wants to book he will be redirected to the payment system. He has to pay a fixed amount of cost in advance (for example 5%). Admin can fix and change this. If he pays, booking request will be approved.

6.Information management: Only the admin can access this system. It includes the following functionalities**:**

(i)Staff management: Here, admin manages staff assigns them to different works. Admin can view whether staff is available or not and if available which work he is assigned to. It also comprises staff status whether he's active or not. Then admin can reassign any available and not active staff to remaining works.

(ii) Human Resource Management: This section deals with all the information, including personal information, designation, salary, contact information of all staff of the cafeteria. Admin can modify any information from any staff.

(iii)Accounting system: It comprises daily, weekly and monthly income, expenditure, profit and loss of the cafeteria from external software.

Here used Tally. Only admin can update, access and modify this information.

(iv) **Data Analytics:** It informs us of the business condition and recommends future plans for the cafeteria on the information provided by the accounting system. It analyzes the relative demand for every food item. It also finds out on which days customers' pressure is relatively higher. System will create a monthly financial report based on the data analytics and accounting software. Only admin can view and download this financial report.

7. Activity Log: Whenever users interact with the system activity log will be recorded with time. Admin can view the Activity Log of any users. Authorized users can view and delete their own activity log. If guest users order food and make a payment, this activity will be recorded against the reference user's id. For every users, ip_address and timestamp will be recorded which can only see by admin.

8. FeedBack: Authorized users can give feedback about cafeteria’s services. They can rate and comment on food items, delivery systems, and booking systems. There is also a complaint hub to post any complaint about the cafeteria’s overall service system.

Only admin can view these feedbacks and give a reply to them.

- Rating and comment: Authorized users are allowed to give a rating about any food item they consumed, the delivery system they were serviced, the arrangement system of their bookings and make comment on their ratings. This rating will be recorded and shown in the system. System will ask the user to rate and also users can rate their items whenever they want.
- Complaint hub: Authorized users can post any complaint and give suggestions about the cafeteria. Only admin can view the complaint hub and users can view their own complaints only.

4. SCENARIO BASED MODELING OF “Cafeteria Management System”

4.1 INTRODUCTION

Although the success of a computer-based system or product is measured in many ways, user satisfaction resides at the top of the list. If we understand how end users (and other actors) want to interact with a system, our software team will be better able to properly characterize requirements and build meaningful analysis and design models. Hence, requirements modeling begins with the creation of scenarios in the form of Use Cases, activity diagrams and swim lane diagrams.

4.2 DEFINITION OF USE CASE

A Use Case captures a contract that describes the system behavior under various conditions as the system responds to a request from one of its stakeholders. In essence, a Use Case tells a stylized story about how an end user interacts with the system under a specific set of circumstances. A Use Case diagram simply describes a story using corresponding actors who perform important roles in the story and makes the story understandable for the users. The first step in writing a Use Case is to define that set of “actors” that will be involved in the story. Actors are the different people that use the system or product within the context of the function and behavior that is to be described. Actors represent the roles that people play as the system operators. Every user has one or more goals when using system.

Primary Actor

Primary actors interact directly to achieve required system function and derive the intended benefit from the system. They work directly and frequently with the software.

Secondary Actor

Secondary actors support the system so that primary actors can do their work. They either produce or consume information.

4.3 USE CASE DIAGRAMS

Use Case diagrams give the non-technical view of overall system.

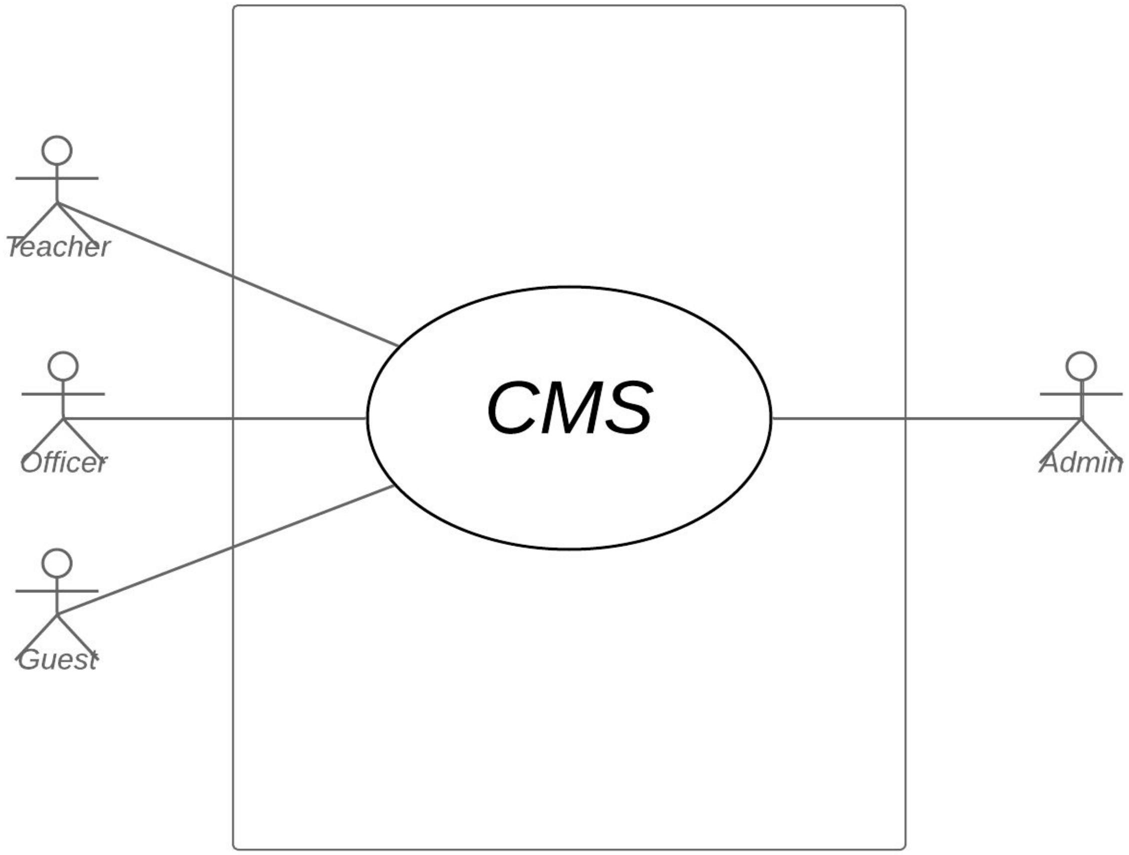


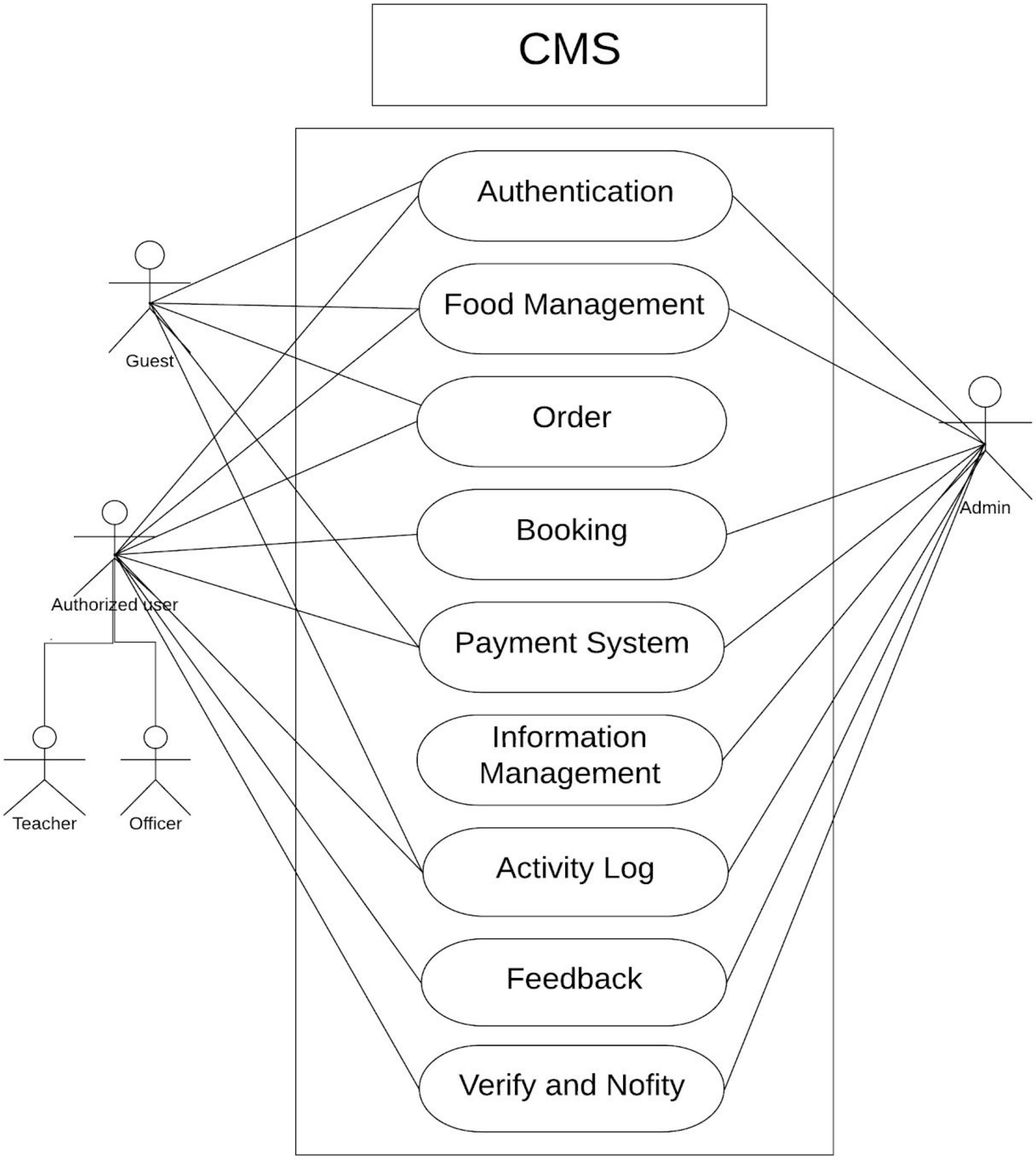
figure1:main

Description of Use case diagram level-0:

After analyzing the user’s story, we found 4 actors that directly or indirectly interacts with the system. Primary actors are those who will play action and get a reply from the system whereas secondary actors only produce or consume information.

The actors are -

1. Admin
2. Teacher
3. Officer
4. Guest



Level: 1

figure2:CMS

Description of level-1 use case diagram:

There are nine subsystems in Cafeteria Management System. They are as follows:

1. Authentication
2. Food Management
3. Order
4. Booking
5. Payment System
6. Information Management
7. Activity log
8. Feedback
9. Verify and Notify

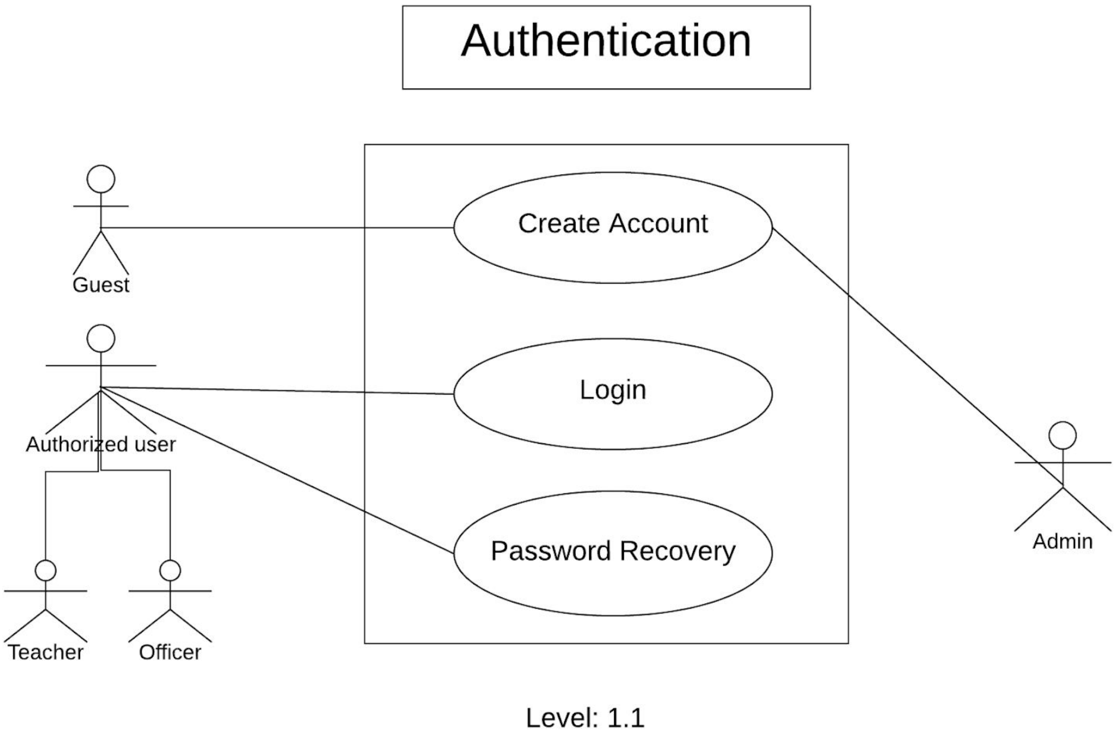


figure3:authentication

Description of level-1.1 use case diagram:

CMS can be used by authorized user and guest user. A guest user can authenticate by creating an account. Authentication section has following functionalities -

1. Create Account
2. Login
3. Password Recovery
4. Edit Account

Action and Reply:

Action-1: Users provide credentials info to create an account.

Reply-1: For valid information system will allow users to create an account.

Action-2: Users provide invalid credentials.

Reply-2: System will show error message and allows to try again.

Action-3: Users provide information that he/she wants to recover password.

Reply-3: System will send request code to the user's mail address and give permission to the users for the recovery of password.

Action-4: Users change their account information.

Reply-4: Changed info will be saved to the database.

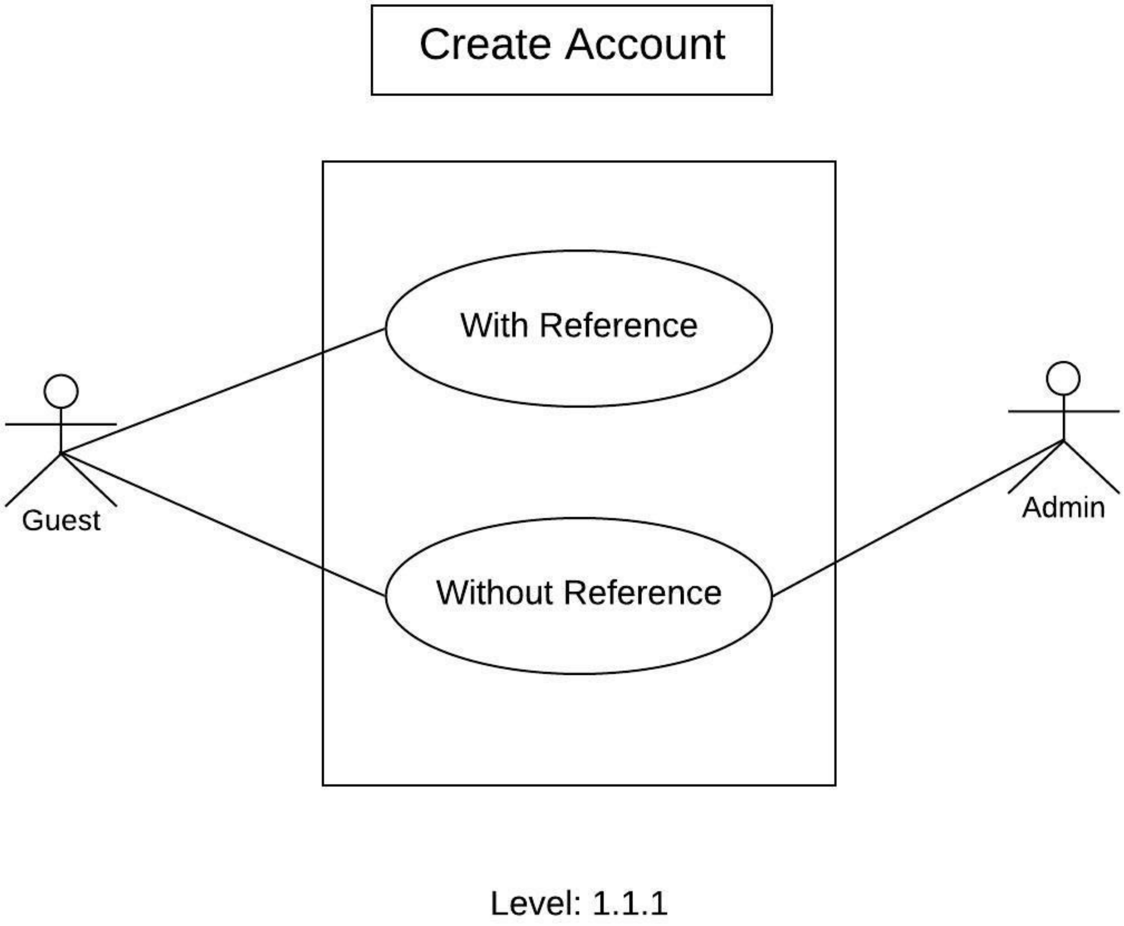


Figure4:create account

Description of level-1.1.1 use case diagram:

There’s two ways to create an account. They are following:

1. With reference
2. Without reference

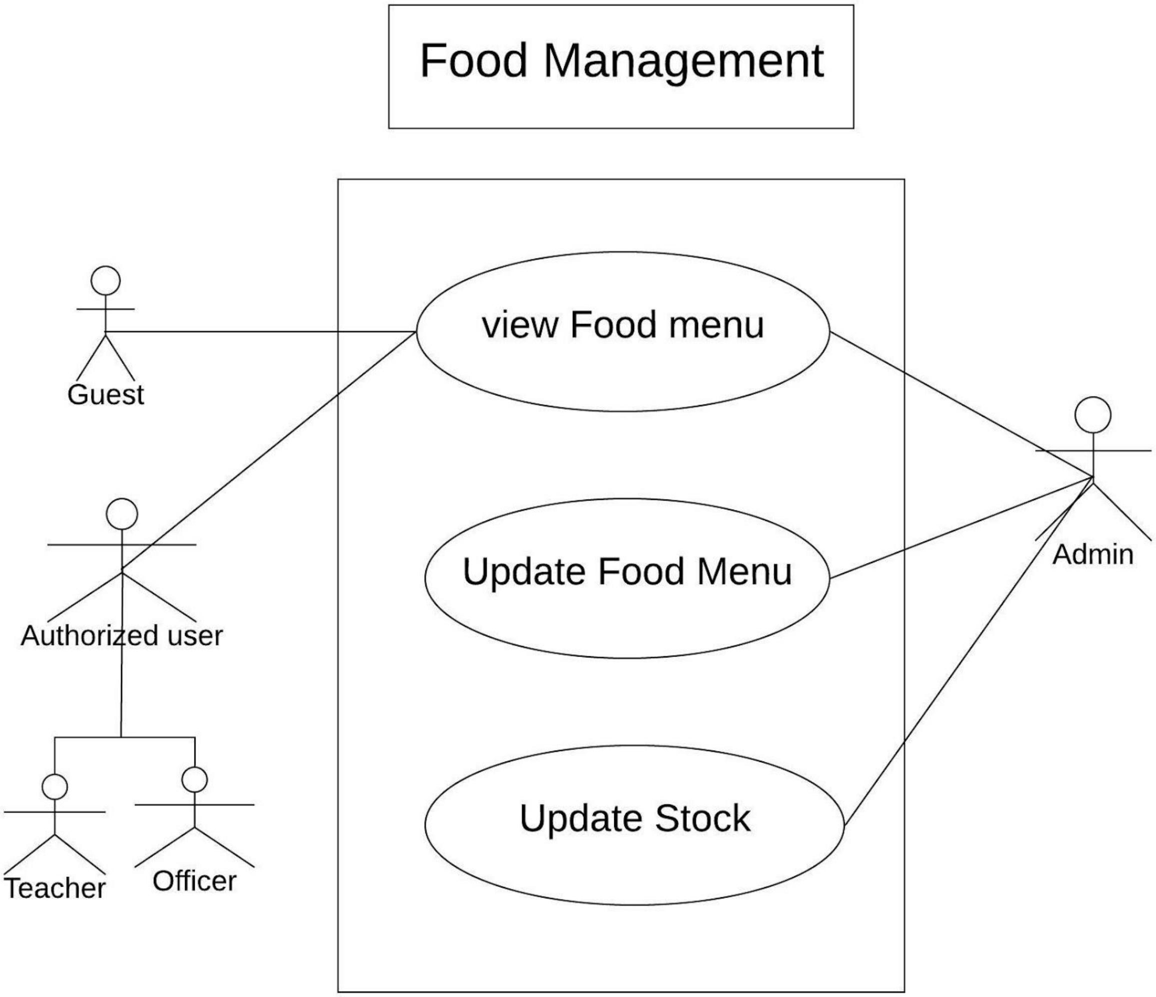
Action and Reply:

Action-1: Users provide credentials without reference email address.

Reply-1: Admin will check the given credentials. For valid information system will allow users to create an account.

Action-2: Users provide credentials with reference email address.

Reply-2: Selected user from each department who added by admin will check the given credentials. For valid information and valid reference system will allow users to create an account.



Level: 1.2

Figure5:food management

Description of level-1.2 use case diagram:

Everyday's of a week, food menu is fixed with specific items.

This Food Management subsystem can be divided into three parts. They are as follows:

1. View food menu
2. Update food menu
3. Update stock

The admin initially update the quantity of the items according to that day's demand. Admin can also add, reduce and modify the food menu in future if needed. After an order is confirmed, the quantity of that remaining food item will be automatically reduced by the system. On the other hand, when an order is canceled, remaining food item is increased automatically. Anyone can see the food menu.

Action and Reply:

Action-1: Users want to show the food menu.

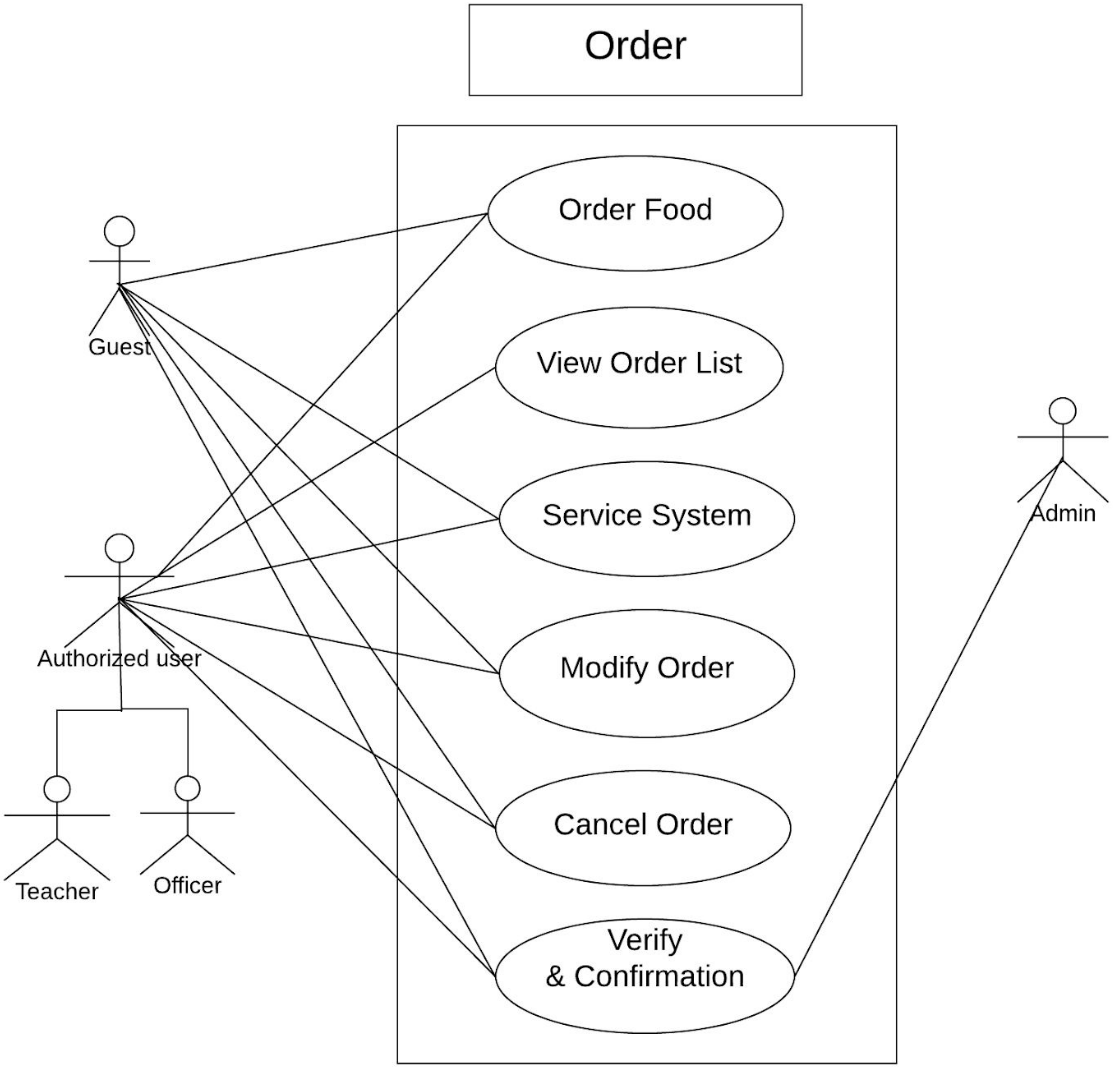
Reply-1: System will show the food menu via interface.

Action-2: Admin update food menu.

Reply-2: Updated food info will be saved in database.

Action-3: Admin update stocked food quantity.

Reply-3: Updated stocked food info will be saved in database.



Level: 1.3

figure6:order

Description of level-1.3 use case diagram:

After successful login authorized users will be able to order food items from the food menu. If users order food manually, admin will add this order info to the order list. From the food menu, a user chooses food items with corresponding quantity. A guest user can also order food with reference email address.

It has the following functionality **:**

1. Order Food
2. View order list
3. Service system
4. Modify order
5. Cancel order
6. Confirmation

Action and Reply:

Action-1: User chooses and confirms food items.

Reply-1: Admin will add this order info to the order list.

Action-2: User modifies order in time.

Reply-2: System will modify this order info to the order list.

Action-3: User modifies order after the fixed time.

Reply-3: System won't allow to modify the order.

Action-4: User cancels order in time.

Reply-4: System will cancel the order.

Action-5: User cancels order after the fixed time.

Reply-5: System won't allow to cancel the order.

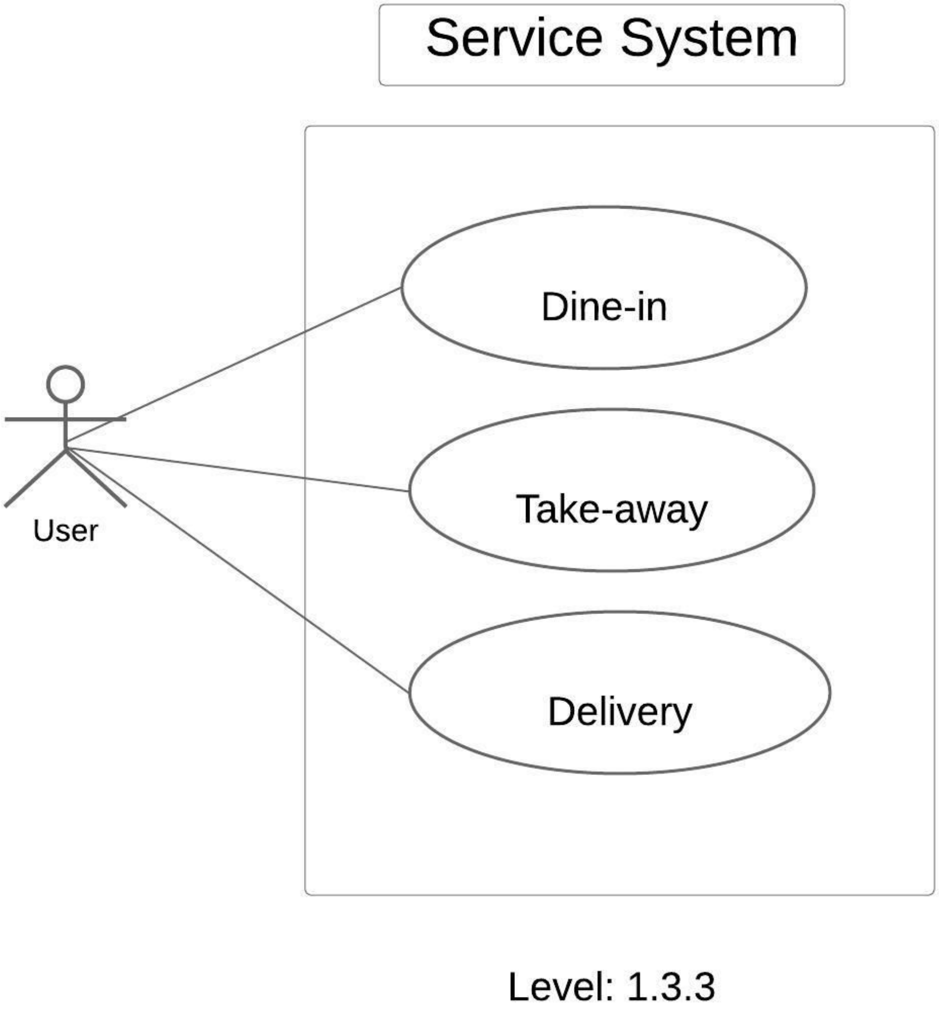


Figure7:service system

Description of level-1.3.3 use case diagram:

When a user orders, he/she will get three options-

❑ Dine-in

❑ Take-away

❑ Delivery

Action and Reply:

Action-1: Seats available and user chooses dine-in option.

Reply-1: System will allow.

Action-2: No seats available and user chooses dine-in option.

Reply-2: System won’t allow.

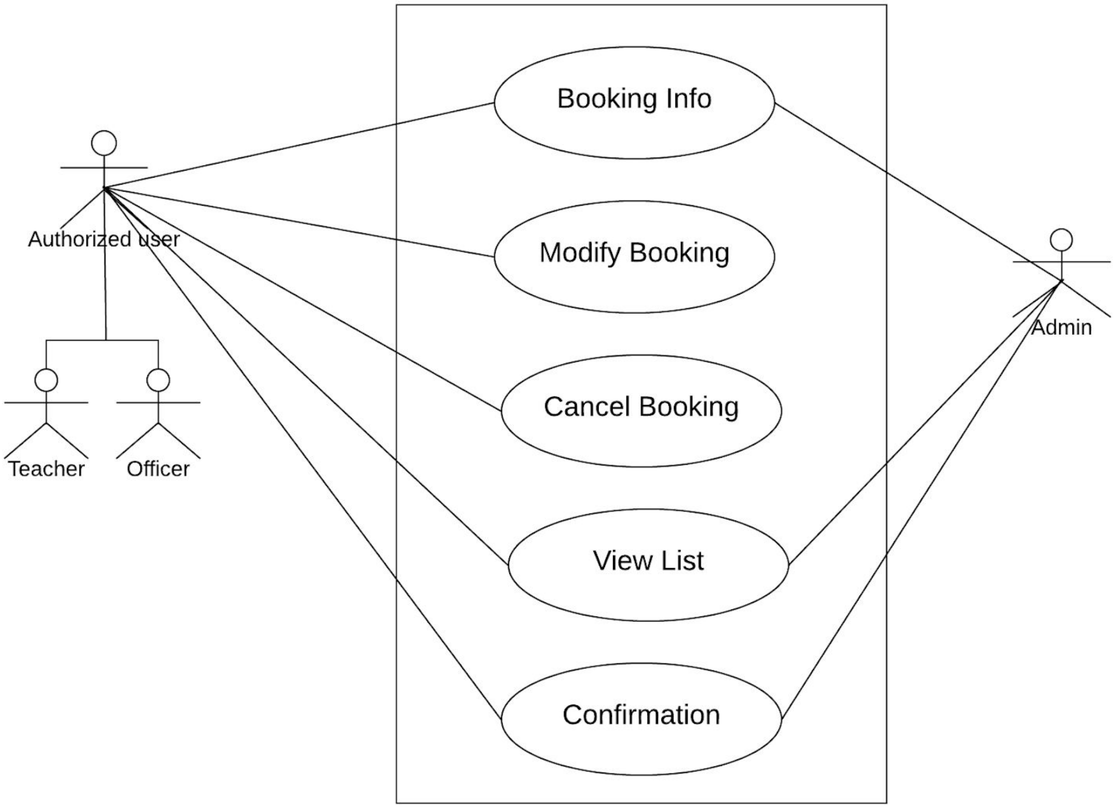
Action-3: User chooses take-away option.

Reply-3: System will allow.

Action-4: User chooses delivery option.

Reply-4: System will allow and assign a staff to deliver the order.

Booking



Level: 1.4

figure8:booking

Description of level-1.4 use case diagram:

In this Booking subsystem, users can modify booking and this subsystem has five parts. They are:

1. Booking info
2. Modify booking
3. Cancel booking
4. View list
5. Confirmation

Action and Reply:

Action-1: User chooses date and time for booking, selects food menu with corresponding quantity for the program.

Reply-1: System will allow to confirm the booking and show the total cost for the arrangement.

Action-2: User chooses date and time for booking but the date and time slots aren't available.

Reply-2: System won't allow to confirm the booking.

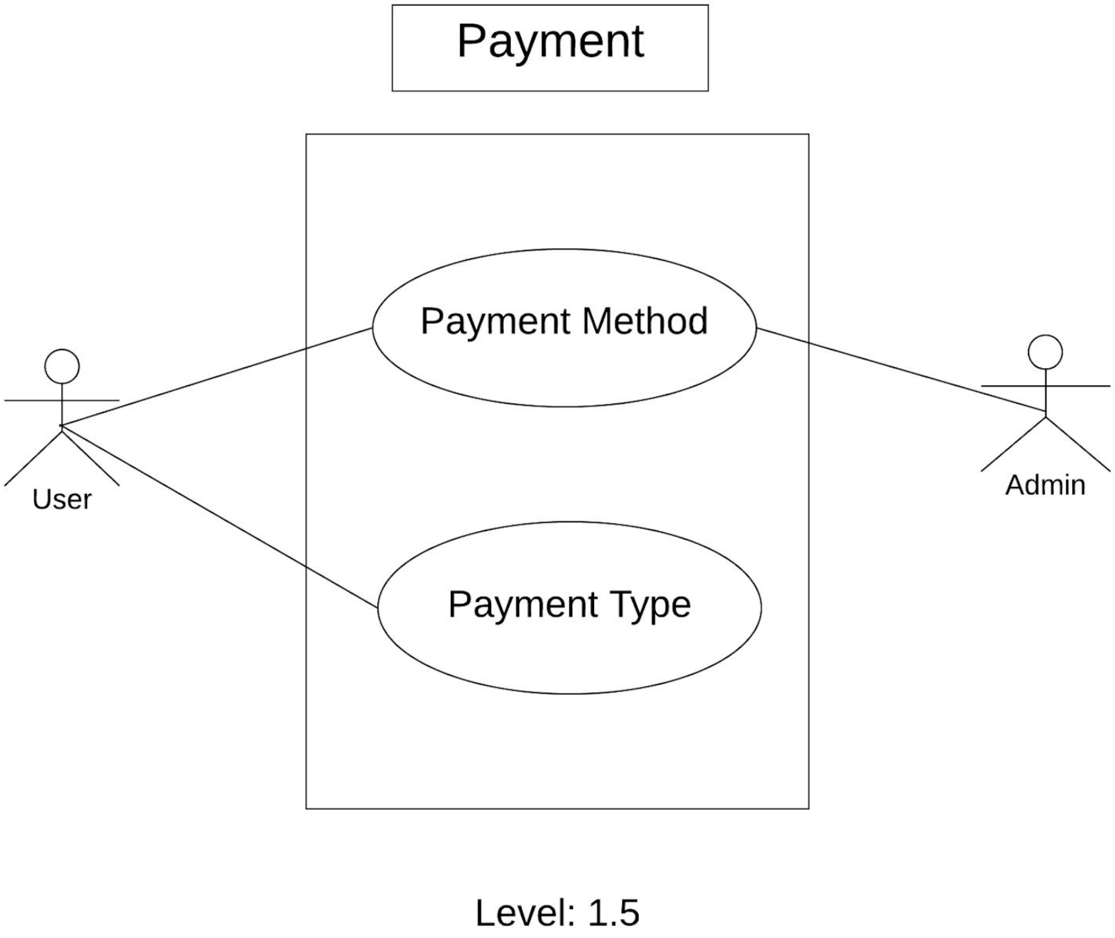
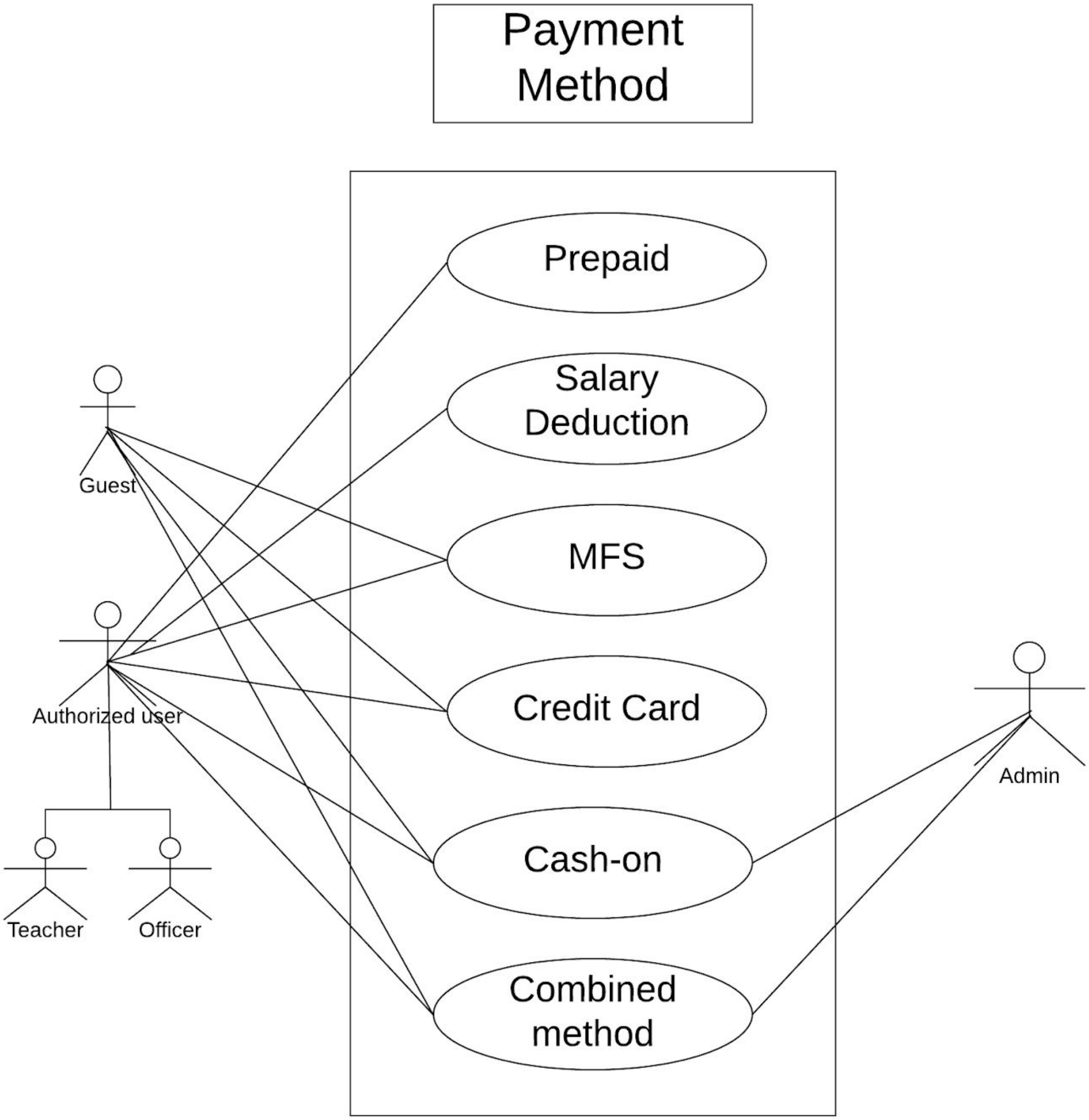


figure9:payment

Description of level-1.5 use case diagram:

Every payment is recorded with transaction id and user id and later added to accounting software. It has two parts-

1. Payment method
2. Payment type



Level: 1.5.1

Figure10:payment method

Description of level-1.5.1 use case diagram:

Each user will be able to pay the bill by the following procedures**:**

1. Pre-paid system
2. Salary deduction
3. Mobile Financial Services(MFS)
4. Credit card
5. Cash-On
6. combined method

Action and Reply:

Action-1: User wants to pay through prepaid system.

Reply-1: System will allow and update the database.

Action-2: User wants to pay through salary deduction.

Reply-2: System will allow and sent the notification through email/sms.

Action-3: User wants to pay through Bkash/Rocket.

Reply-3: System will allow and update the database.

Action-4: User wants to pay through credit card.

Reply-4: System will allow and update the database.

Action-5: User wants to pay through cash-on.

Reply-5: System will allow and Admin update the database.

Action-6: User wants to pay through combined method.

Reply-6: System will allow and sent the notification through email/sms.

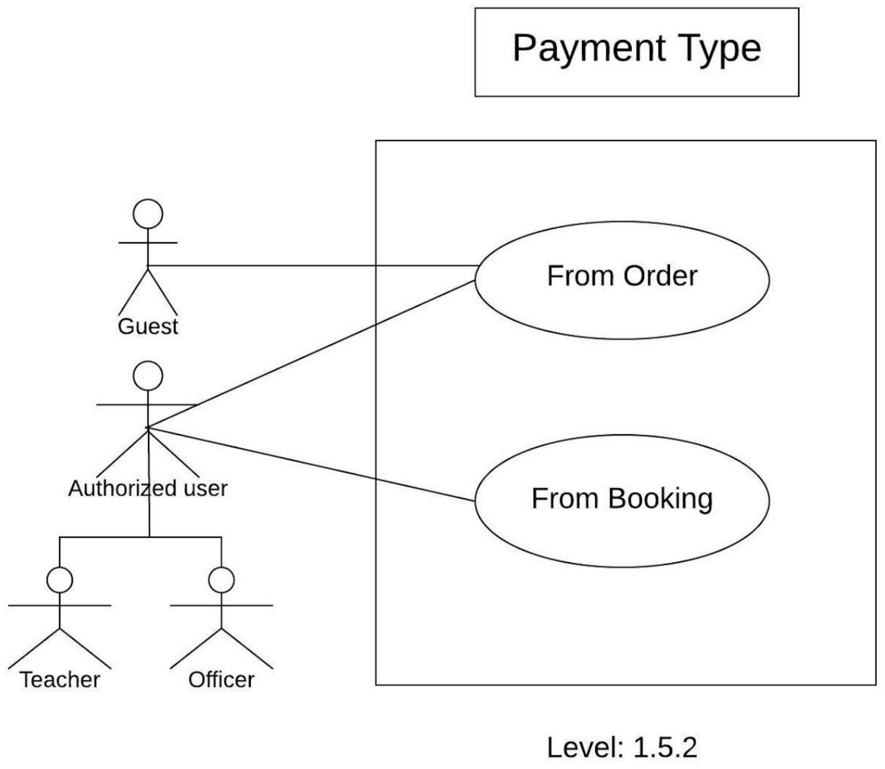


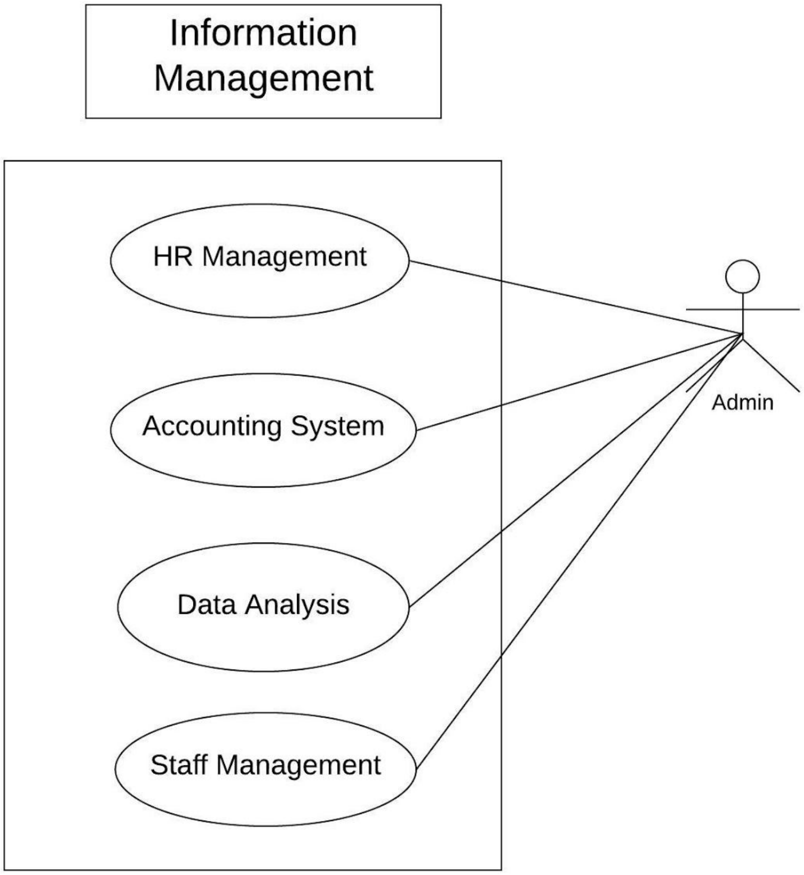
Figure11:payment type

Payment type has two parts-

1. Payment from order
2. Payment from booking

Action-1: Authorized users will choose order type(for booking or order). But guest users will only choose order type.

Reply-1: System will allow to do that.



Level: 1.6

Figure12:information management

Description of level-1.6 use case diagram:

Only the admin can access this system. It includes the following functionalities:

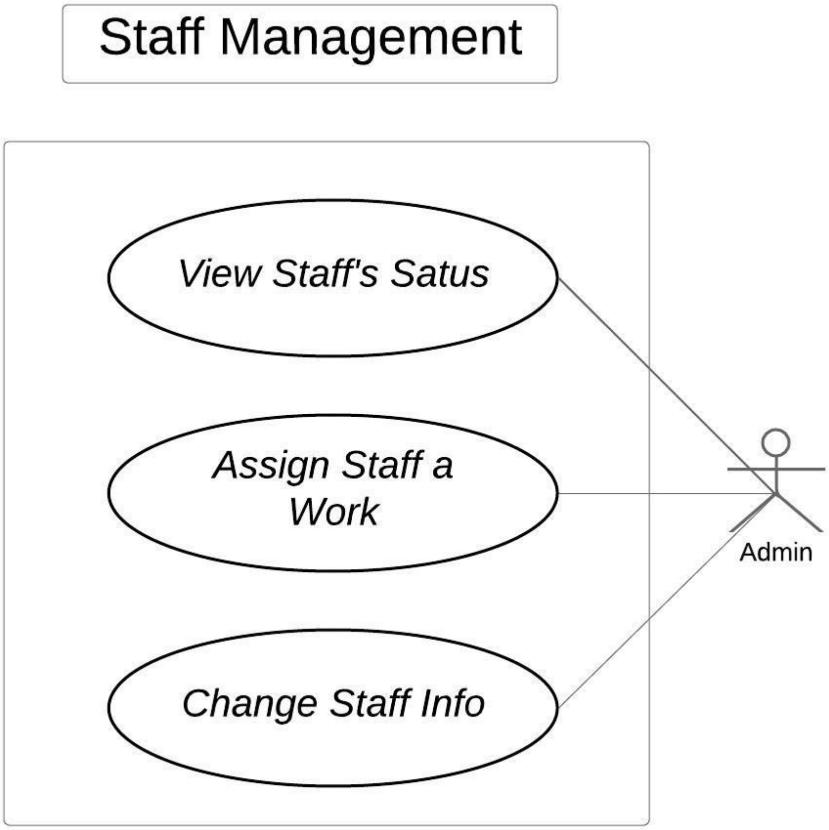
1. Human Resource Management
   2. Accounting system
   3. Data analysis
   4. Staff management

Action-1: Admin will view, update, insert, and delete info from database.

Reply-1: System will allow and connect to the database.

Action-2: Admin can show financial report.

Reply-2: System will calculate data from Accounting system and give report data to admin.



Level: 1.6.4

Figure13:staff management

Description of level-1.6.4 use case diagram:

Here, admin manages staff assigns them to different works. Admin can view whether staff is available or not and if available which work he is assigned to. It also comprises staff status whether he’s active or not. Then admin can reassign any available and not active staff to remaining works.

Action-1: Admin will view staff’s information from the database via HR management.

Reply-1: System will allow to do that.

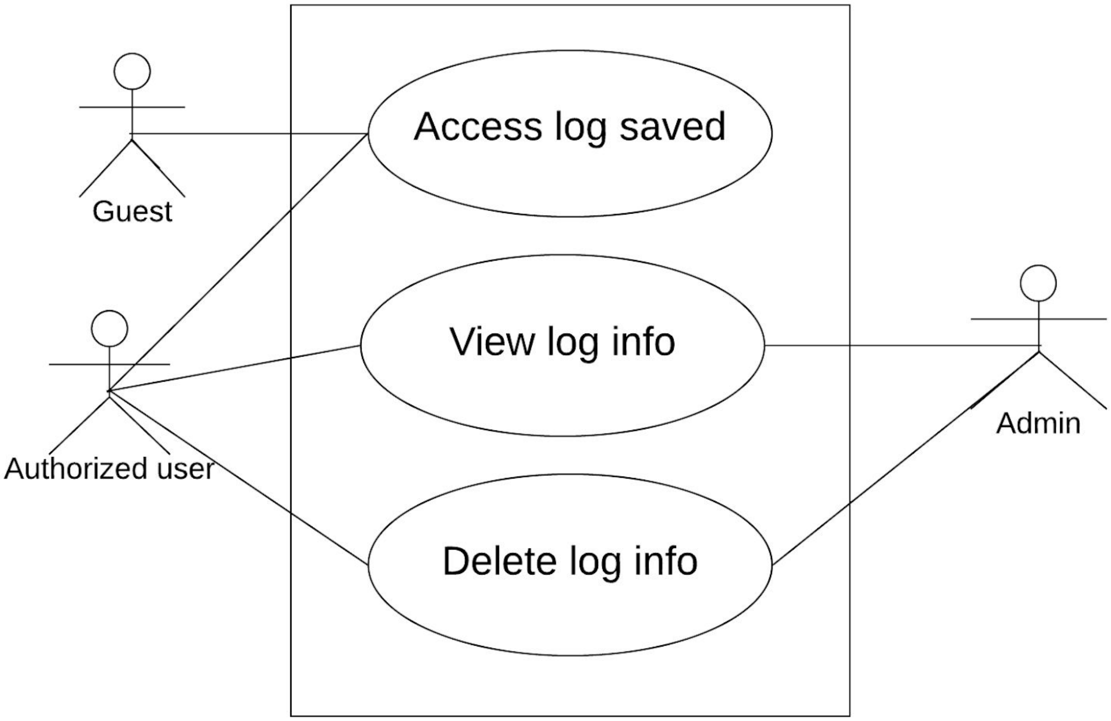
Action-2: Admin can assign a staff work by changing his status.

Reply-2: System will allow to change their status.

Action-3: Admin will change staff’s personal info.

Reply-3: System will store this changed info in database.

Activity Log



Level: 1.7

figure14:Activity log

Description of level-1.7 use case diagram:

Whenever any users interact with the system activity log will be recorded with timestamp and IP address. Admin can view the Activity Log of any users. Authorized users can view and delete their own activity log. If guest users order food and make payment, this activity will be recorded against the reference user's id. For every users, ip_address and timestamp will be recorded which can only see by admin.

Action-1: Authorized users and guest users access the system.

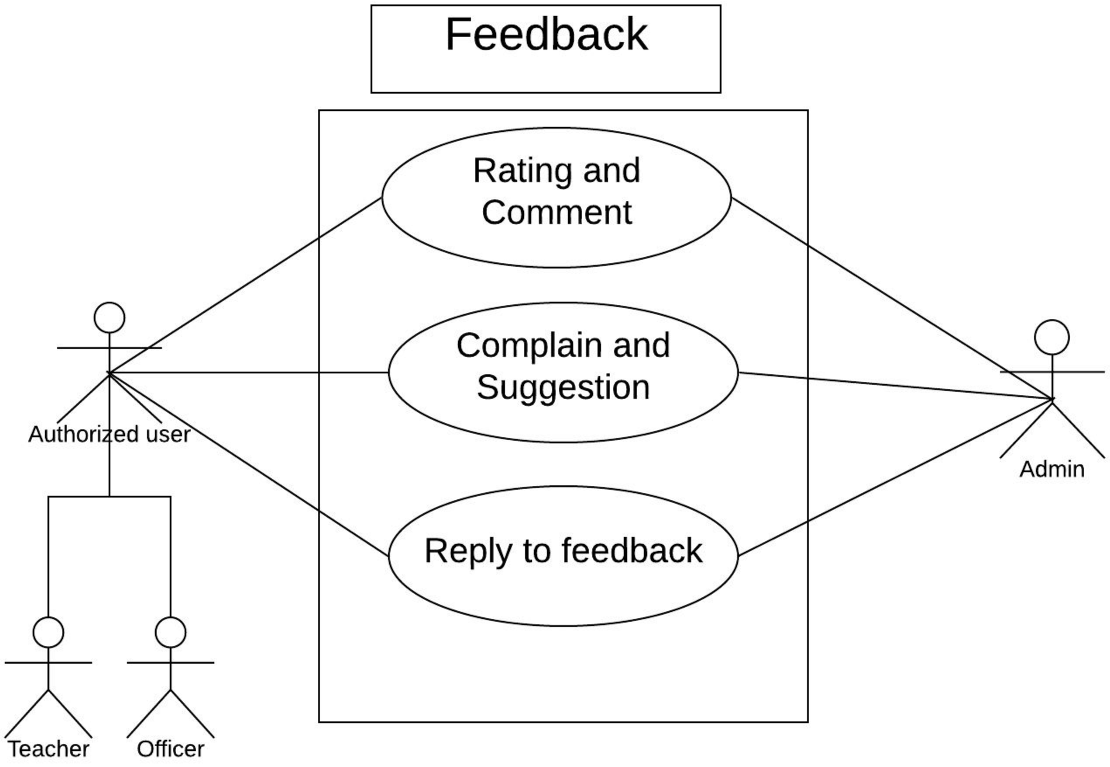
Reply-1: System will store log info into the database.

Action-2: Authorized users show and delete his own log info.

Reply-2: System will give permission to do that.

Action-3: Admin shows and deletes all users log info.

Reply-3: System will allow to do that.



Level: 1.8

figure15:feedback

Description of level-1.8 use case diagram:

Authorized users can give feedback on cafeteria’s services. They can rate and comment on food items, delivery systems, and booking systems. There is also a complaint hub to post any complaint about the cafeteria’s overall service system.
Only admin can view these feedbacks and give a reply to them.

Action and Reply:

Action-1: Authorized users want to comment and rate on the cafeteria's services.

Reply-1: System will allow and admin will give the reply.

Action-2: Authorized users can complain about cafeteria’s services.

Reply-2: System will allow and Admin can view all complain.

Action-3: Admin can give feedback against comment.

Reply-3: System will allow to give feedback which is shown by authorized users.

4.4 ACTIVITY DIAGRAM OF CMS

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration, and concurrency.

4.4.1 Cafeteria Management System

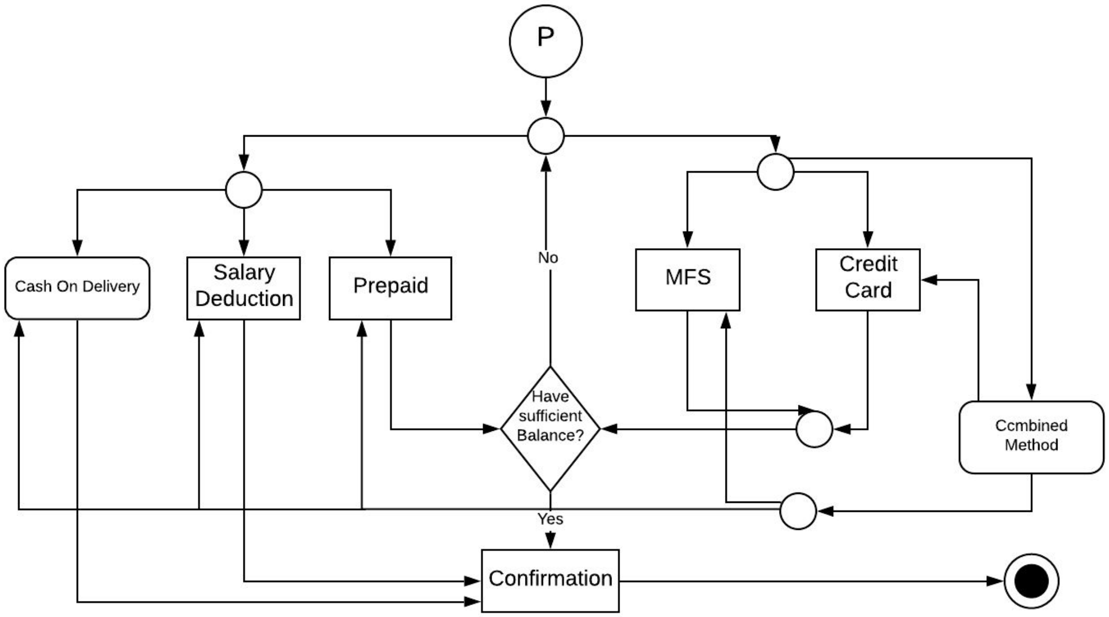


Figure:CMS

figure16:ac_CMS

4.4.1.1 Authentication

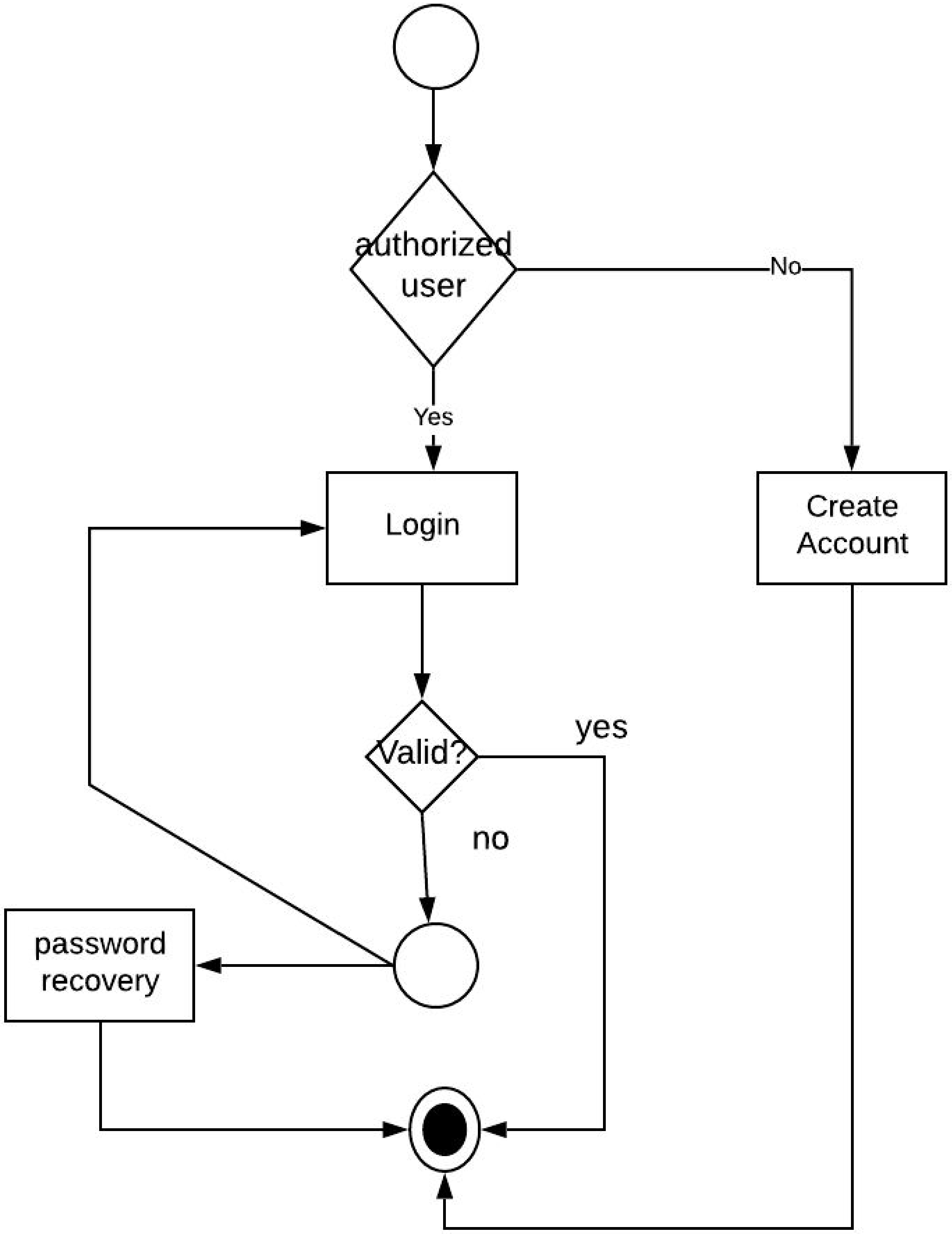
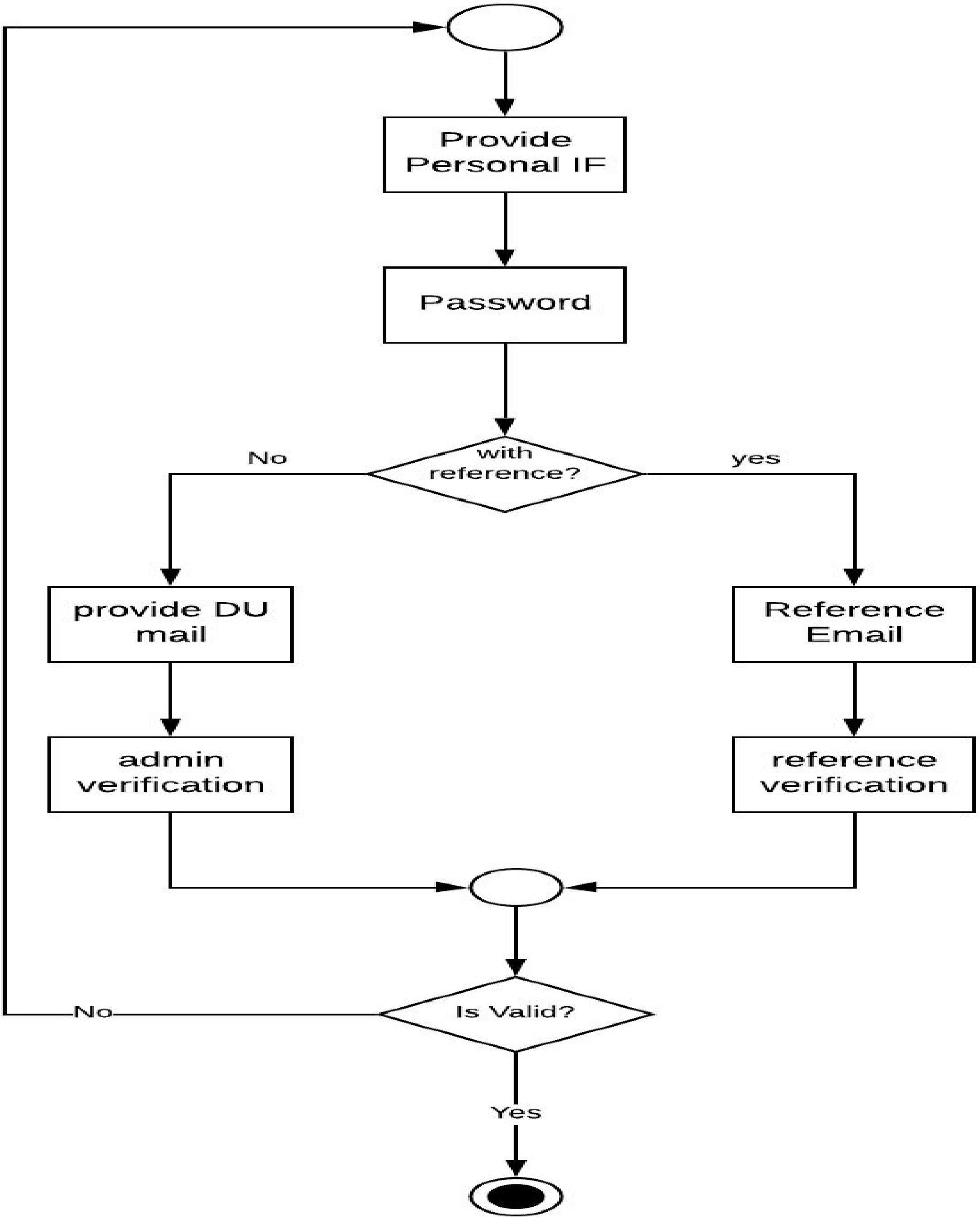
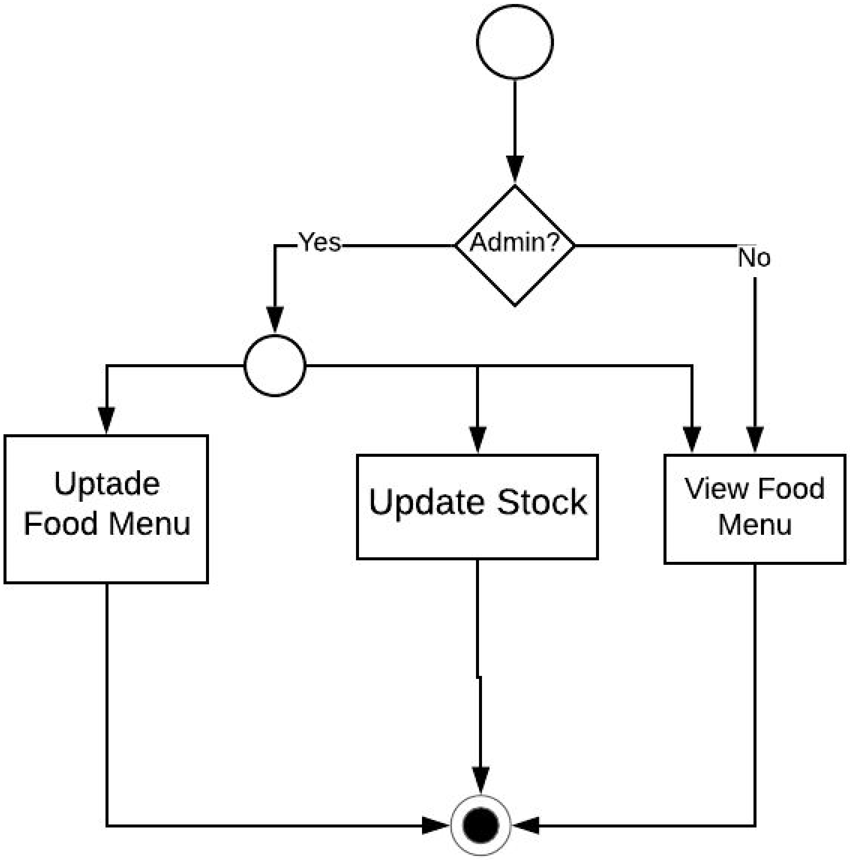


figure17:ac_authentication

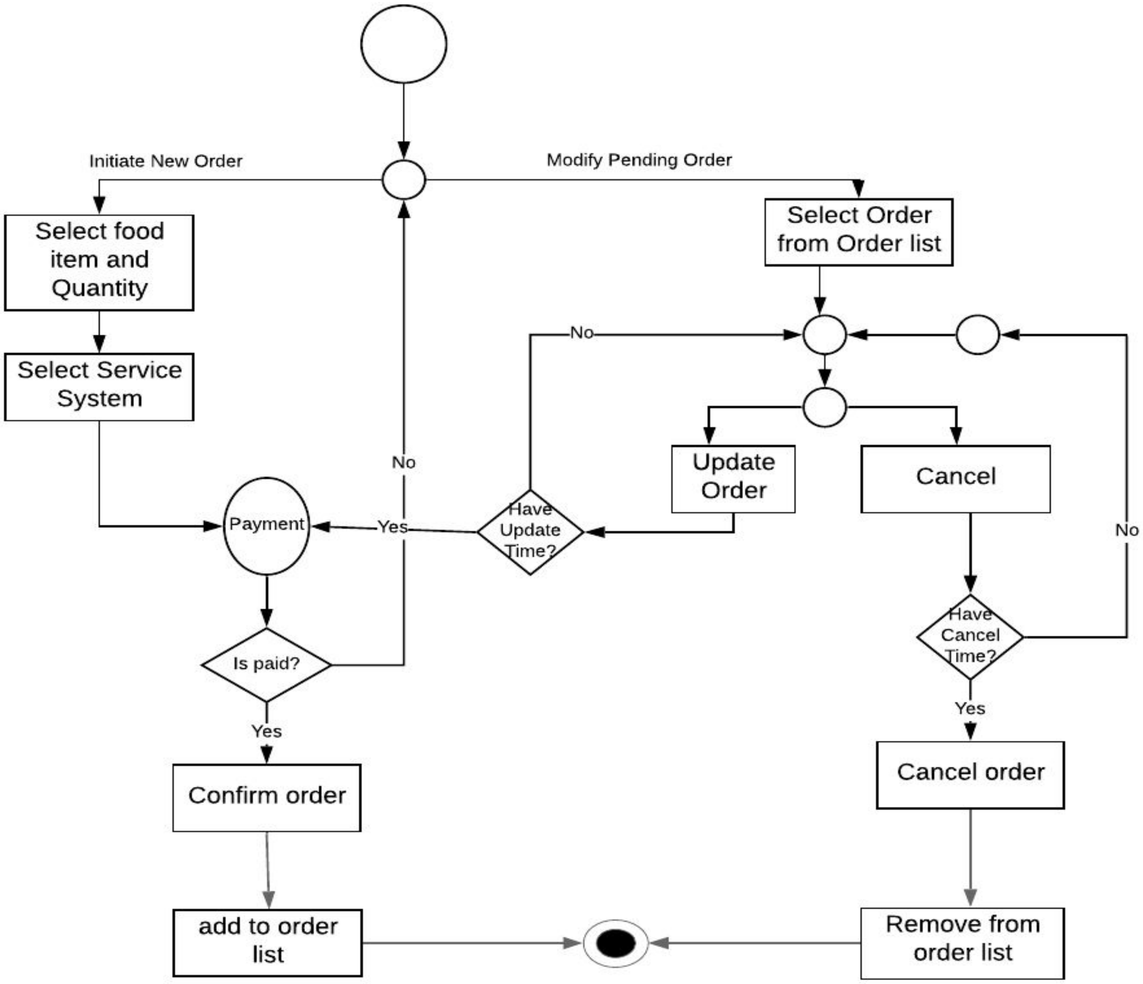
4.4.1.1.1 Create account



4.4.1.2 Food management

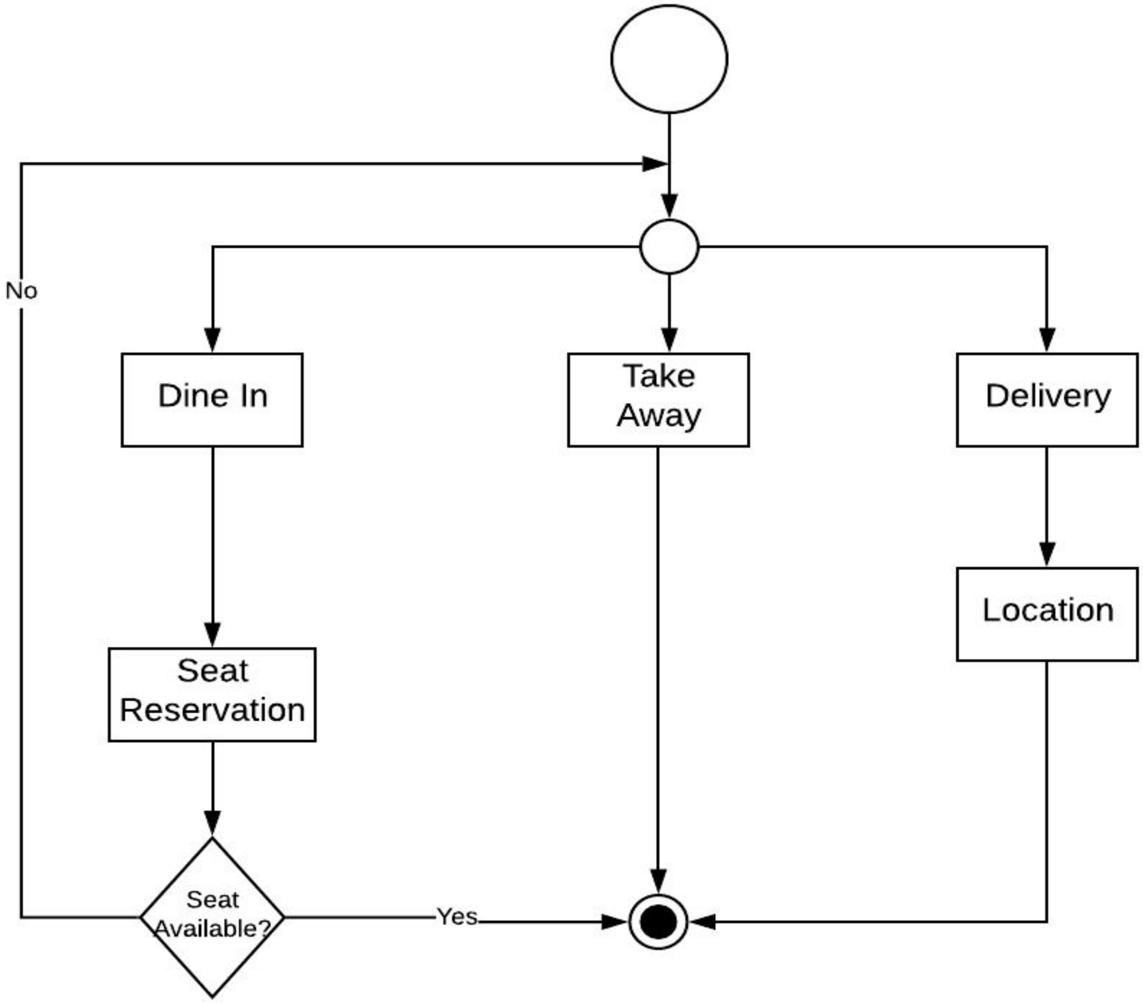


4.4.1.3 Order

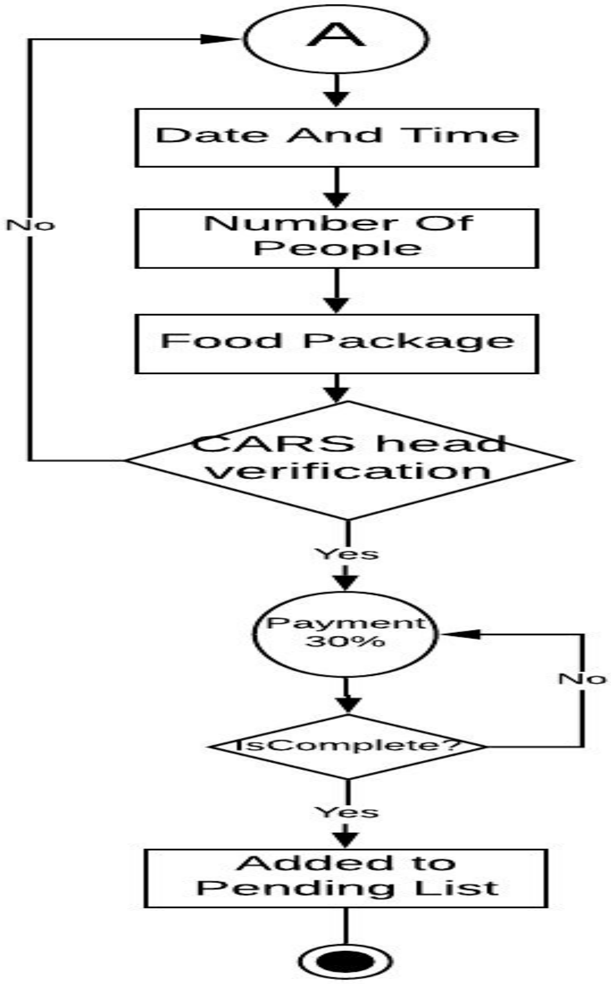


Initiate New Order
Modify Pending Order
Select food item and Quantity
Select Order from Order list
Select Service System
No
Update Order
Cancel
No
No
Payment
Yes
Have Update Time?
Have Cancel Time?
Is paid?
Yes
Yes
Confirm order
Cancel order
add to order list
Remove from order list

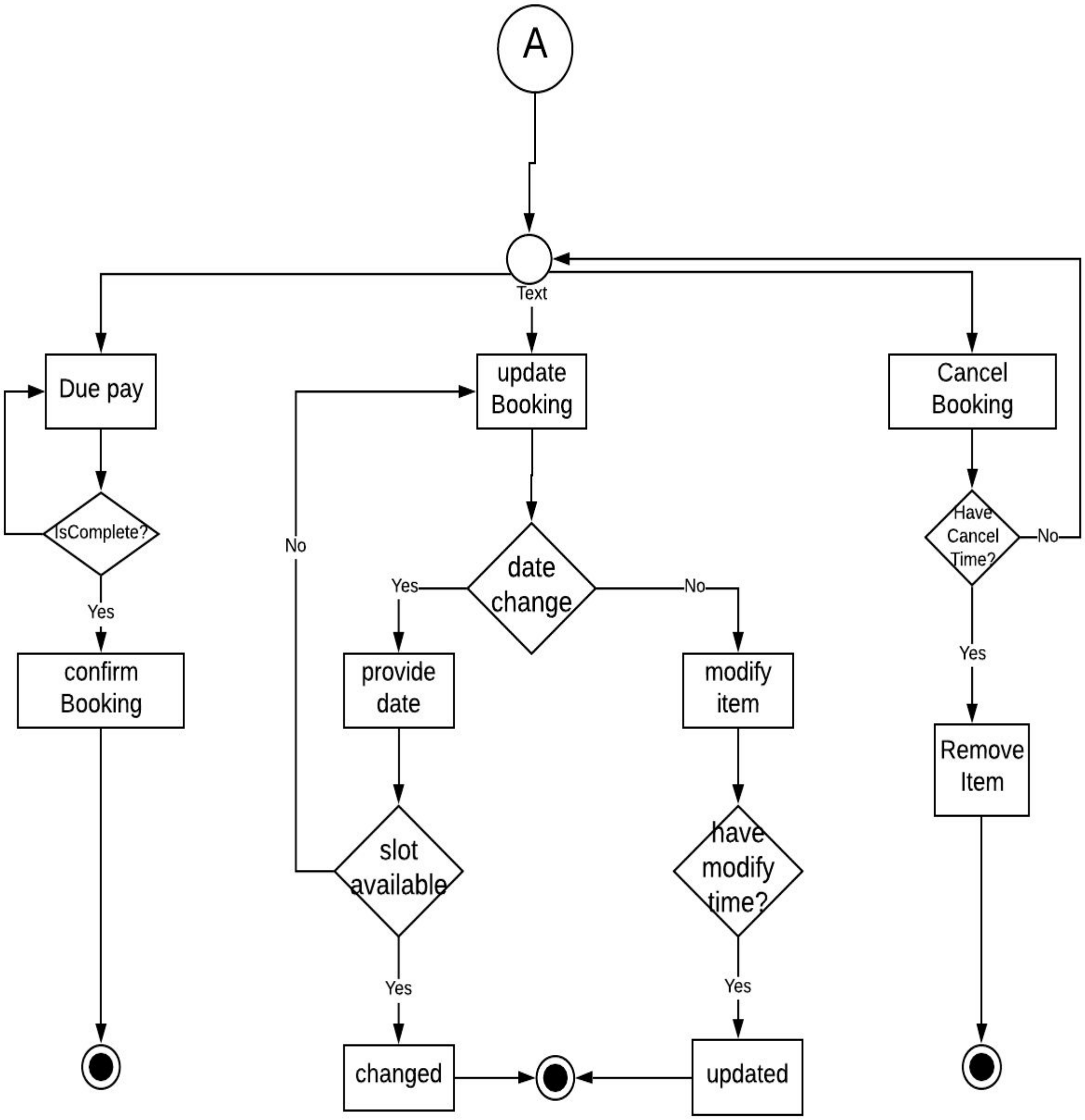
4.4.1.3.3 Service system



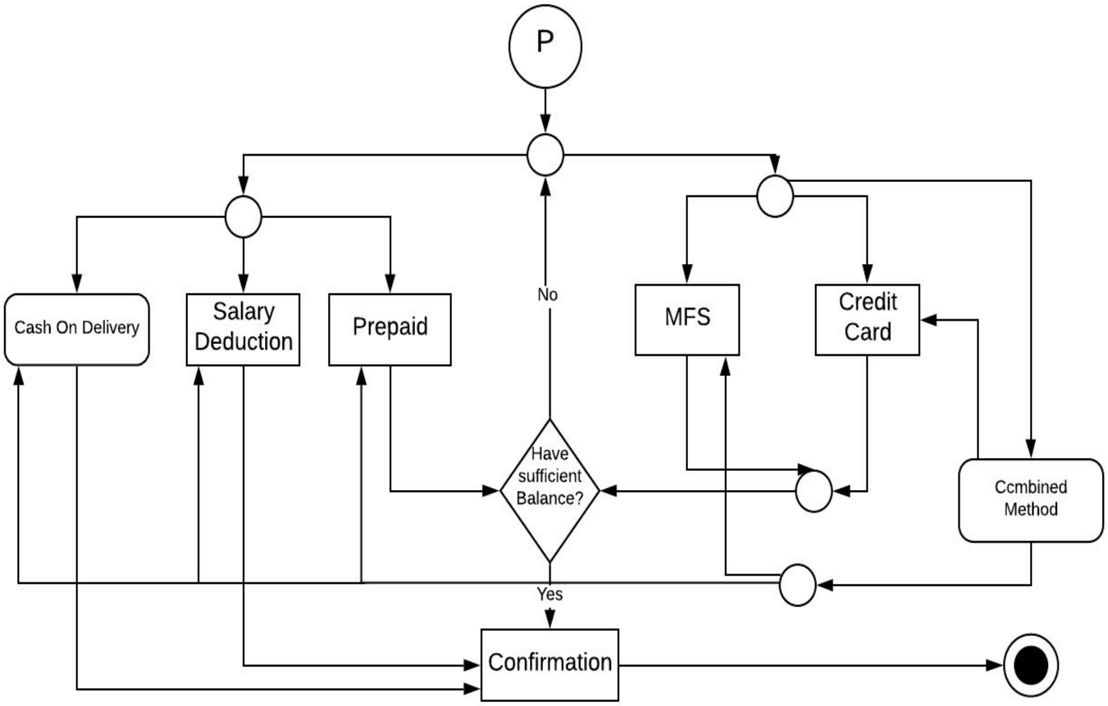
4.4.1.4.1 Initialize Booking



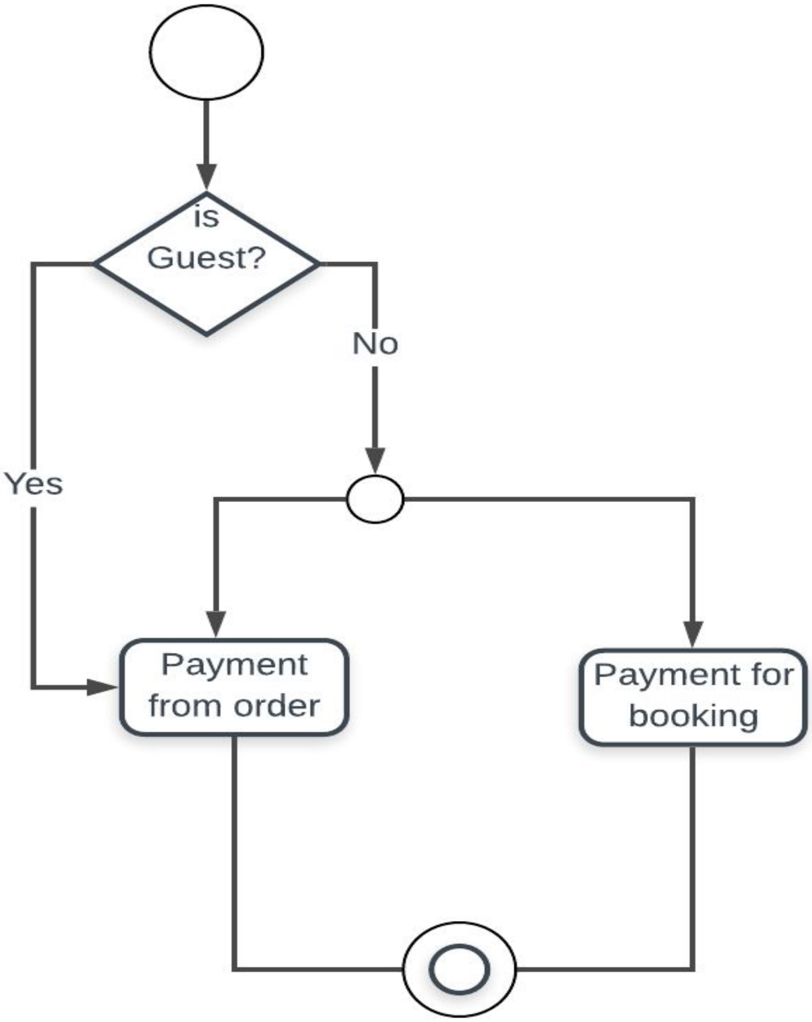
4.4.1.4.2 Modify Booking



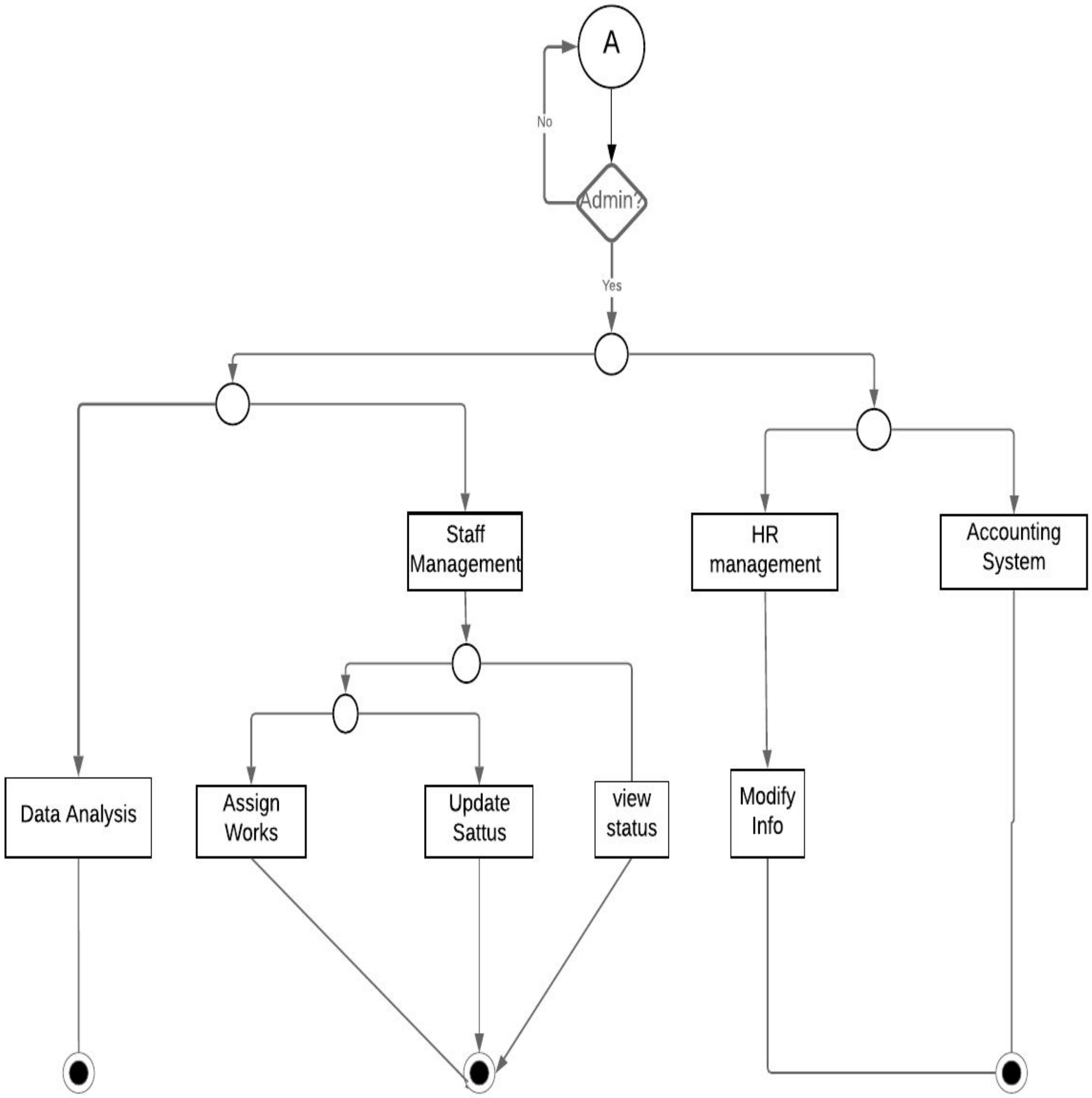
4.4.1.5.1 Payment method



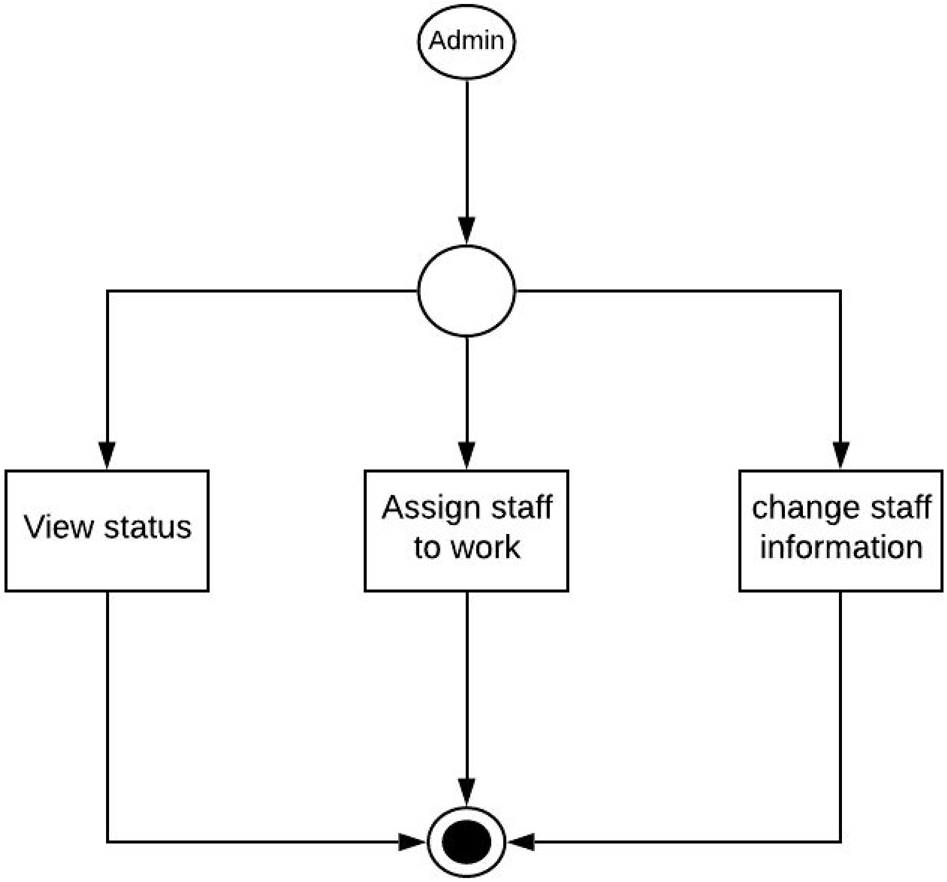
4.4.1.5.2 Payment type



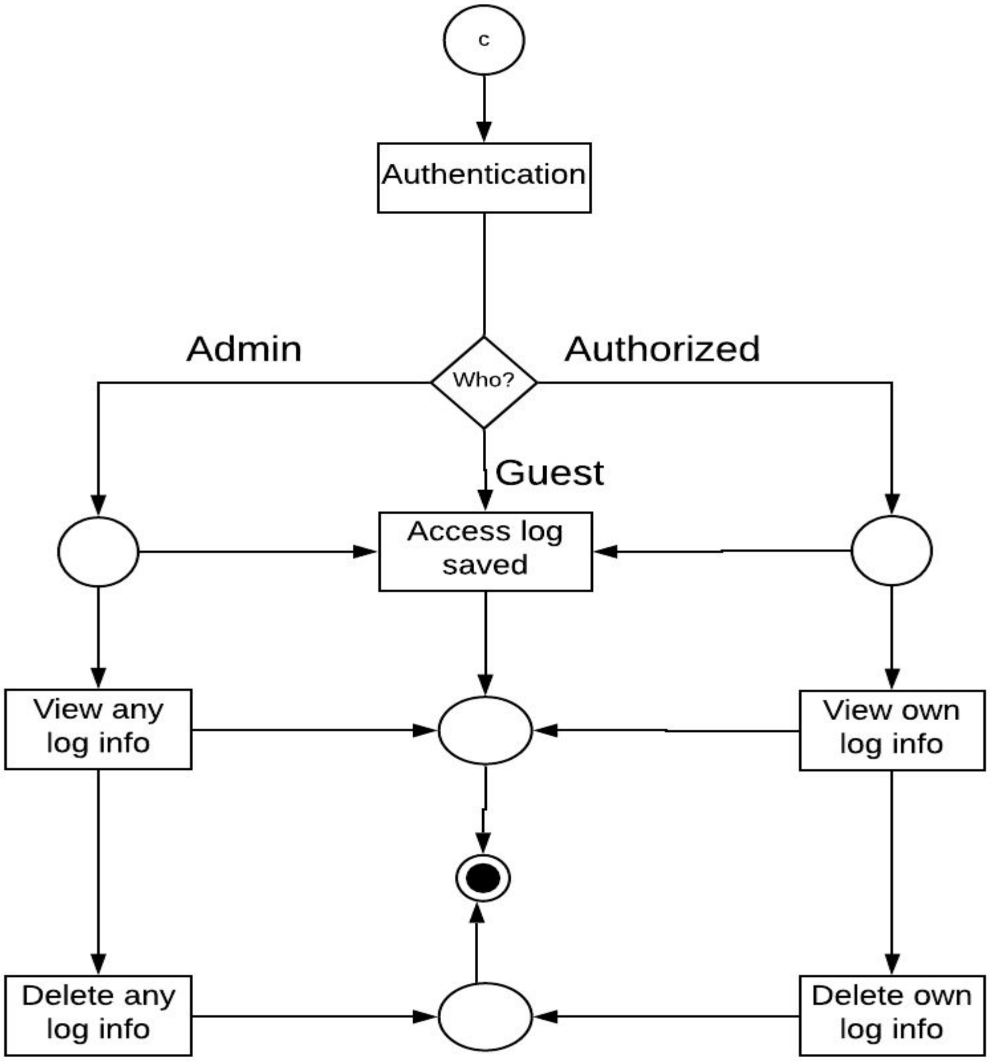
4.4.1.6 Information management



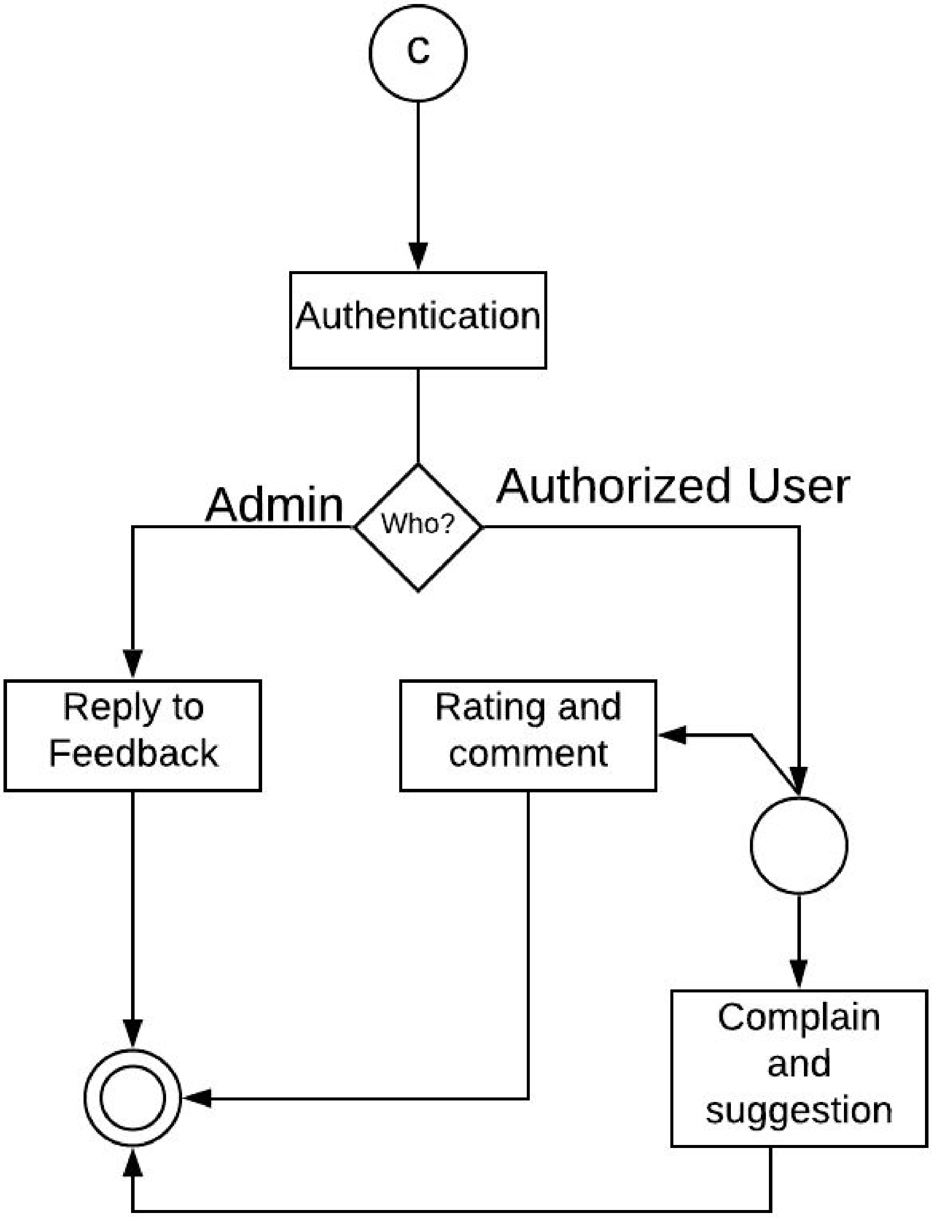
4.4.1.6.4 Staff management



4.4.1.7 Activity Log

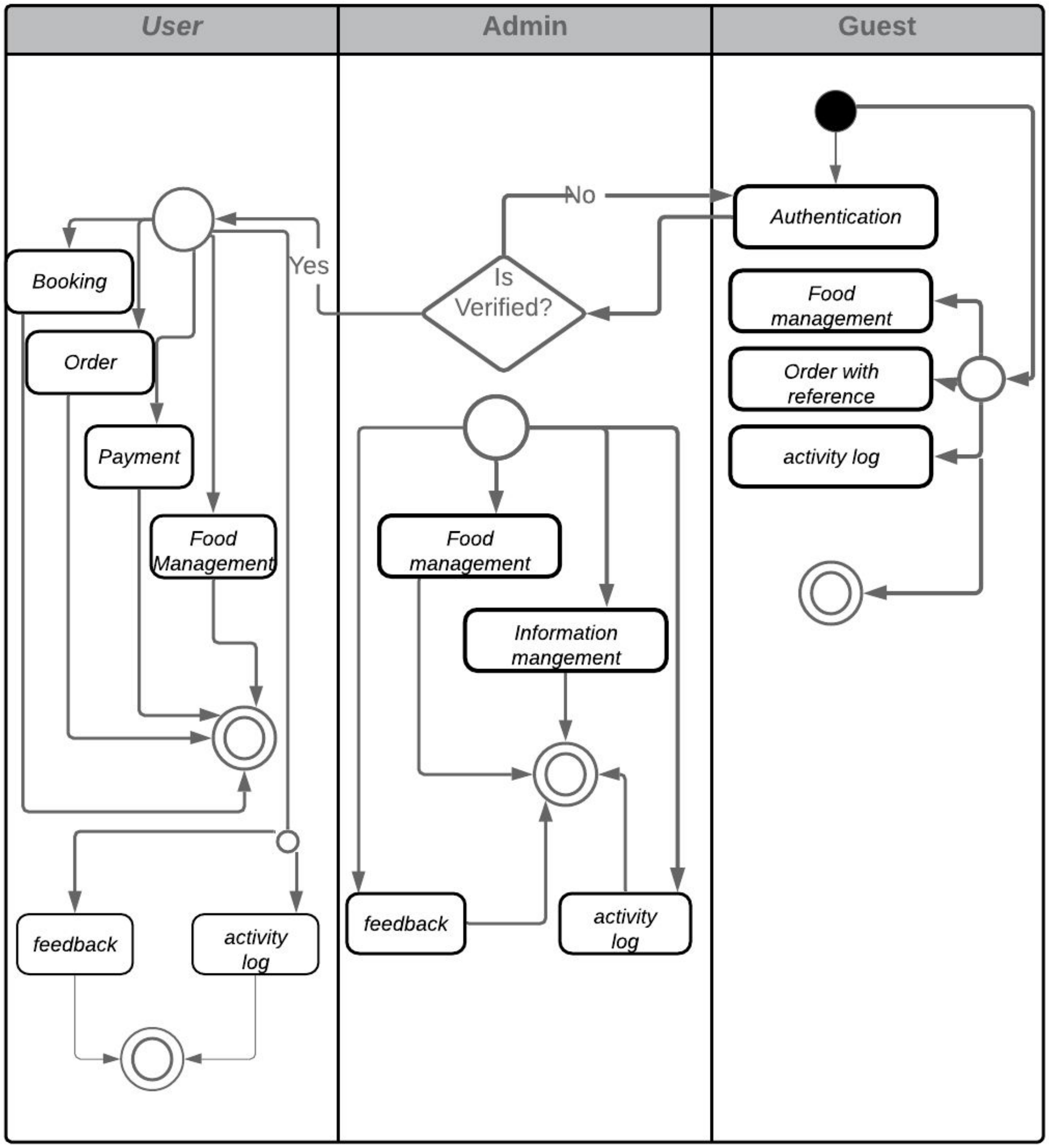


4.4.1.8 Feedback

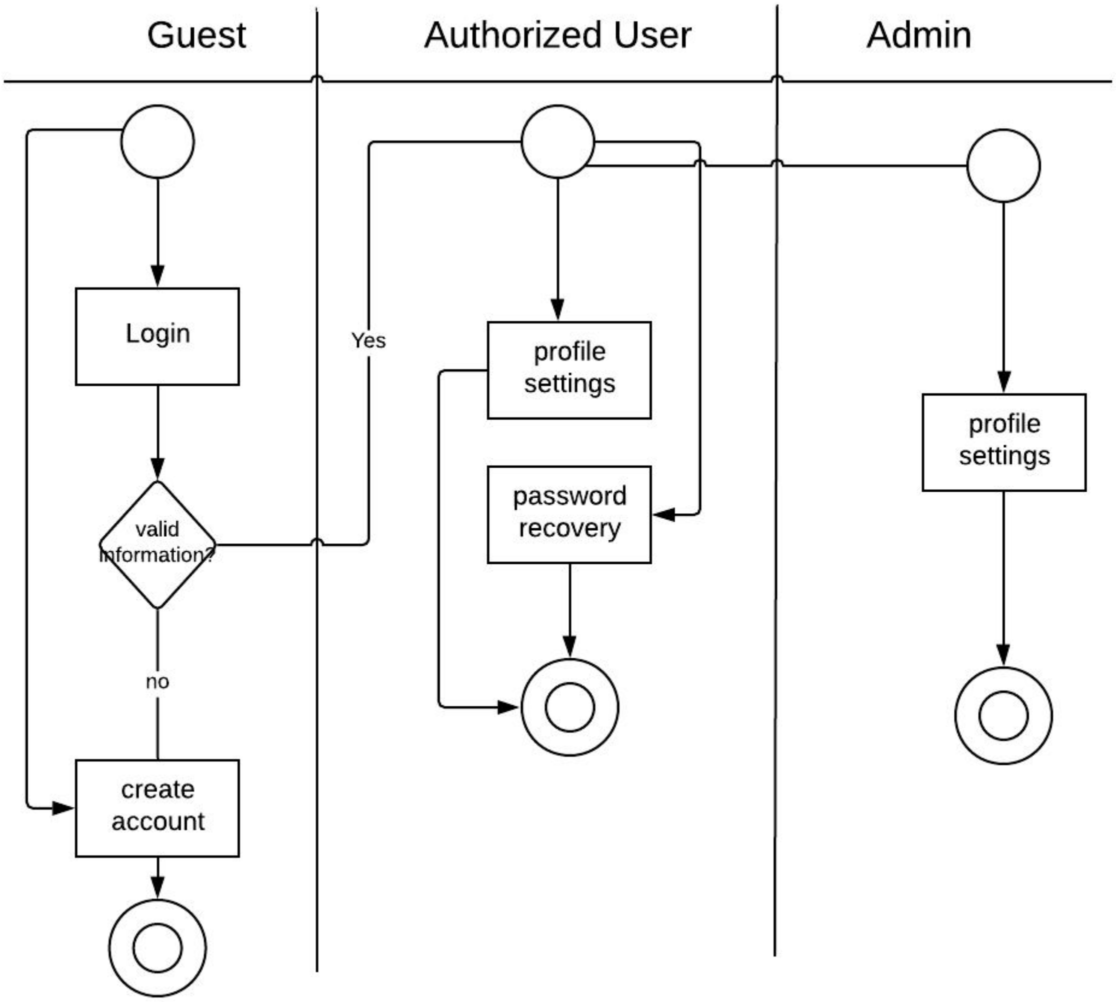


4.5 SWIMLANE DIAGRAMS OF GMS

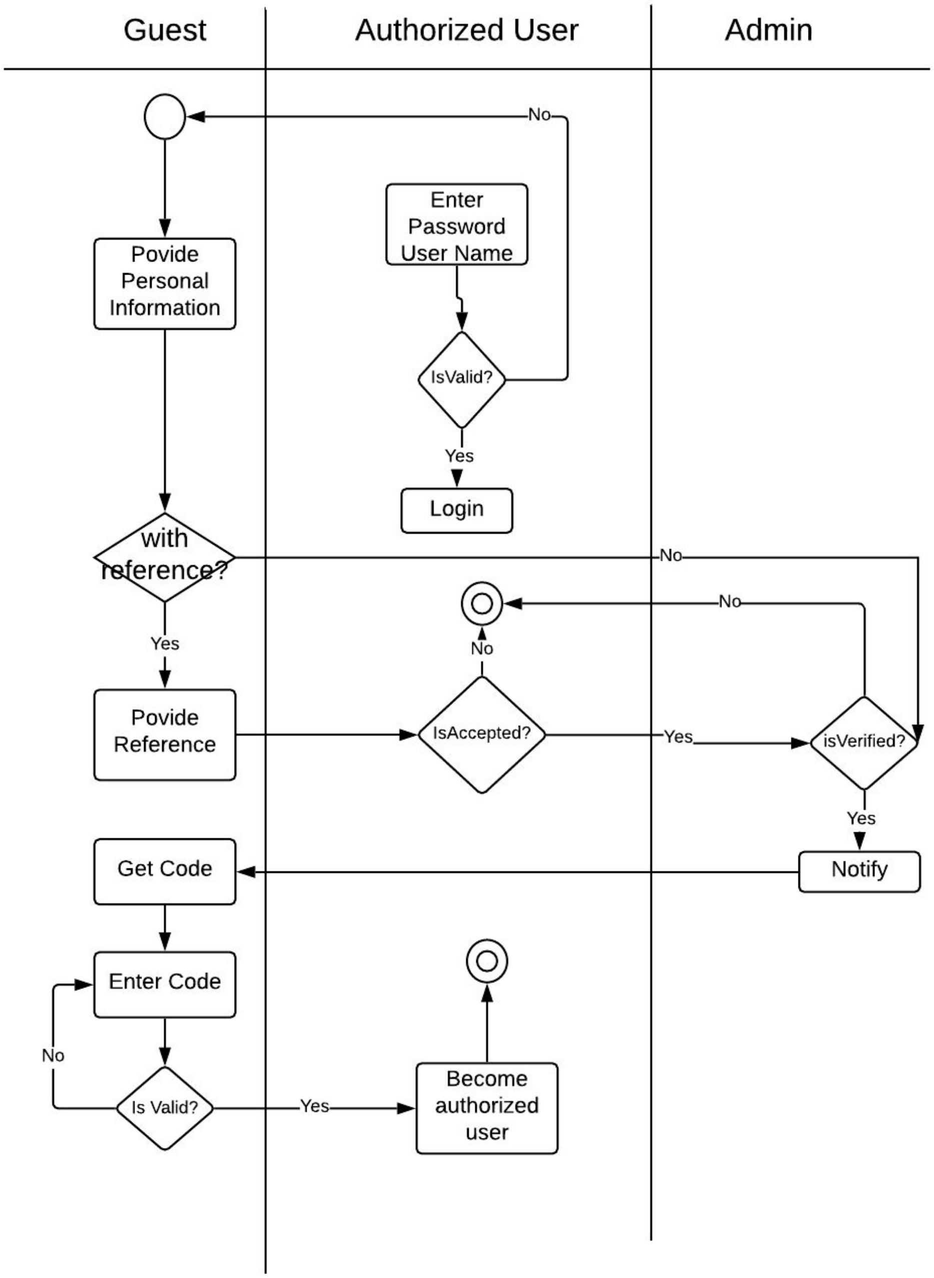
4.5.1 Cafeteria Management System



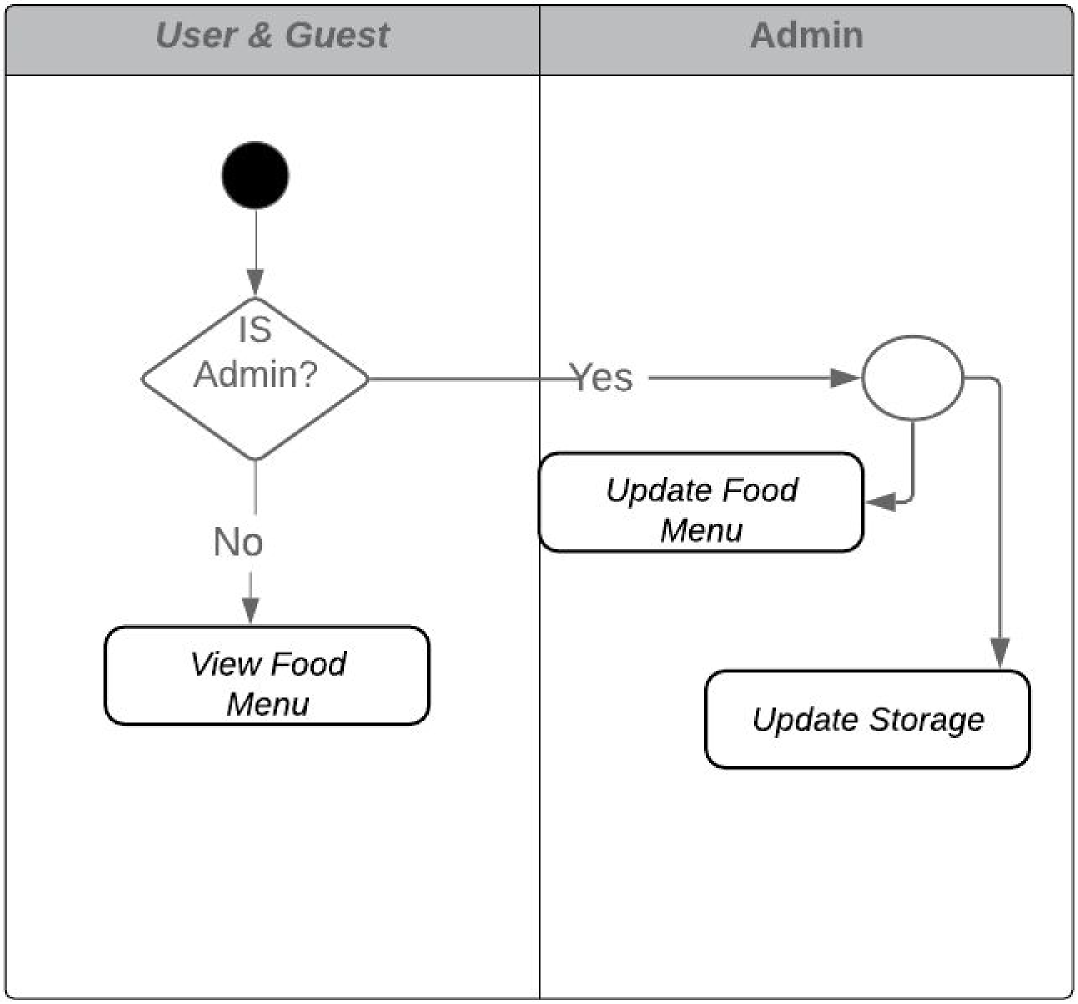
4.5.1.1 Authentication



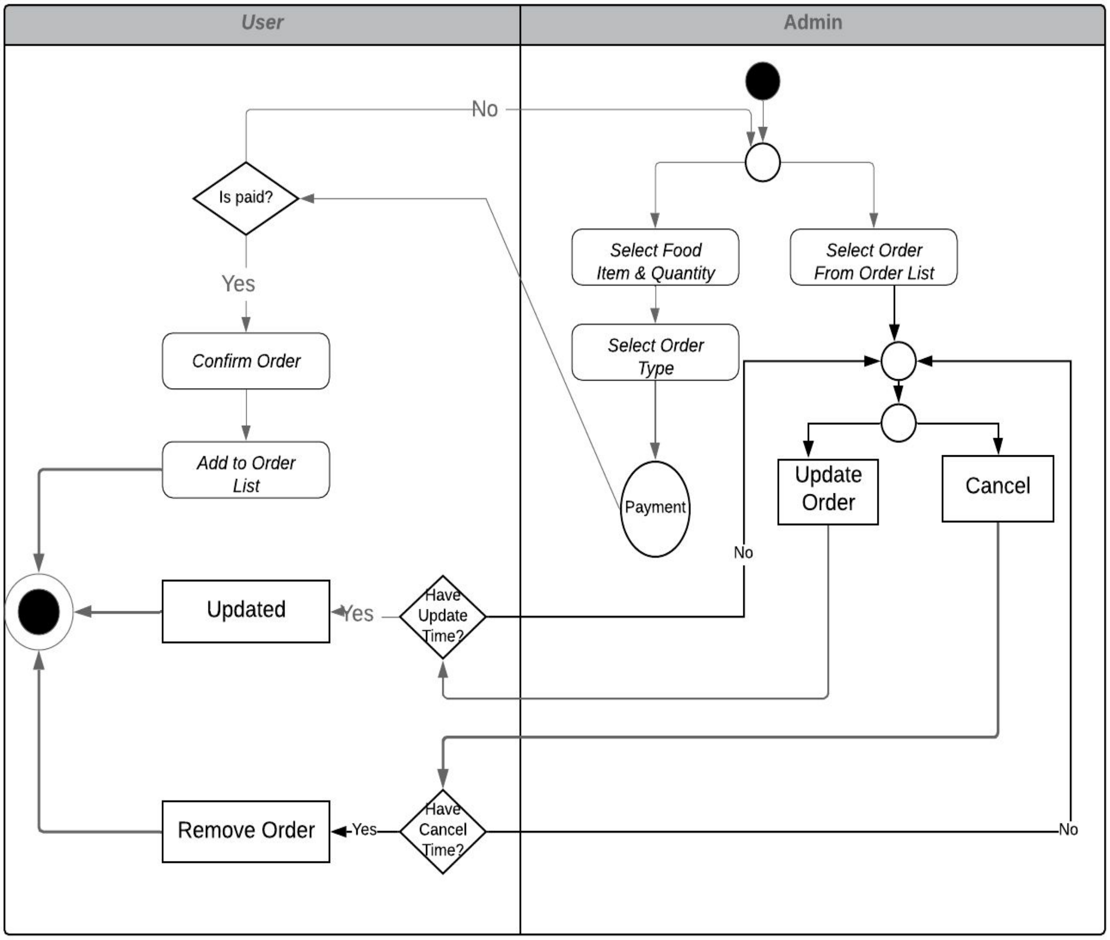
4.5.1.1.1 Create account



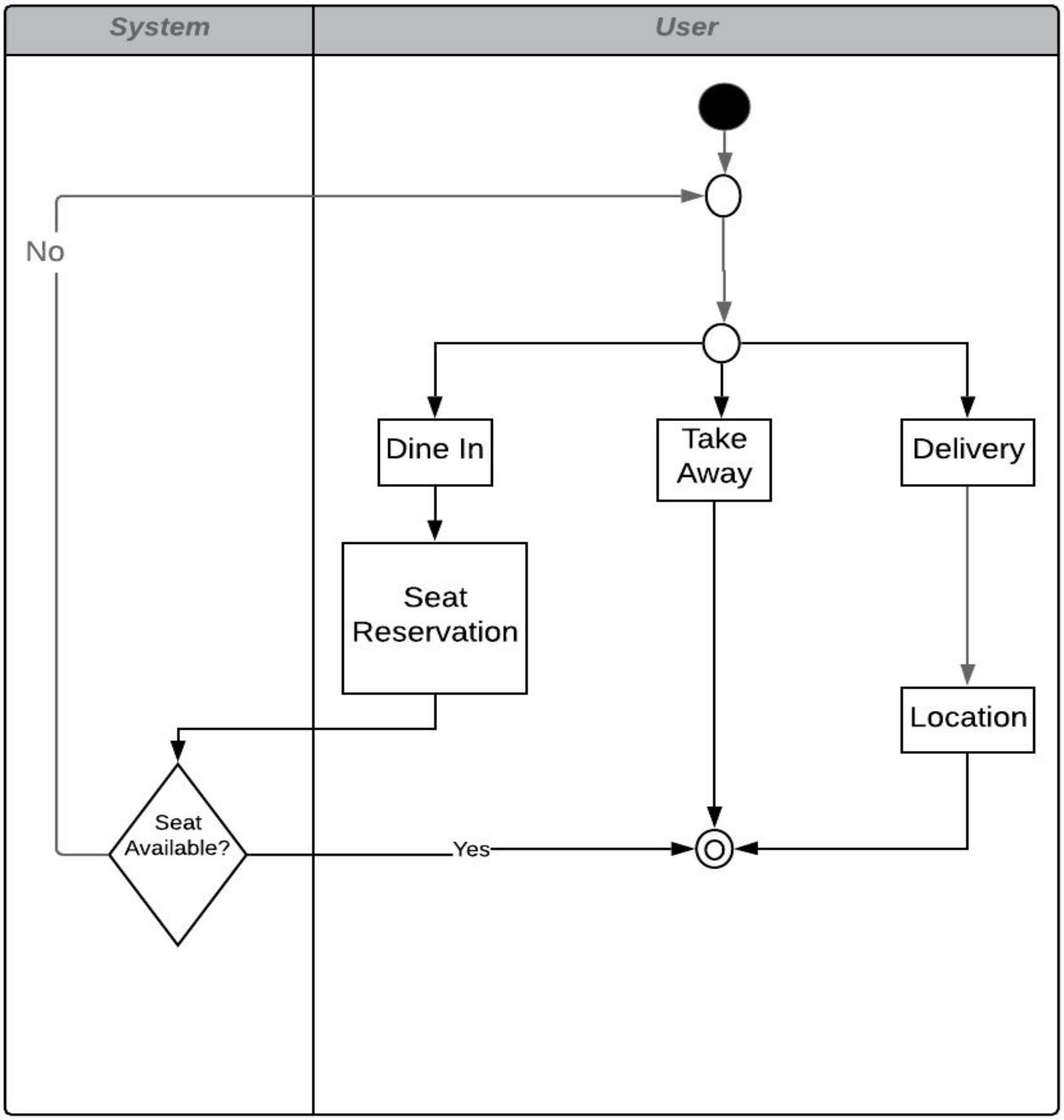
4.5.1.2 Food management



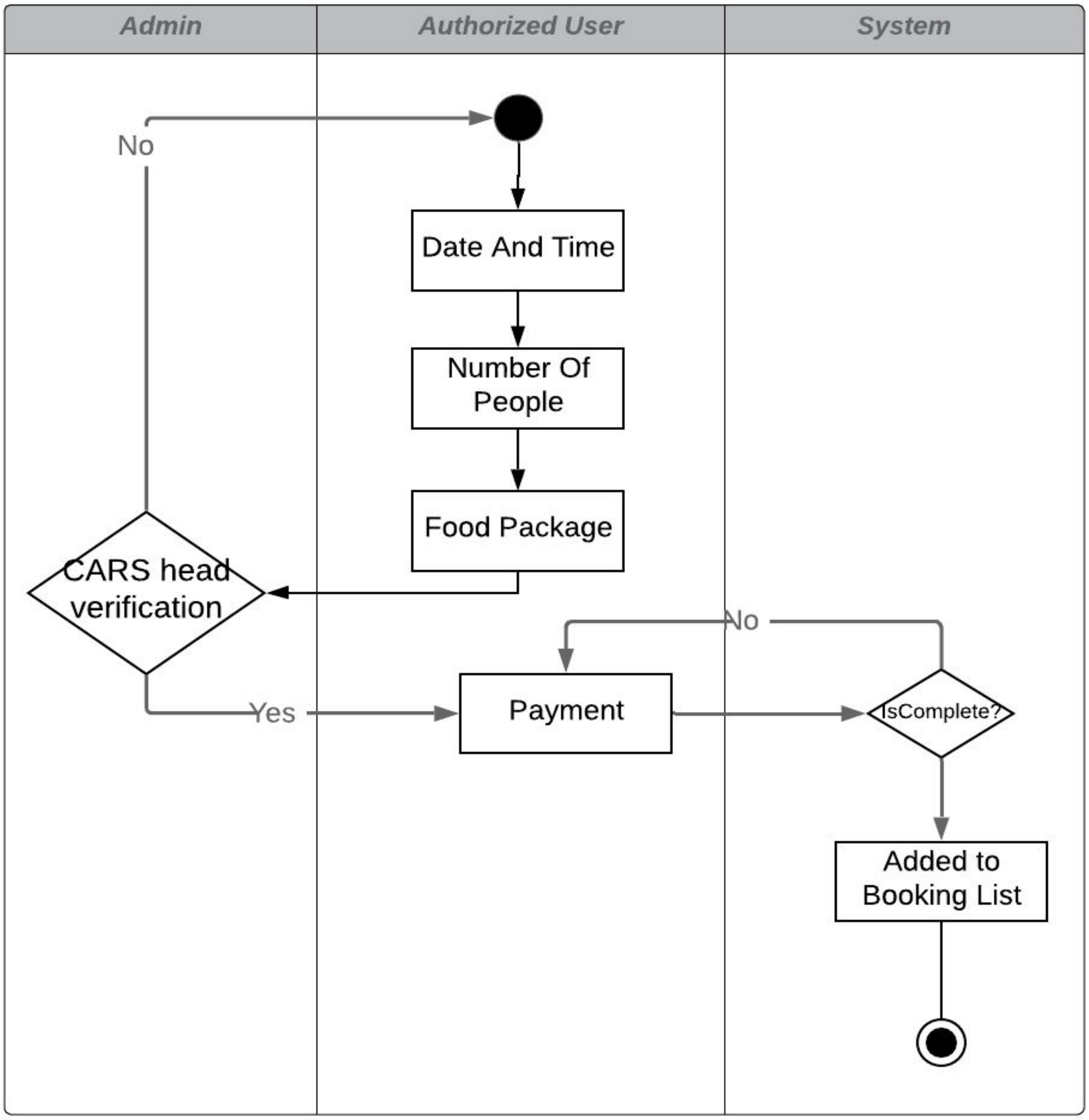
4.5.1.3 Order


User
Admin
No
Is paid?
Yes
Confirm Order
Add to Order List
Select Food Item & Quantity
Select Order From Order List
Select Order Type
Payment
Update Order
Cancel
No
Updated
Yes
Have Update Time?
Remove Order
Yes
Have Cancel Time?
No

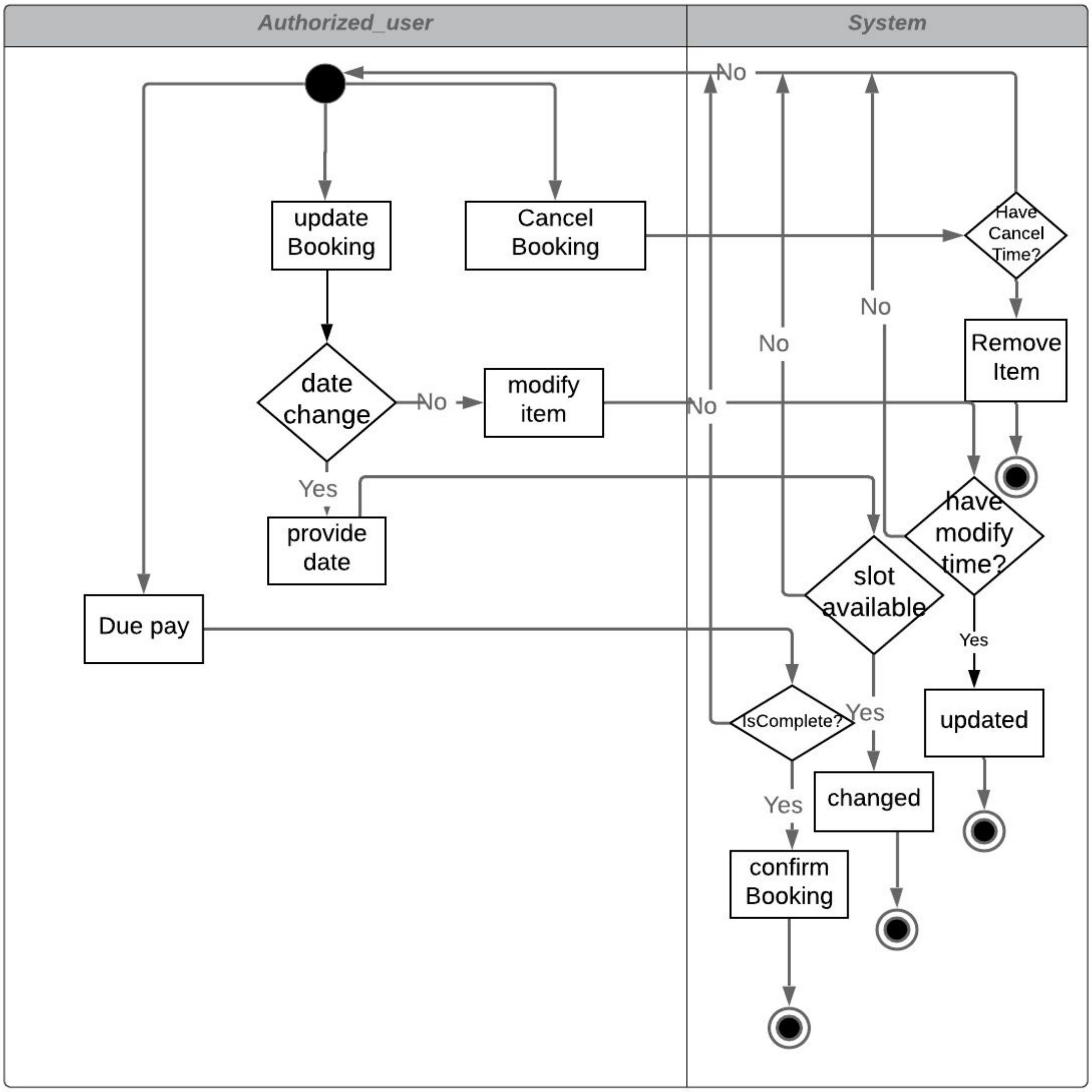
4.5.1.3.3 Service system



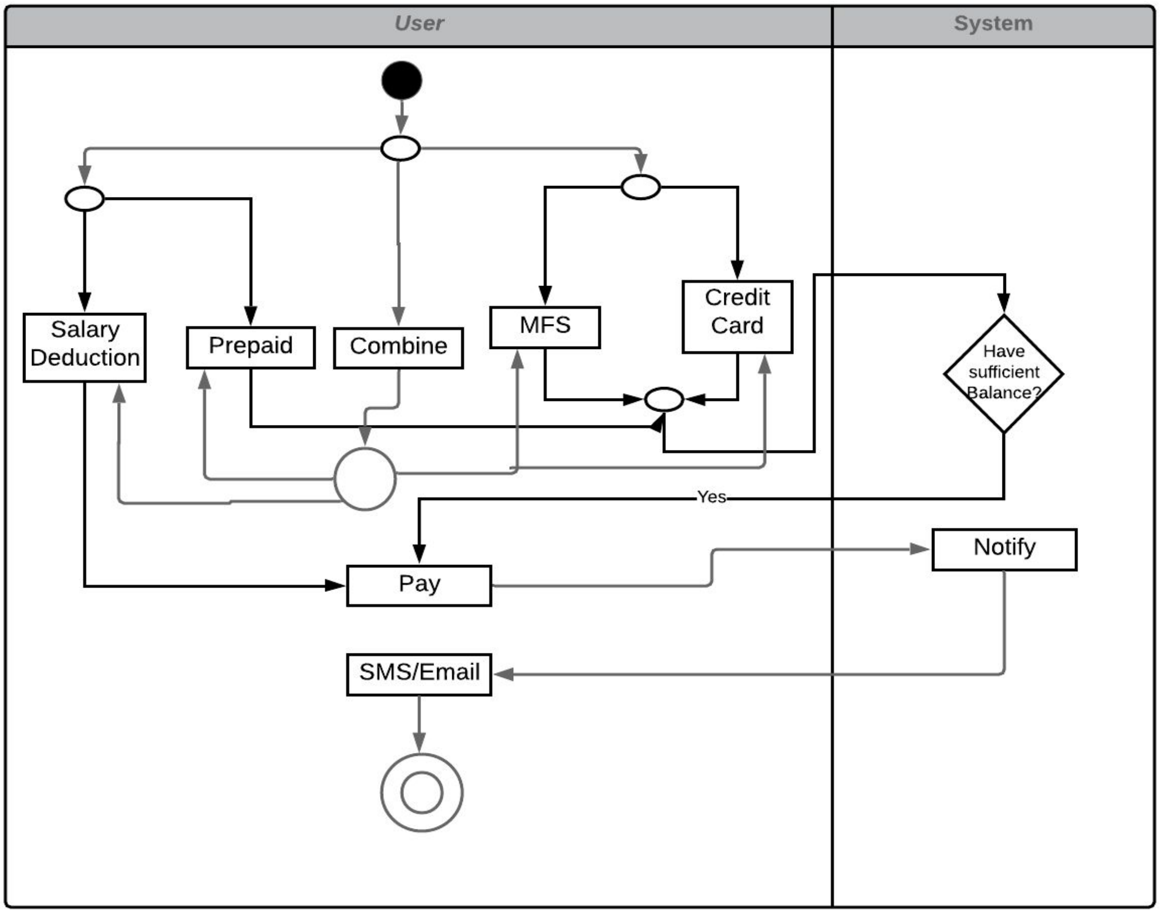
4.5.1.4.1 Initialize Booking



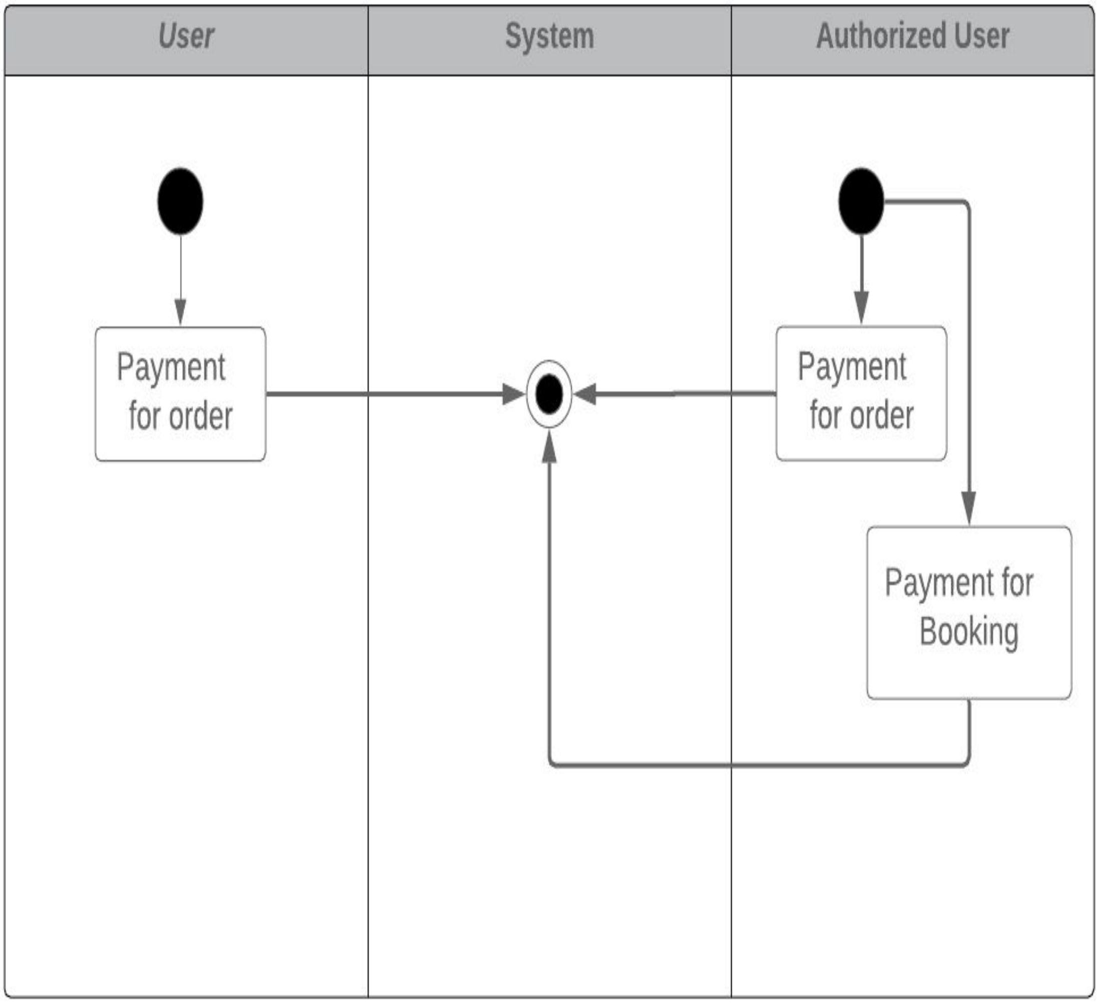
4.5.1.4.2 Modify Booking



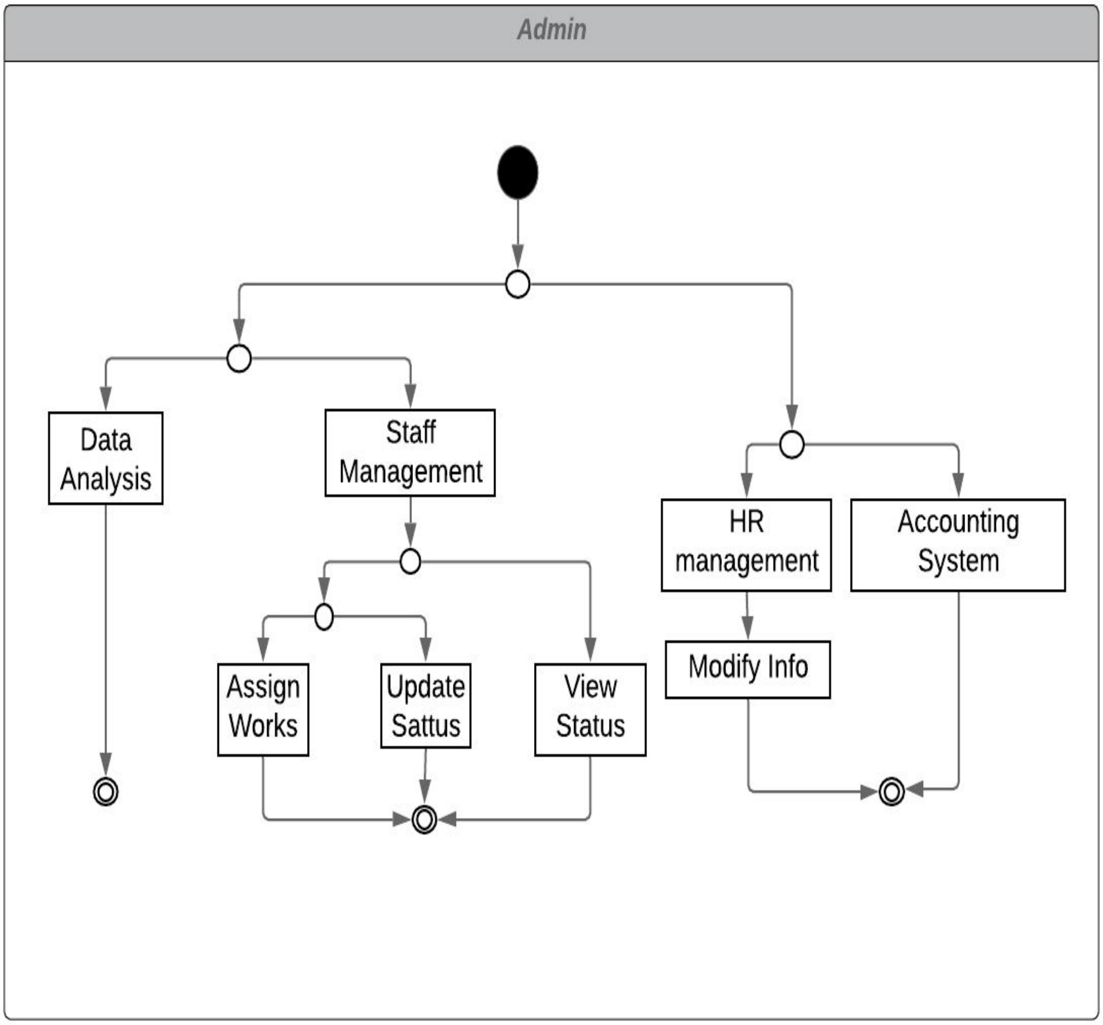
4.5.1.5.1 Payment method



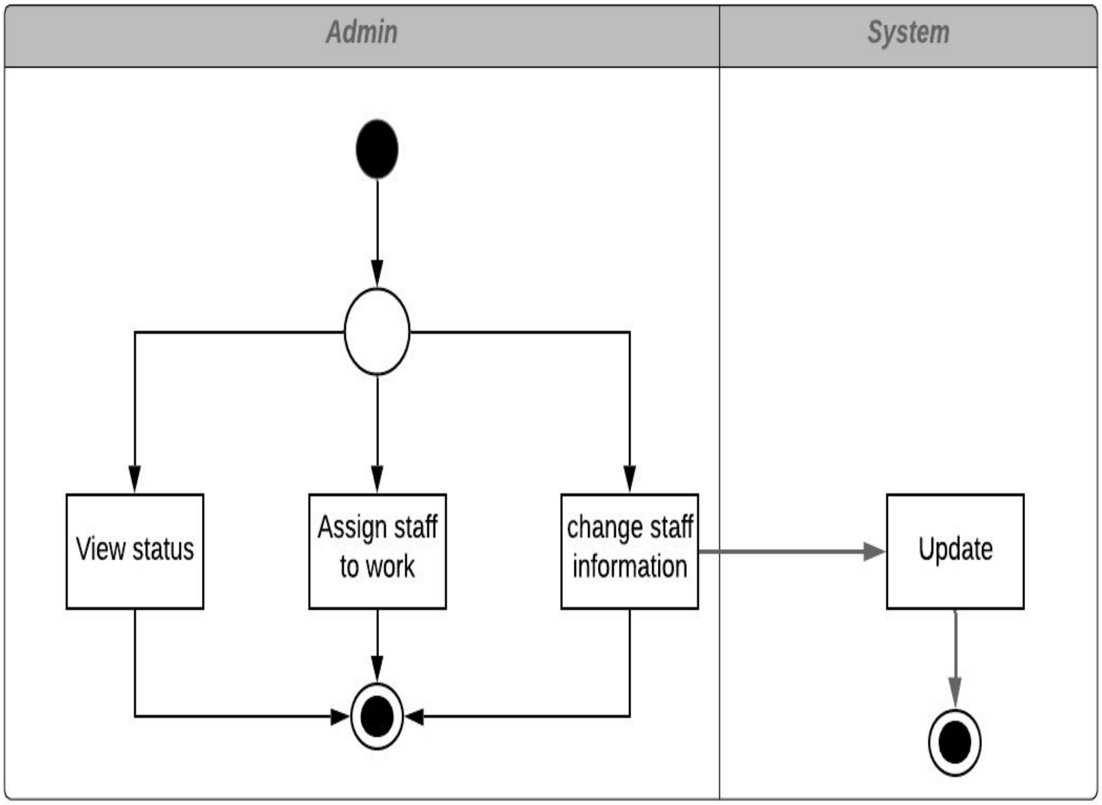
4.5.1.5.1 Payment type



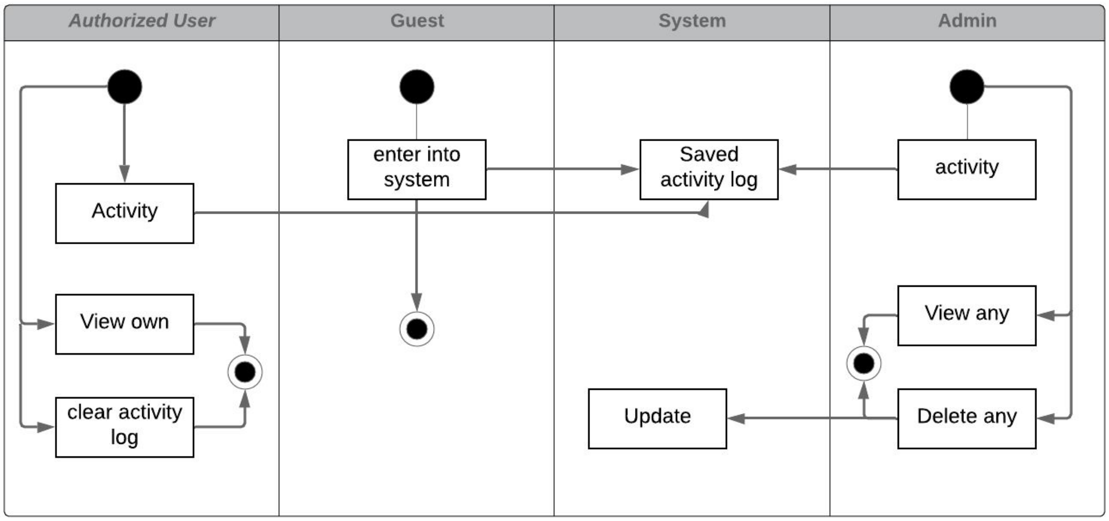
4.5.1.6 Information management



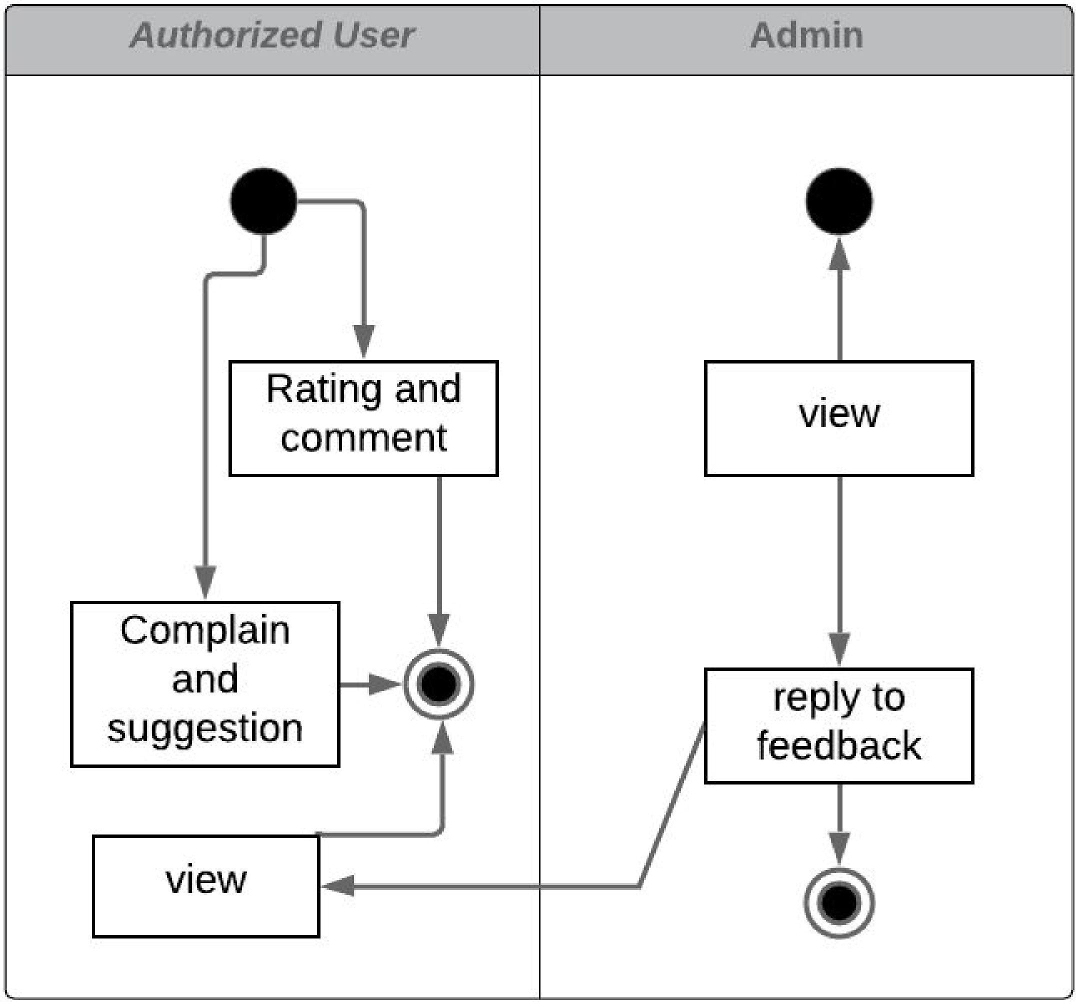
4.5.1.6.4 Staff management



4.5.1.7 Activity Log



4.5.1.8 Feedback



5 DATA MODELING OF CMS

5.1 DATA MODELING CONCEPT

If software requirements include the necessity to create, extend or interact with a database or complex data structures need to be constructed and manipulated, then the software team chooses to create data models as part of overall requirements modeling. The entity-relationship diagram (ERD) defines all data objects that are processed within the system, the relationships between the data objects and the information about how the data objects are entered, stored, transformed and produced within the system.

5.1.1 DATA OBJECTS

A data object is a representation of composite information that must be understood by the software. Here, composite information means an information that has a number of different properties or attributes. A data object can be an external entity, a thing, an occurrence, a role, an organizational unit, a place or a structure.

5.1.1.1 NOUN IDENTIFICATION

We extracted all nouns from story weather they are in the problem space or solution space.

Table 1: *Noun Identification*

SL	Noun	Attributes
1	User	12,22,21
2	Reference	
3	System	
4	Name	
5	Employee Id	
6	Designation	
7	Email Address	
8	Mobile Number	
9	Department name	
10	User Id	

11	Password	
12	Reference Email Id	
13	Teacher	4-11
14	Officer	4-11
15	Admin	4-11
16	User Information	4-11
17	Account Information	4-11
18	Authorized User	4-11
19	University Of Dhaka	13,14
20	Account	13,14,15
21	Ip Address	
22	Time Duration	
23	Guest	12,21,22
24	Price	
25	Quantity	
26	Food Item	24,25
27	Food Menu	26
28	Functionality	
29	Location	
30	Cost	
31	Delivery	29,30

32	Take Away	
33	Dine-In	
34	Order type	31-33
35	Order_id	
36	Order_Food_info	10,34,35
37	Program Name	
38	Time	
39	Date	
40	Booking Status	
41	Booking	37-40
42	Reference	10
43	Food Package	
44	Food Management	26,43
45	Quantity	
46	Stock Management	26,45
47	Stock Storage	,
48	Lunch	
49	Snacks	
50	Dine-In	
51	Take-Away	
52	Delivery	

53	Buffet	
54	Dinner	
55	Service	48-54
56	Accounting Software	
57	Refreshment Item	
58	Office	29
59	Income	
60	Expenditure	
61	Profit	
62	Loss	
63	Business Condition	
64	Future Plan	
65	Data Analytics	59-63
66	Due	
67	Bill	
68	Advance Money	
69	Payment For Booking	66-68
70	Payment For Order	
71	Payment Type	69-70
72	Personal Information	
73	Contact Information	72,73

74	Staff	
75	Email/SMS	15,18
76	Balance	
77	Pre-paid System	
78	Salary Reduction	
79	Digit	
80	Activity Log	
81	Salary	
82	Human Resource Management	74,81
83	Credit Card	
84	Cash-On	
85	Combined Method	
86	Mobile Financial Service	
87	Third Party Agent	
89	Payment Method	77,78,86,83,84,85,87
90	Seat	
91	Seat Reservation System	22,90
92	Authentication	
93	Method	
94	Domain	
95	Verification Request	

96	Verification Code	
97	Complaint	
98	Suggestion	
99	Complaint Hub	97,98
101	Comment	
102	Ratting	
103	Reply	
104	Feedback	99,101,102,103
105	Interface	
106	Staff Status	
107	Work Hour	
108	Staff Management	74,106,107
109	Database	17,44,74,80,82
110	Transaction ID	
111	Payment	71,89,110
112	Transaction	111
113	Information Management	65,82,108
114	Confirmation	
115	CARS Head	
116	Notification	
117	Advance	

118	Ordering	
119	Cafeteria Services	41,118
120	Booking System	
121	Delivery System	
122	Financial Report	65

5.1.1.2 ***Potential Data Objects:***

1. User : 12, 21, 22
2. Teacher: 4-11
3. Officer : 4-11
4. Admin : 4 - 10
5. User Information : 4 - 11
6. Account Information : 4-11
7. Authorized user : 4-11
8. University of Dhaka : 13, 14
9. Guest : 12, 21, 22
10. Food Menu : 26
11. Food item : 24, 25
12. Delivery:29, 30
13. Order Type : 31-33
14. Order_food_info : 10, 34, 35
15. Booking : 37-40

16. Food management: 26, 43
17. Service : 48-54
18. Data Analytics : 59-63
19. Payment for booking : 66-68
20. Human Resource management : 74,81
21. Payment method : 77,78,86,83,84,85,87
22. Seat Reservation System : 22, 90
23. Complaint hub : 97,98
24. Staff management: 74, 106,107
25. Database:17,44,74, 80,82
26. Payment: 81, 89, 110
27. Transaction: 111
28. Information management: 65, 82, 108
29. Cafeteria Services:41, 118
30. Feedback : 99,101,102,103

5.1.1.3 Analyzing and Finalizing DataObject:

1. Teacher, Officer, Admin, Authorize User and Account Information all to have almost the same attributes so we have taken only Teacher, Officer, and Admin by merging all attributes. We have taken three data object rather only one authorized user because those are our main actor. So we need to identify them separately.

2. Activity log conveys all attributes of User and Guest User along with activities, ip, and timestamp. So we discarded User and Guest User.

3. We merged Food_menu, Food_item, and Food_management into Food as those have few attributes.

4. Order_type and Order_info have been merged into Order by following same manner in 3.

5. Booking contains only attributes of itself. It merges no other potential data object as it has sufficient attributes for making a separate data object.

6. Transactions store information related to all payment along with Order and Booking information.

7. Human resource management and Staff information merged into Staff as they convey little bit different attributes. So they are more relevant data object and we can merge them into one .

8. Feedback and Complain Hub merged into Feedback because of similarities.

9. Delivery stores available delivery location with corresponding cost. It is considered as a simple lookup data object for calculating total cost while delivering a food item in a location.

10. Account provides quick access to username and password for login purpose. We know Username and password is the most frequently used data in our system. So for performing quick access to database it should not be stored in such a data object that conveys a lot of attributes

11. Analytics holds all financial information of each days expenditure, losses, profit so that we can calculate yearly financial information also.

5.1.1.4 Final Data Objects:

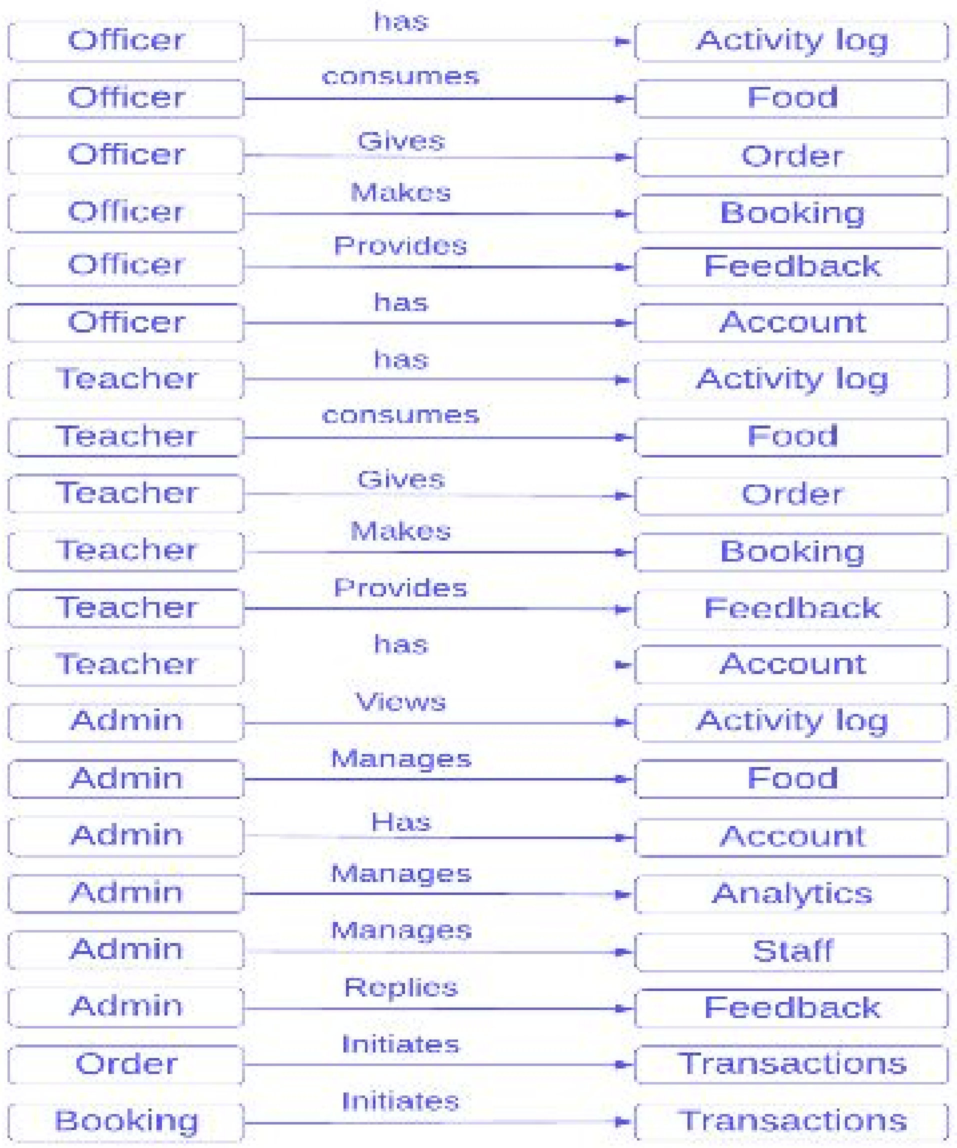
Table 2: schema

1	**Teacher (**Name, T_id, Employee_id, Email Address, Mobile Numbe Department Name, Designation, Reference_id, balance**)**
2	**Officer (**Name, O_id, Employee_id, Email Address, Mobile Number Department Name, Designation, Reference_id, balance**)**

3	**Admin (**Name, A_id, Email Address, Mobile Number **)**
4	**Activity log (**U_id, Ip_Address, Time Stamp, Description**)**
5	**Food (**Item Name, Food_id , Food Type, Price, Quantity, Rating, Comment, Package_id, Day **)**
6	**Order (**Order_id, U_id, Order_type, Food_id, Quantity, Number Of Seat, Total_Cost, Location, Due, Time**)**
7	**Booking (**Booking_id, U_id , Package_id, Service_Type, Confirmation_date, Service_Date, Total_Cost, Due, Number Of Seats
8	**Transaction (** Payment Method, Order_id , Booking_id, Transaction_ U_id, Amount, Date, Time**)**
9	**Staff (**Staff_name, S_id, Position, Salary, Working_Hour, Contact N Address, Status**)**
10	**Feedback (** U_id , F_id , FeedBack_type , Feedback , Food_id , B_ic Date , Time**)**
11	**Delivery (**Location, Cost**)**
12	**Analytics (**Date, Income, Expenditure, Profit, Loss, Account_Payabl Account_receivable**)**
13	**Account (**U_id, User Name, Password**)**

5.2 DATA OBJECT RELATIONSHIPS:

Data objects are connected to one another in different ways.



Officer	has	Activity log
Officer	consumes	Food
Officer	Gives	Order
Officer	Makes	Booking
Officer	Provides	Feedback
Officer	has	Account
Teacher	has	Activity log
Teacher	consumes	Food
Teacher	Gives	Order
Teacher	Makes	Booking
Teacher	Provides	Feedback
Teacher	has	Account
Admin	Views	Activity log
Admin	Manages	Food
Admin	Has	Account
Admin	Manages	Analytics
Admin	Manages	Staff
Admin	Replies	Feedback
Order	Initiates	Transactions
Booking	Initiates	Transactions

5.3 Schema Diagram

Table 3: Schema for Teacher

Teacher		
Attributes	Type	size
Name	Varchar2	80
T_id	Varchar2	20
Employee_id	Varchar2	20
Email Address	Varchar2	30
Mobile Number	Varchar2	15
Department Name	Varchar2	30
Designation	Varchar2	20
Reference_id	Varchar2	20
balance	number	10

Table 4: Schema for Officer

Officer		
Attributes	Type	size
Name	Varchar2	80
O_id	Varchar2	20

Employee_id	Varchar2	20
Email Address	Varchar2	30
Mobile Number	Varchar2	15
Department Name	Varchar2	30
Designation	Varchar2	20
Reference_id	Varchar2	20
balance	number	10

Table 5: Schema for Admin

Admin		
Attributes	Type	size
Name	Varchar2	80
A_id	Varchar2	20
Email Address	Varchar2	30
Mobile Number	Varchar2	15
Reference_id	Varchar2	30

Table 6: Schema for Activity log

Activity log		
Attributes	Type	size
U_id	Varchar2	20
ip_Address	Varchar2	20
Timestamp	Varchar2	20
description	Varchar2	20
		20
		200

Table 7: Schema for Account

Account		
Attributes	Type	size
U_id	Varchar2	20
User name	Varchar2	30
password	Varchar2	20

Table 8: Schema for Booking

Booking		
Attributes	Type	size
Booking_id	Varchar2	20
U_id	Varchar2	20
package_id	Varchar2	20
Service type	varchar2	20
confirmation_date	varchar2	20
service_date	Varchar2	20
total_cost	number	10
Due	number	10
Number of seats	number	10

Table 9: Schema for Order

Order		
Attributes	Type	size
order_id	Varchar2	20
U_id	Varchar2	20
Order type	Varchar2	20

food_id	varchar2	10
quantity	number	10
Number of seats	number	10
total_cost	number	10
location	Varchar2	10
Due	number	10
time	varchar2	10

Table 10: Schema for Transaction

Transaction		
Attributes	Type	size
Transaction_id	Varchar2	20
U_id	Varchar2	20
Order_id	Varchar2	20
booking_id	varchar2	20
amount	number	10
date	varchar2	20
time	varchar2	20
Payment method	Varchar2	20

Table 11: Schema for Staff

Staff		
Attributes	Type	size
Staff_name	Varchar2	30
S_id	Varchar2	20
Position	Varchar2	30
salary	number	10
working_hour	varchar2	10
contact_no	varchar2	20
address	varchar2	20
Status	Varchar2	20

Table 12: Schema for Feedback

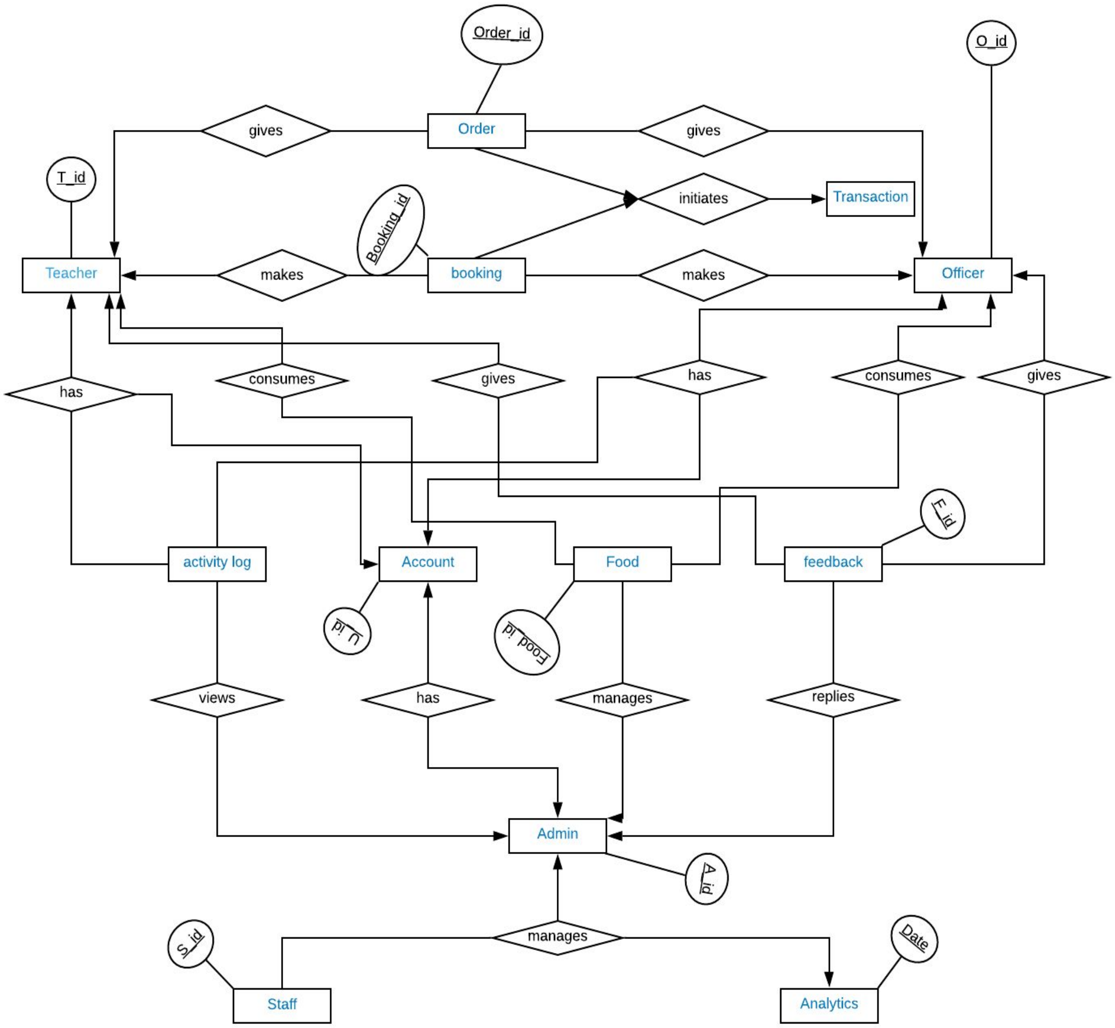
Feedback		
Attributes	Type	size
U_id	Varchar2	30

F_id	Varchar2	30
Feedback_type	Varchar2	30
Feedback	varchar2	20
food_id	varchar2	20
booking_id	varchar2	20
date	varchar2	20
time	Varchar2	20

Table 13: Schema for Data_Analytics

Data_Analytics		
Attributes	Type	size
Date	Varchar2	20
Income	number	10
Expenditure	number	10
Profit	number	10
Loss	number	10
Accounts_payable	number	10
Accounts_receivable	number	10

5.4 Entity Relation Diagram



Class based modeling of CMS

This section is intended to describe class based modeling of CMS.

5.1 Class based modeling concept

Class-based modeling represents the objects that the system will manipulate, that will the operations applied to the objects, relationships between objects and the collaborations that occur between the classes that are defined.

6.2 GENERAL CLASSIFICATION

To identify the potential classes, we have first selected the nouns from the
solution space of the story. These were then characterized in seven general
classification. The seven general characteristics are as follows

General Classification:

1.External entities

2.Things

3.Occurrences/Event

4.Roles

5.Organizational unit

6.Places

7.Structures

Table 14: General Classification

No	Noun list	General Classification
1.	Cafeteria management system	7
2.	Teacher	4,5,7
3.	Officer	4,5,7
4.	University of Dhaka	6
5.	Account	5,7
6.	Authorized user	4,7
7.	Guest user	1,4,7

8.	Food menu	2
9.	Order food	3
10.	Authentication	3
11.	Create an account	3
12.	User	4,5,7
13.	Name	2
14.	Employee id	2
15.	Designation	2
16.	E-mail address/SMS	2
17.	Mobile number	2
18.	Department name	2
19.	Location	6
20.	User id	2
21.	Password	2
22.	Reference mail	2
23.	Department	2
24.	Admin	4,7
25.	Verification	3
26.	Request	3
27.	Verification code of 6 digit	2
28.	Change information	3

29.	Account information	2
30.	Verification process	3
31.	Cafeteria	6
32.	Lunch	3
33.	Snacks	3
34.	Storage	2
35.	Log in	3
36.	Service system	3
37.	Dine-in	3
38.	Take away	3
39.	Delivery	3
40.	Payment system	3
41.	Confirmation	3
42.	Seat reservation system	3
43.	Time duration	2
44.	Home	6
45.	Office	6
46.	Notification	3
47.	Dinner	3
48.	Party lunch	3
49.	Buffet	3

50.	Program	3
51.	Time slot	2
52.	Date & Time	2
53.	Booking	3
54.	Car’s head	4
55.	Timestamp	2
56.	Data analysis	2
57.	Payment method	3
58.	Bill	2
59.	Salary deduction	3
60.	Record	2
61.	Staff	4,7
62.	MFS(Mobile financial services)	3
63.	Credit card	3
64.	Transaction information	2
65.	Cash on	3
66.	Accounting software	3
67.	Human resource management	2, 4, 7
68.	Order	3, 7
69.	System	4,7
70.	Interface	4,7

71.	Database management	1,7
72.	Financial report	2,3
73.	Comment	2
74.	Feedback	3
75.	Rating	2
76.	Comment	2
77.	Complaint Hub	2
78.	Reply	2

6.3 Selection Criteria:

1. Retained Information
2. Needed Services
3. Multiple Attributes
4. Common Attributes
5. Common Operation
6. Essential Requirement

Table 15: General Classification

No	Noun list	Selection Criteri
1.	Teacher	1,2,3,4,5
2.	Officer	1,2,3,4,5
3.	University of Dhaka	6
4.	Account	1,2, 3, 4, 5
5.	Authorized user	1, 2, 3, 4, 5
6.	Guest user	1, 2, 3, 4, 5, 6
7.	Food menu	1, 2
8.	Authentication	6
9.	Create an account	1
10.	User	1, 2, 3, 4, 5
11.	Admin	1, 2, 3
12.	Storage	1,2
13.	Data analysis	1, 2
14.	Staff	1, 2, 3, 4, 5
15.	Order	1, 2, 3, 4, 5
16.	System	6
17.	Interface	6
18.	Database management	1,2

19.	**Financial report**	**1,2**
20	**HR management**	**1,2,3**

6.4 Class attributes and methods identification:

We now identify the attributes and methods associated with the potential classes to better find out the attributes and methods of each class.

Table 16: Associate noun and identification

Class name	Attributes	Methods

Teacher	name,
	employee id, designation
	E-mail address,
	mobile number, departmer
	name, location,
	user id,
	password,
	reference email id
	balance
		signIn()
		signOut()
		changeInfo()
		setPassword()
		viewMenu()
		orderFood()
		modifyOrder()
		cancelOrder()
		initializeBooking()
		confirmBooking()
		modifyBooking()

		cancelBooking()
		payingBill()
		verifyAnApplicant()
		giveAFeedback()
		giveRating&Comment()
		getRating&Comment()
		setBalance()
		getBalance()
		getRating()

Officer	name,
	employee id, designation
	E-mail address,
	mobile number, departme
	name, location,
	user id,
	password,
	reference email id
	balance
		signIn()
		signOut()
		changeInfo()
		setPassword()
		viewMenu()
		orderFood()
		modifyOrder()
		cancelOrder()
		initializeBooking()
		confirmBooking()
		modifyBooking()
		cancelBooking()
		payingBill()
		verifyAnApplicant()
		giveAFeedback()

		giveRating&Comment()
		getRating&Comment()
		setBalance()
		getBalance()

Account	name,
	employee id, designation
	E-mail address,
	mobile number, departme
	name, location,
	user id,
	password,
	reference email id
	balance
		signUp()
		signIn()
		changeInfo()
		setPassword()
		viewMenu()
		orderFood()
		modifyOrder()
		cancelOrder()
		initializeBooking()
		confirmBooking()
		modifyBooking()
		cancelBooking()
		payingBill()
		verifyAnApplicant()
		getBalance()
		setBalance()

Authorized use	name,
	employee id, designation
	E-mail address,
	mobile number, departme
		signIn()
		signOut()
		changeInfo()
		setPassword()

	name, location,
	user id,
	Password
	balance
		viewMenu()
		orderFood()
		modifyOrder()
		cancelOrder()
		initializeBooking()
		confirmBooking()
		modifyBooking()
		cancelBooking()
		payingBill()
		verifyAnApplicant()
		giveAFeedback()
		giveRating&Comment()
		getRating&Comment()
		setBalance()
		getBalance()

Guest user	reference email id	viewMenu()
		orderFood()
		modifyOrder()
		cancelOrder()

User	IP address,
	timestamp
		viewMenu()
		orderFood()
		modifyOrder()
		cancelOrder()
		signUp()

		getIPAddress()
		getTimeStamp()

Admin	name,
	employee id, designation
	E-mail address,
	mobile number,
	user id,
	password
		signIn()
		signOut()
		changeInfo()
		setPassword()
		viewFoodMenu()
		modifyFoodMenu()
		verifyBooking()
		updateStockStorage()
		viewActivityLog()
		addCashOnPayment()
		addReferenceUser()
		assignStaffToWork()
		viewStaffInfo()
		modifyStaffInfo()
		giveReplyToFeedback()

Booking	booking id
	booking type
	authorized user id
	number of people
	food package id
	quantities
		changeBookingDate ()
		setCancelDate ()
		setTimeSlot()
		setStatus()
		setNumberOfPeople()

	date
	cancel date
	time slot
	status
		setFoodPackages()
		setQuantities()

Order	order type
	order id
	date
	time
	authorized user id
	cancel time
	order status
	quantity
	delivery location
	staff id
		setOrderType()
		changeFoodItem()
		setQuantity()
		setCancelTime()
		setOrderStatus()
		setStaffForDelivery()

Payment	transaction id
	payment type
	payment method
	user id
	amount
	status
	date
		choosePaymentType()
		choosePaymentMethod()
		setAmount()
		setStatus()
		getAmount()
		getStatus()
		getDate()

Food management	food id
	food name
	quantity
	type
	date
	price
		setQuantity()
		setPrice()
		addToStock()
		addToFoodMenu()
		changeDate()
		removeAnItem()

Financial repor	transaction id
	amount
	payment type
	payment method
	accounts payable
	accounts receivable
	income
	expenditure
	profit
		showReport()
		calculateProfit()
		setAccountsPayable()
		setAccountsReceivable()
		getAccountsPayable()
		getAccountsReceivable()

Data analytics	Date
	Number of customers
	Income
	Expenditure
	Food id
		showTheProfitableDay()
		getNumberOfCustomers()
		showTheMostSoldFood()
		showAnalyticsReport()

HR manageme	staff id
	staff name
	attendance
	salary
	working status
	working hour
		changeStaffInfo()
		getAttendance()
		getStatus()
		setStatus()
		fireAnStaff()
		addAnStaff()
		setWorkingHour()
		getWorkingHour()

DataBase management		insert()
		update()
		delete()
		search()

Interface	user	ShowMenu()
		getInput()
		authenticate()
		getOrderInfo()
		getBookingInfo()
		getPaymentInfo()
		showMessage()
		showFinancialReport()
		showRating&comment()

System	Interface	connectToDatabase()

	Database	modifyDatabase()
		connectToAccountingSoftware
		)
		verify()
		sendNotification()
		updateFoodMenu()
		updateFinancialReport()
		connectToThirdPartyAgent()
		saveActivityLog()

6.5 Finalizing Classes:

To identify final classes we need to first check that if there can be any hierarchies or merges. These are given below -

1. Teacher, Officer and Authorized user have same attributes and methods. So, we merge them as Authorized user.
2. User and Guest user have common functionalities. So, we merge them as User. Attributes are -

❑ IP address

❑ Timestamp

❑ Reference email id

3. Since we have two different types of classes named Authorized user and Admin, we can eliminate the Account class to avoid data redundancy.

4. Data analytics class has almost similar execution as Financial report. Also this class is small and doesn’t do any important particular work. So , we merge it into Financial report.

5. We have used two new classes to cover the CMS system.

One is System and other is Interface.

Attributes of System class :

❖ Interface

❖ Database Management

And methods are :

❑ connectToDatabase()

❑ modifyDatabase()

❑ connectToAccountingSoftware()

❑ verify()

❑ sendNotification()

❑ updateFoodMenu()

❑ updateFinancialReport()

❑ connectToThirdPartyAgent()

❑ saveActivityLog()

6.The Interface class interacts with all users and so users perform anything through the Interface class. So it has no attributes but users.

The methods are:

❏ ShowMenu()
❏ getInput()
❏ authenticate()
❏ getOrderInfo()
❏ getBookingInfo()
❏ getPaymentInfo()
❏ showMessage()
❏ showFinancialReport()
❏ showRating&Comment()

So after the analysis we found below classes :

- System
- Interface
- User
- Authorized user
- Admin
- Order
- Booking
- Payment
- Food management
- HR management
- Financial report
- Database management

6.6 Class Index Card:

After identifying our final classes we have generated the following class cards.

Table 17: Class Card for System Class

System	

Interface
Database
	connectToDatabase()
	modifyDatabase()
	connectToAccountingSoftware()
	verify()
	sendNotification()
	updateFoodMenu()
	updateFinancialReport()
	connectToThirdPartyAgent()
	saveActivityLog()

Responsibilities	***Collaborator***
Managing database	Database
Creating & showing interface	Interface
Sending notification	Authorized user

Updating food menu	Food management
Updating financial report	Financial management

Table 18: Class Card for Interface Class

Interface	

user	ShowMenu()
	getInput()
	authenticate()
	getOrderInfo()
	getBookingInfo()
	getPaymentInfo()
	showMessage()
	showFinancialReport()
	showRating&Comment()

Responsibilities	***Collaborator***
Letting the user using th	User

system	
Getting input from user	User
Taking order, booking & payment request	User
Showing message	User
Showing financial repor	System,Admin
Showing staff's, authorized user's information	Database management,Admin
Showing rating & comment	Database

Table 19: Class Card for User Class

User	

IP address
timestamp
reference email id
	viewMenu()
	orderFood()
	modifyOrder()
	cancelOrder()
	signUp()
	getIPAddress()
	getTimeStamp()

Responsibilities	***Collaborator***
Viewing food menu	Food management
Ordering food	Order
Creating account	Admin , System

Table 20: Class Card for Authorized User Class

Authorized user	

name,
employee id,
designation,
E-mail address,
mobile number,
department name,
location,
user id,
password
Booking_id
food_id
	signIn()
	signOut()
	changeInfo()
	setPassword()
	viewMenu()
	orderFood()
	modifyOrder()
	cancelOrder()
	initializeBooking()
	confirmBooking()
	modifyBooking()
	cancelBooking()
	payingBill()
	verifyAnApplicant()
	giveAFeedback()
	giveRating&Comment()
	getRating&Comment()
	setBalance()
	getBalance()

Responsibilities	***Collaborator***
Signing in & signing ou	Database management

Booking for parties	Booking,Admin
Making payment	Payment,Admin
Verifying an unauthorized user	Database management
Giving rating and showing rating	Database management
Showing balance	Database management

Table 21: Class Card for Admin Class

Admin	

name,
E-mail address,
mobile number,
user id,
password
Staff_id
Booking_id
food_id
	modifyFoodMenu()
	verifyBooking()
	updateStockStorage()
	viewAccessLog()
	addCashOnPayment()
	addReferenceUser()
	assignStaffToWork()
	viewStaffInfo()
	modifyStaffInfo()

	giveReplyToFeedback()
Responsibilities	***Collaborator***
Updating food menu ar stock	Food management
Verifying booking	Booking, Authorized user
Managing staff information	HR management
Verifying unauthorized user	Database management
Viewing access log	User , Database management
Recording cash on payment	Payment
Assigning a staff for delivery	Order
Replying to comments and complains	Database

Table 22: Class Card for Booking Class

Booking	

booking id
booking type
authorized user id
number of people
food package id
quantities
date
cancel date
time slot
status
	changeBookingDate ()
	getCancelDate ()
	getTimeSlot()
	getStatus()
	getNumberOfPeople()
	getFoodPackages()
	getQuantities()
	setCancelDate ()
	setTimeSlot()
	setStatus()
	setNumberOfPeople()
	setFoodPackages()
	setQuantities()

Responsibilities	***Collaborator***
Changing booking in	Authorized user, Admin
Recording booking info	Database management

Table 23: Class Card for Order Class

Order	
order type	getOrderType()
order id	getQuantity()
date	getCancelTime()
time	getOrderStatus()
authorized user id	setOrderType()
cancel time	setQuantity()
order status	setCancelTime()
quantity	setOrderStatus()
delivery location	changeFoodItem()
staff id	setStaffForDelivery()
Responsibilities	***Collaborator***
Changing order info	Authorized user, Admin
Recording order info	Database management

Table 24: Class Card for Payment Class

Payment	
transaction id	choosePaymentType()
payment type	choosePaymentMethod()
payment method	setAmount()
user id	setStatus()
amount	getAmount()
status	getStatus()
date	getDate()
Responsibilities	***Collaborator***
Changing payment inf	User, Admin
Recording payment inf	Database management

Table 25: Class Card for Food management Class

Food management	

food id
food name
quantity
type
date
price
	setQuantity()
	setPrice()
	addToStock()
	addToFoodMenu()
	changeDate()
	removeAnItem()
	setRating&Comment()

Responsibilities	***Collaborator***
Storing stock informatio	Admin, Database management
Updating food informatio	Admin, Database management
Adding food item to foo menu	Admin
Adding rating and comment	Authorized_User, Database

Table 26: Class Card for HR management Class

HR management	

staff id
staff name
attendance
salary
working status
working hour
	changeStaffInfo()
	getAttendance()
	getStatus()
	setStatus()
	fireAnStaff()
	addAnStaff()
	setWorkingHour()
	getWorkingHour()

Responsibilities	***Collaborator***
Update staff informatio	Admin

Table 27: Class Card for Financial report Class

Financial report	

transaction id
amount
payment type
payment method
accounts payable
accounts receivable
income
expenditure
profit
	showReport()
	calculateProfit()
	setAccountsPayable()
	setAccountsReceivable()
	getAccountsPayable()
	getAccountsReceivable()

Responsibilities	***Collaborator***
Calculating profit	Payment
Showing financial repor	Admin
Recording debit and credit information	Database management

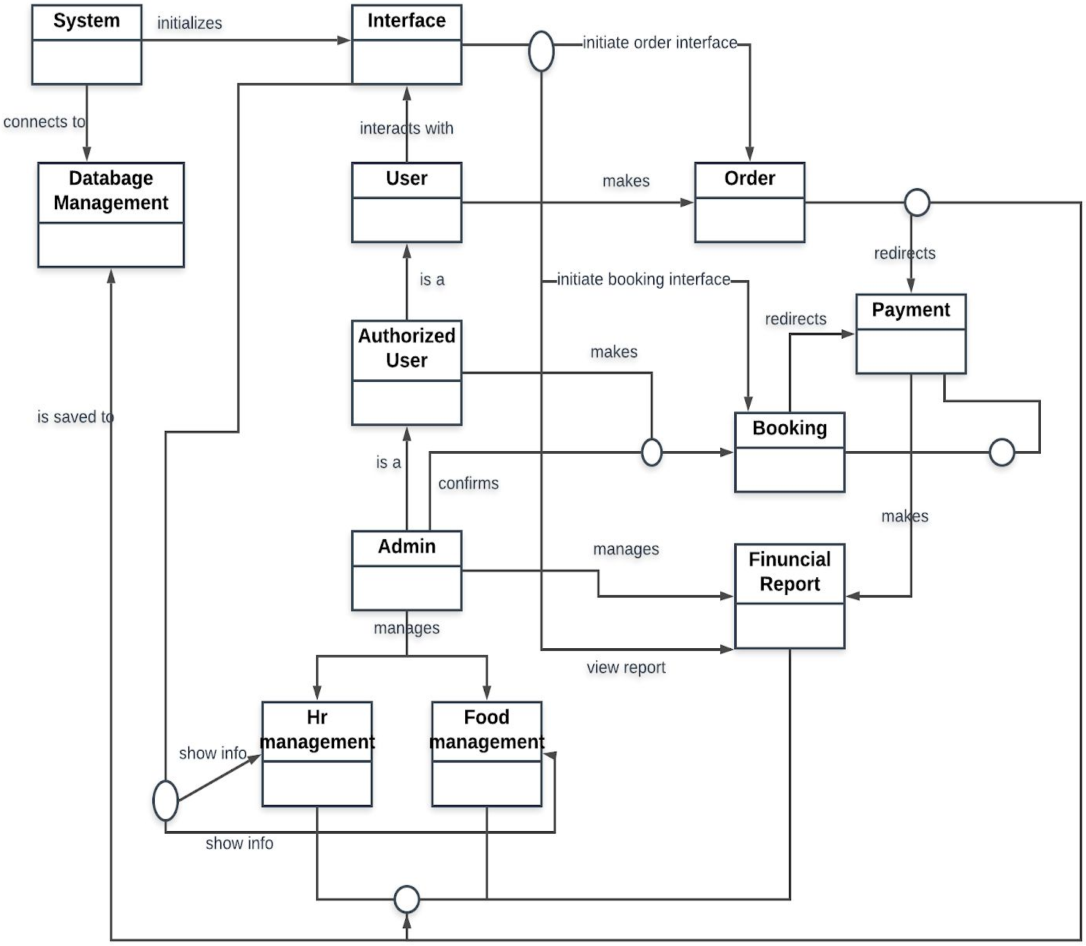
Table 28: Class Card for Database management Class

Database management	

	insert()
	update()
	delete()
	search()

Responsibilities	***Collaborator***
Handles all databases i the system	User, HR management , Food management, Financial report, Orde Booking []

Class Diagram



7 BEHAVIORAL MODELING OF GMS

7.1 STATE TRANSITION DIAGRAM:

State diagram represents active states for each class the events (triggers). For this, we identified all the events, their initiators and collaborators.

Table: Event Identification

Serial N	Event	Event name	Initiator	Collaborator
1.	View food menu	view_menu	User	Food management
2.	Order food with reference	ref_order	User	Authorized use Order
3.	interact with the system	interact	Interface	System
4.	Create an account with reference	sign_up_with_ref	User	System, Authorized use
5.	Create an account	sign_up_without_r	User	System, Admir

	without reference			
6.	Provide informatior	prvd_info	User	
7.	Initially, some valid users added	add_valid_user	Admin	Database
8.	Verification reques will be sent	verify_req	System	Authorized use
9.	Verification code consists of 6 digits w be sent	verify_code	System	User
10.	Log into the systen	log_system	Authorized user	System, Database
11.	Match information i Dhaka University's Database	verify_info	Admin	
12.	Get the services of CMS	get_services	Authorized user	System
13.	Provide username, password for login	authentic_info	User	Interface
14.	Reset password an provide email	reset_pass	Authorized user	

15.	Verification code fo reset password	verify_code_for_pa s	System	User, Database
16.	Change account inf	change_info	Authorized user	Database
17.	Provides lunch, snacks, and refreshment items	provide_lunch	System	Food management
18.	Initially update the number of food item	init_update_quanti	Admin	System
19.	Add, reduce and modify food menu	modify_food_men	Admin	System
20.	Reduce remaining food items after confirming food orde	reduce_food_quan y	System	
21.	Increase remaining food items after confirming food orde	increase_food_qua tity	System	
22.	Access stock management	acc_stock_mng	Admin	Food management

23.	View stocked food	view_stock	Admin	Food management
24.	Calculate needed food quantity and items to buy	cal_food	Admin	Food management
25.	Update and modify food items with the corresponding quantity	update_stock	Admin	Food management
26.	Order food items	order_food	Authorized user	Order
27.	Order food manuall	order_manual	User	Order
28.	Add order info	order_info	Admin	Order, Databas
29.	Cost will be shown	show cost	Interface	User
30.	Redirected to the payment system	payment_gui	System	Interface, Payment
31.	Seat reservation for particular time	seat_rsrvnt	Admin	User, Order
32.	Provide a location fo delivery	provide_loc	User	System

33.	Provide time duratio for modifying food items	provide_time	System	Authorized use
34.	Notify via SMS / E-mail	notifying	System	Authorized use
35.	Modify food order	modify_order	User	Order
36.	Cancel food order	cancel_order	User	Order
37.	Provide a valid user email for order	ref_email	User	Order
38.	Forwarding food ord	forward_order	Authorized user	User, Order
39.	Booking for party lunch and buffet	booked	Auhtorizeduer	Booking
40.	Choose an availabl date and time	choose_slot	Authorized user	Booking
41.	Choose food packag with the corresponding quantity	choose_food	Authorized user	Booking
42.	Notification for	notification	Admin	Authorized use

	payment			
43.	Cancel booking for not to complete the payment	cancel_booking_sy	System	Booking
44.	Cancel booking	cancel_booking	Authorized user	Booking
45.	Confirm booking fo completing paymen	confirm_booking	Authorized user	Booking
46.	Payment will be recorded with transaction id and user-id	record_transectio	Payment	Database, Financial repor
47.	Choose payment typ	payment_type	Authorized user	Payment
48.	Choose payment method	payment_methoc	Authorized user	Payment
49.	Modify staff information	modify_satff_info	Admin	HR managemen
50.	Supervise on staff's status, working hour fire and recruitmen	controlling_staff	Admin	HR managemen

51.	Calculating income expenditure, profit, and loss	calculate_earning	Financial report	Database
52.	Update and modify accounting information	modify_account_in o	Admin	Financial report Database
53.	Showing report base on data analysis	data_mining_repo	System	Admin
54.	Activity log will be recorded	log_info	System	User, Database
55.	Modify activity log in	modify_log_info	Authorized user	Database
56.	IP address, timestamp will be recorded	login_info	System	Admin
57.	Give feedback by rating and commen	feedback	Authorized user	Database, System, Admin
58.	Do complain and giv suggestions	complaint_hub	Authorized user	Database, System, Admin
59.	Connect to the	connect_database	System	Database

	database			
60.	Update information	update_info	Database	
61.	Delete information	delete_info	Database	
62.	Search information	search_info	Database	
63.	Insert information	insert_info	Database	
64.	Log out	Logging out	Authorized User	

7.1.1 EVENT AFTER ANALYSIS:

After analysis we merge some events and delete some events which are not important. We found -

1. “reset_pass”, “verify_code_for_pass” and “change_info” merge into “changing info” event.

2. “get_services” event can’t consider an event. Because it means a total services events. So we delete this event.

3. “provide_lunch”, “acc_stock_mng”,“payment_gui” can not be an event because those are not considered as an event.

4. “provide_loc”, “choose_slot”, and “choose_food” are merged into “booked” event.

5. “modify_order”, “cancel_order” and “ref_email” are merge into “change order info”. Because those are single work.

6. “modify_satff_info” and “controlling_staff” are merged into “staff_info”.

7. “calculate_earning” change named “keeping all transaction info”.

Table: Remaining event after analysis.

Serial N	Event	Event name	Initiator	Collaborator
1.	View food menu	view_menu	User	Food management
2.	Order food with reference	reference order	User	Authorized user Order
3.	interact with the system	interact	Interface	System

4.	Create an account with reference	sign up with reference	User	System, Authorized use
5.	Create an account without reference	sign up without reference	User	System, Admin
6.	Provide information	provide info	User	
7.	Initially, some valid users added	add valid user	Admin	Database
8.	Verification reques will be sent	verify request	System	Authorized use
9.	Verification code consists of 6 digits w be sent	verify code	System	User
10.	Log into the system	log system	Authorized user	System, Database
11.	Provide username, password for login	authentic_info	User	Interface
12.	Notify via SMS/ E-mail	notifying	System	Authorized use
13.	Change account inf	changing_info	Authorized	Database

			user	
14.	Initially update the number of food item	init_update_quanti	Admin	System
15.	Add, reduce and modify food menu	modify_food_men	Admin	System
16.	Reduce remaining food items after confirming food orde	reduce_food_quany	System	Interface
17.	Increase remaining food items after confirming food orde	increase_food_quatity	System	Interface
18.	View stocked food	view_stock	Admin	Food management
19.	Calculate needed food quantity and items to buy	calculate food	Admin	Food management
20.	Access stock management and update and modify food items with the	update_stock	Admin	Food management

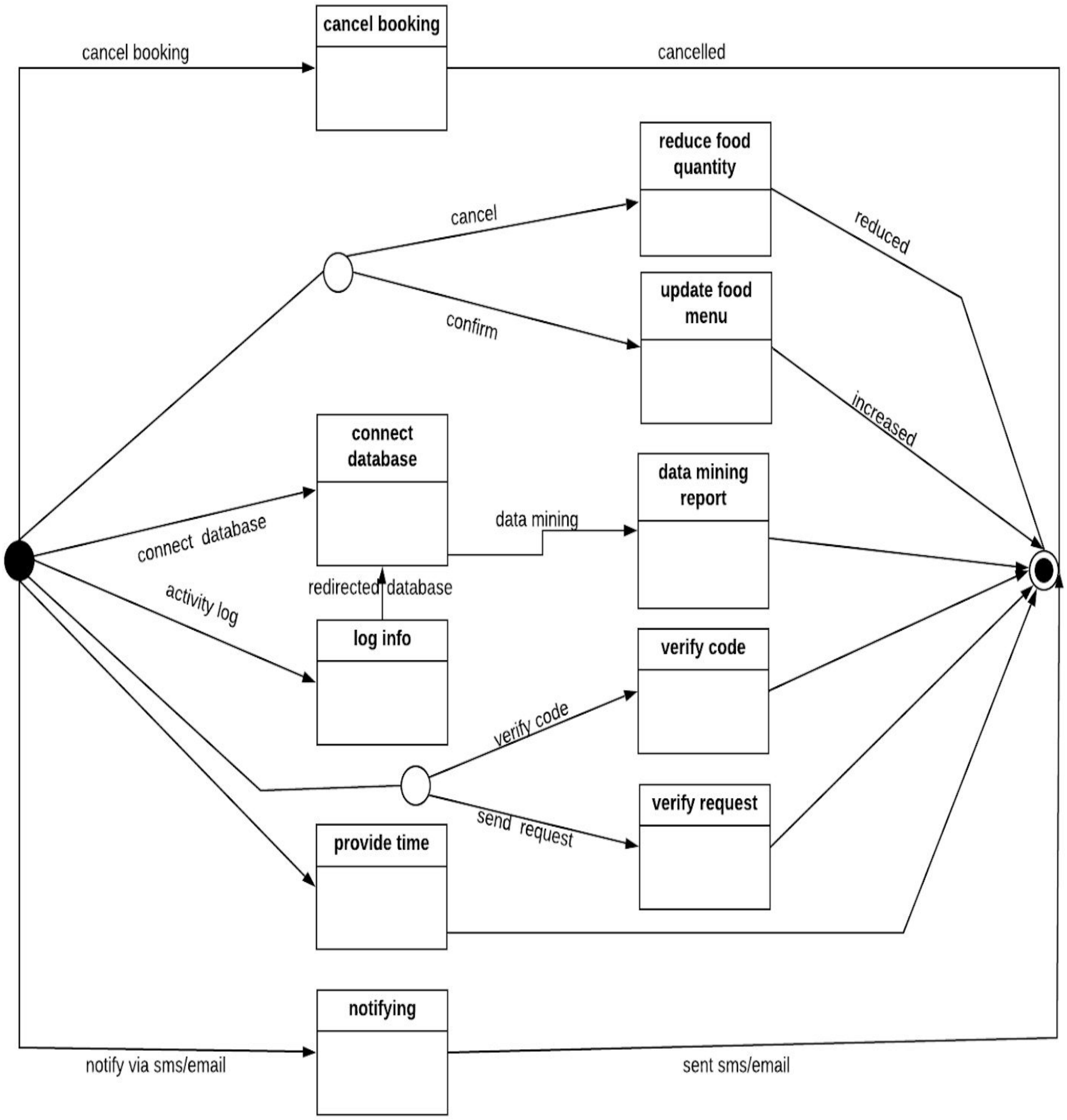
	corresponding quantity			
21.	Order food items	order	Authorized user	Order
22.	Order food manuall	order_manual	User	Order
23.	Cost will be shown	show cost	Interface	User
24.	Seat reservation for particular time	seat reservation	Admin	User, Order
25.	Provide time duratio for modifying food items	provide_time	System	Authorized use
26.	Modify and cancel food order	change order info	User	Order
27.	Forwarding food ord	forward_order	Authorized user	User, Order
28.	Booking for party lunch and buffet	booked	Authorized user	Booking
29.	Cancel booking	cancel booking		
30.	Notification for payment	notification	Admin	Authorized use

31.	Choose payment typ	payment_type	User	Payment
32.	Choose payment method	payment_methoc	User	Payment
33.	Modify staff information	satff_info	Admin	HR managemer
34.	Calculating income expenditure, profit, and loss	keeping all transaction info	Financial report	Database
35.	Update and modify accounting information	modify_acc_info	Admin	Financial report Database
36.	Showing report base on data analysis	data_mining_repo	System	Admin
37.	Activity log will be recorded	log_info	System	User, Database
38.	Modify activity log in	modify_log_info	Authorized user	Database
39.	IP address, timestamp will be recorded	login_info	System	Admin

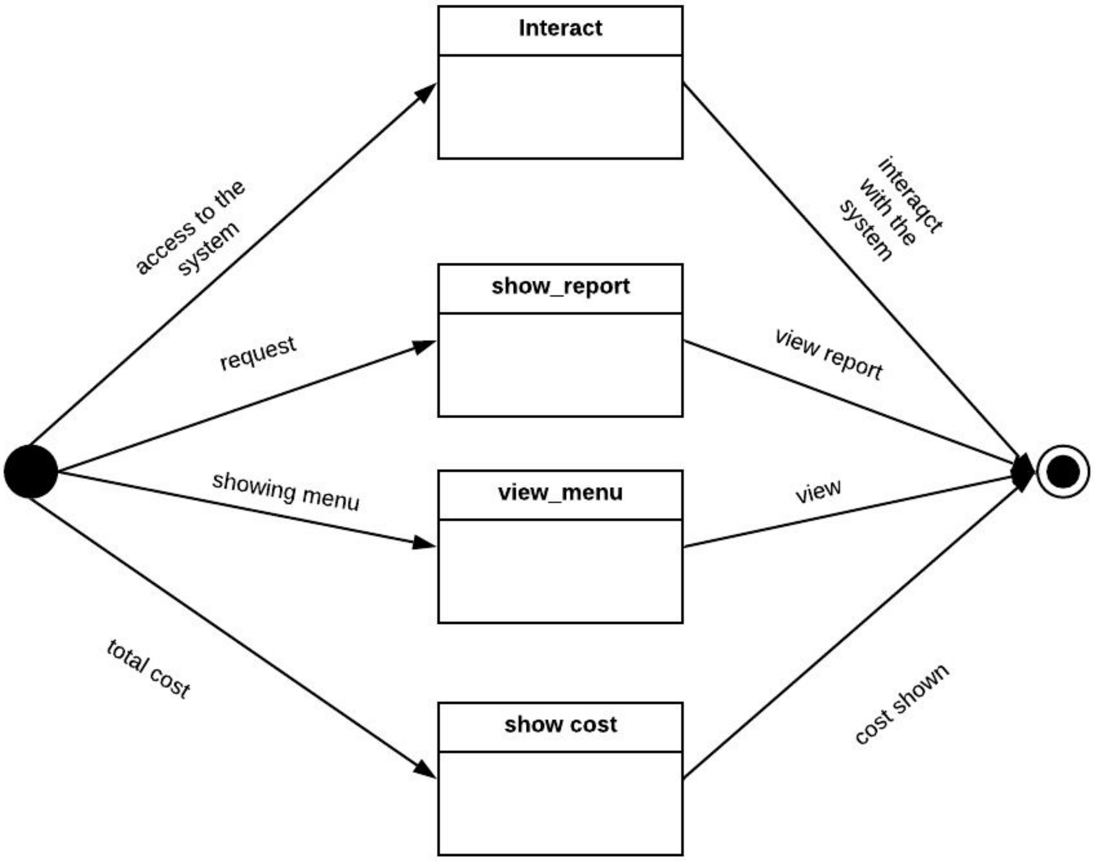
40.	Give feedback by rating and commen	feedback	Authorized user	Database, System, Admin
41.	Do complain and giv suggestions	complaint_hub	Authorized user	Database, System, Admin
42.	Connect to the database	connect_database	System	Database
43.	Update information	update_info	Database	
44.	Delete information	delete_info	Database	
45.	Search information	search_info	Database	
46.	Insert information	insert_info	Database	

7.1.2 STATE TRANSITION DIAGRAM:

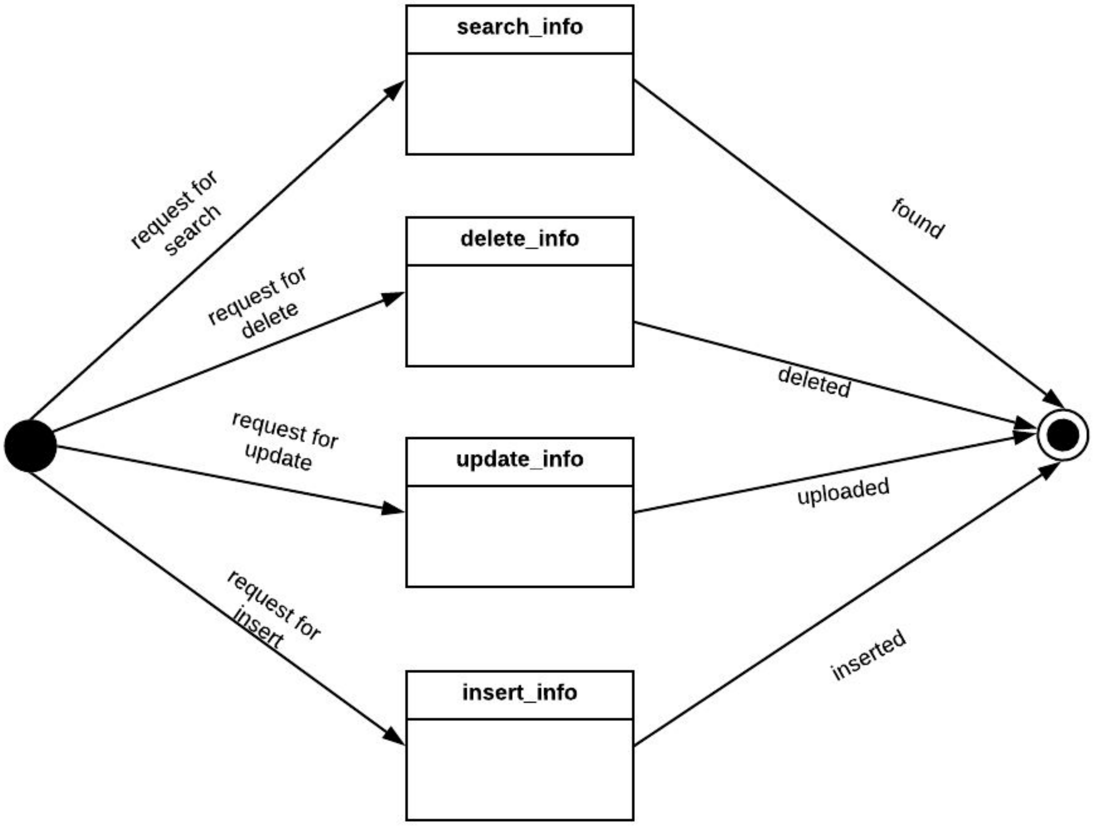
SYSTEM



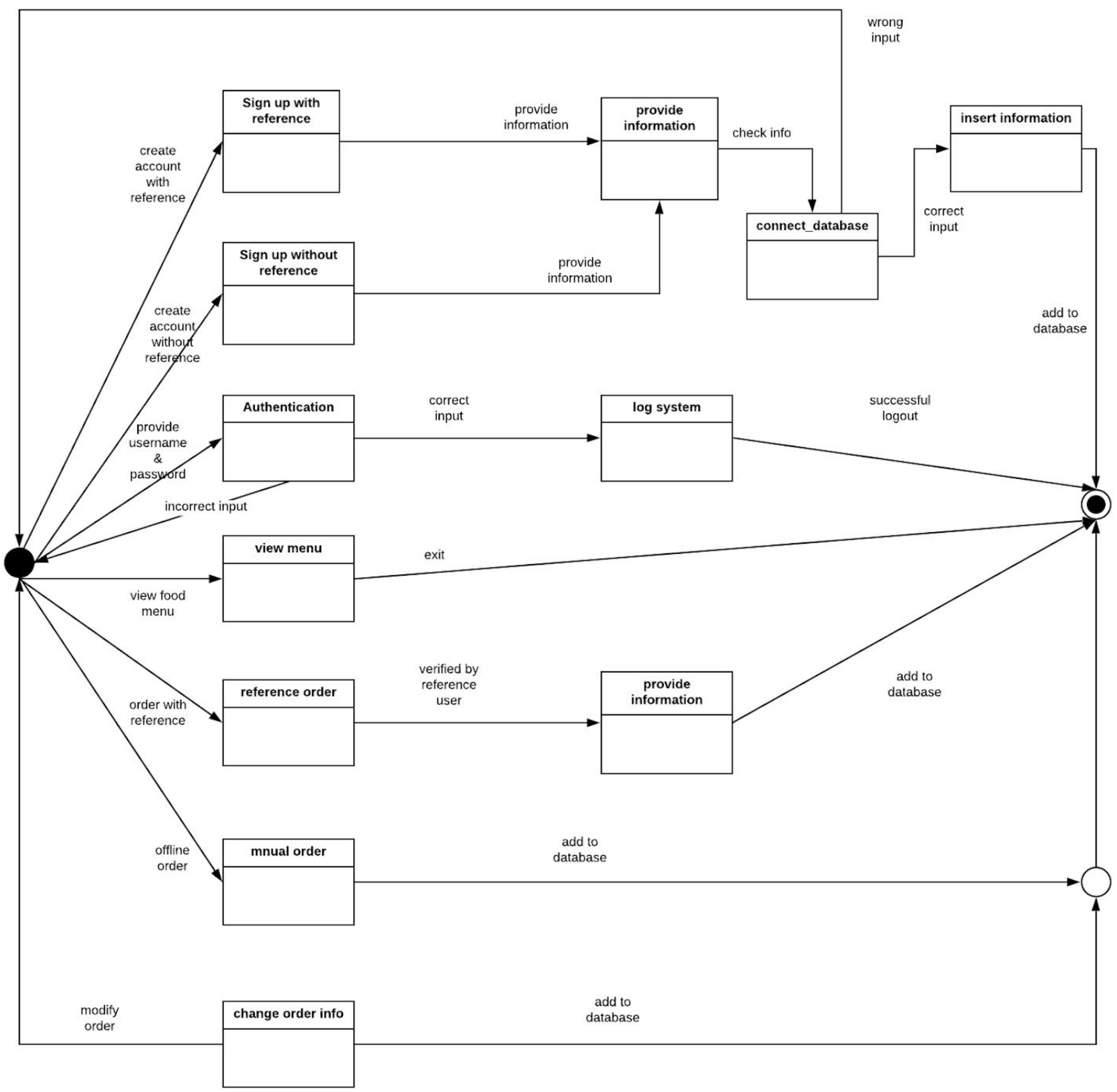
INTERFACE


Interact
show_report
view_menu
show cost
access to the system
request
showing menu
total cost
interaqct with the system
view report
view
cost shown

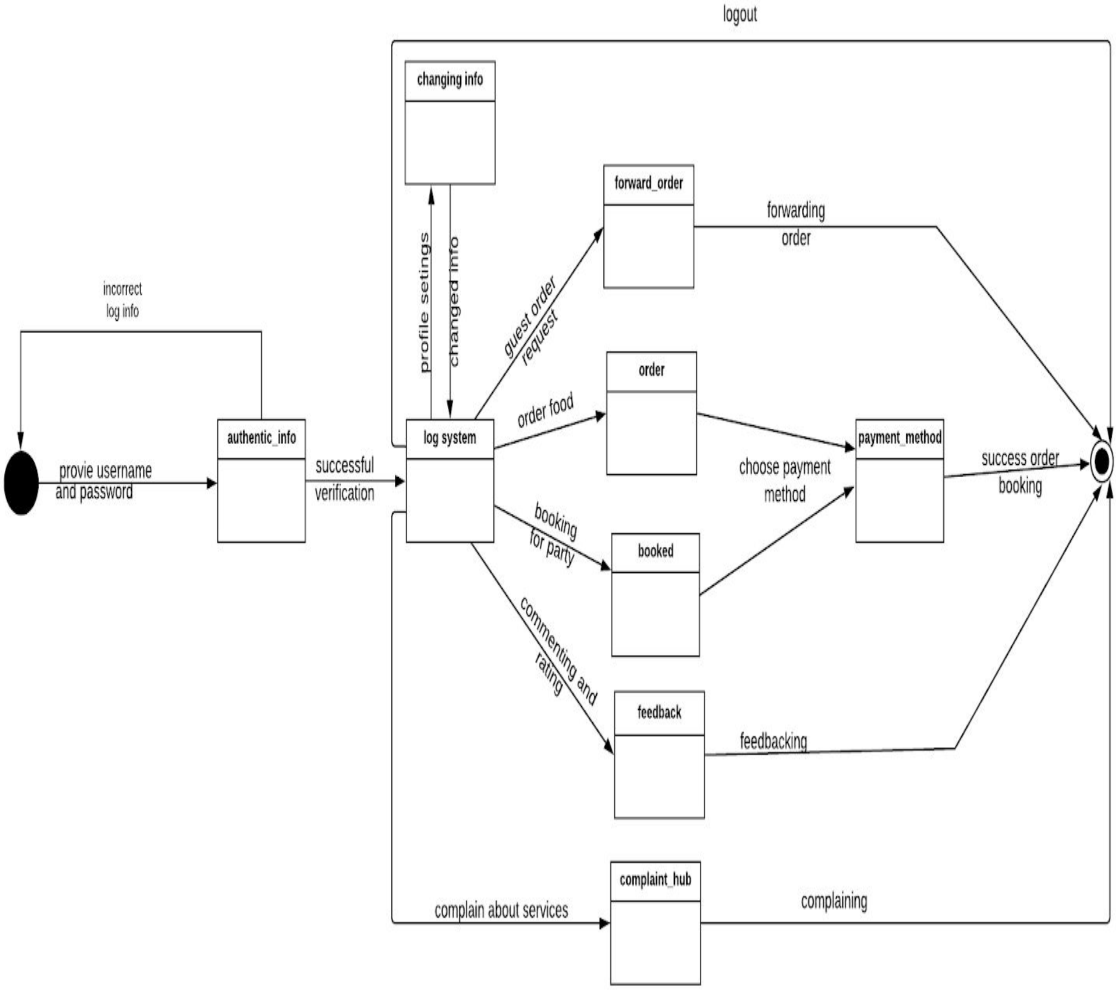
DATABASE


search_info
delete_info
update_info
insert_info
request for search
request for delete
request for update
request for insert
found
deleted
uploaded
inserted

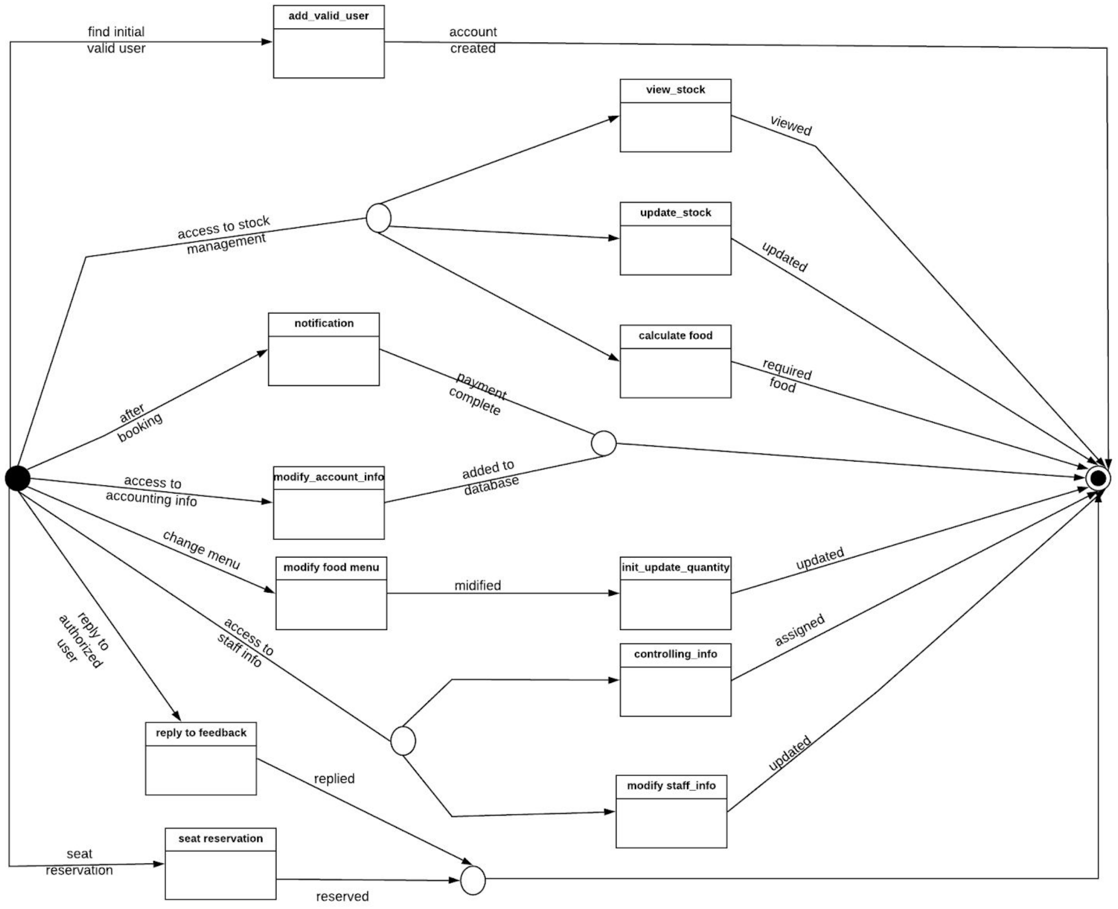
USER


wrong input
Sign up with reference
provide information
provide information
check info
insert information
create account with reference
connect_database
correct input
Sign up without reference
provide information
create account without reference
add to database
Authentication
correct input
log system
successful logout
provide username & password
incorrect input
view menu
exit
view food menu
verified by reference user
provide information
add to database
reference order
order with reference
offline order
mnual order
add to database
modify order
change order info
add to database

AUTHORIZED USER

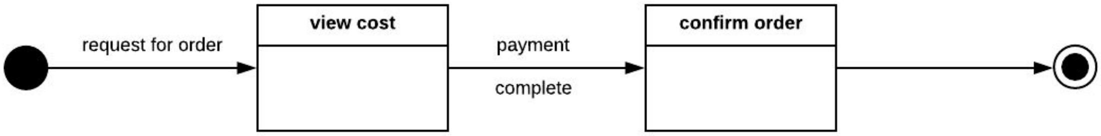


ADMIN

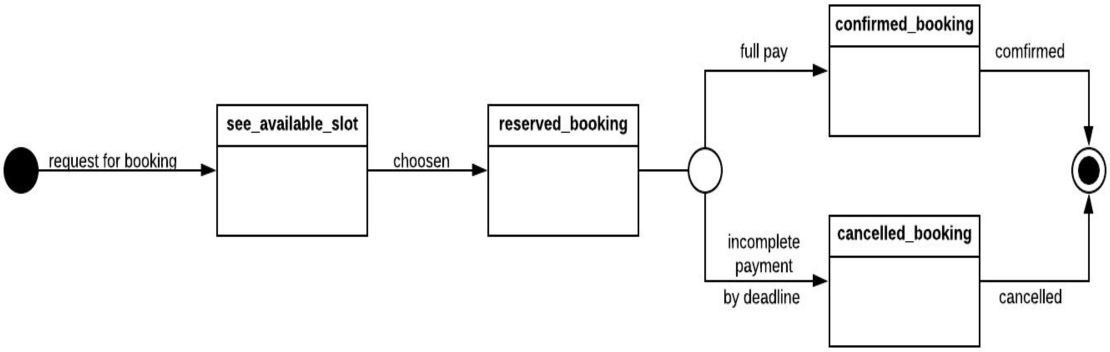


add_valid_user
find initial valid user
account created
view_stock
viewed
access to stock management
update_stock
updated
notification
calculate food
required food
payment complete
after booking
modify_account_info
access to accounting info
added to database
change menu
modify food menu
midified
init_update_quantity
updated
reply to authorized user
access to staff info
controlling_info
assigned
reply to feedback
modify staff_info
updated
replied
seat reservation
seat reservation
reserved

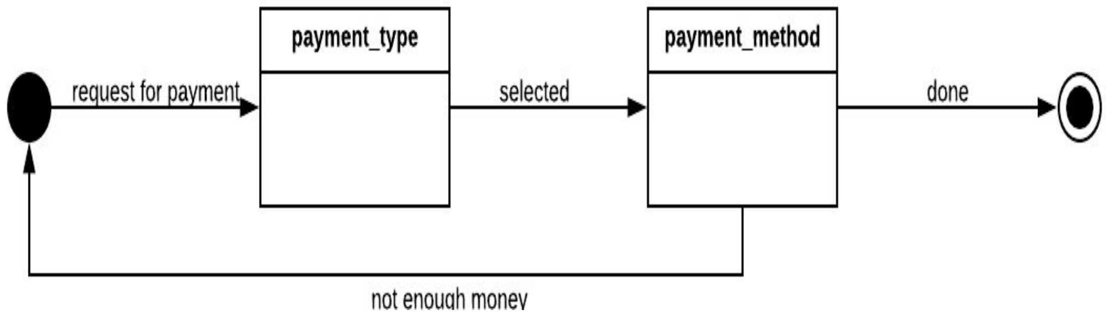
ORDER



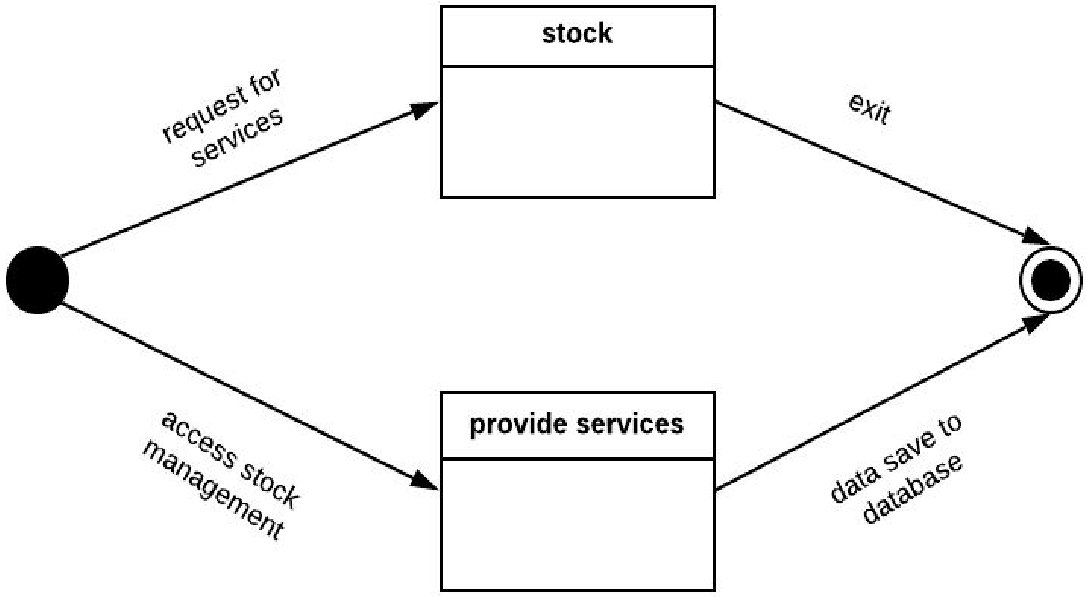
BOOKING



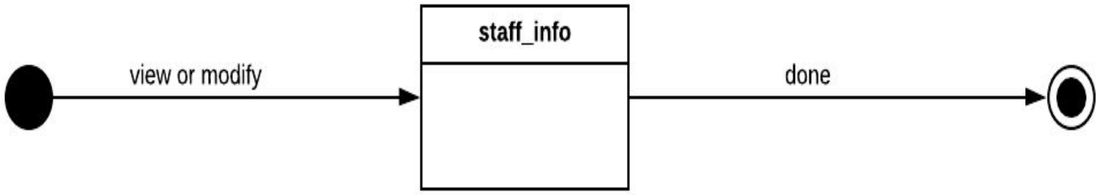
PAYMENT



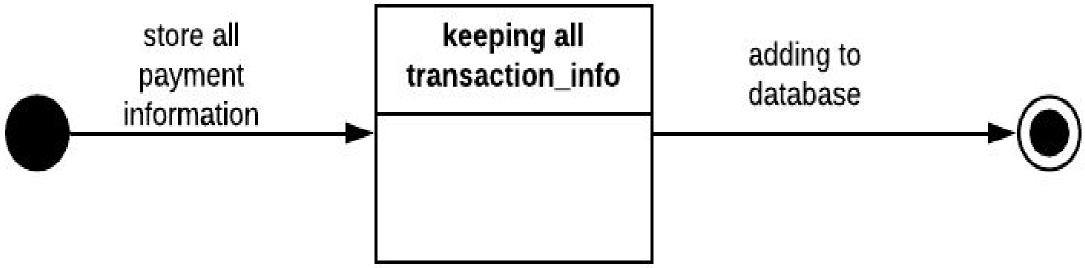
FOOD MANAGEMENT



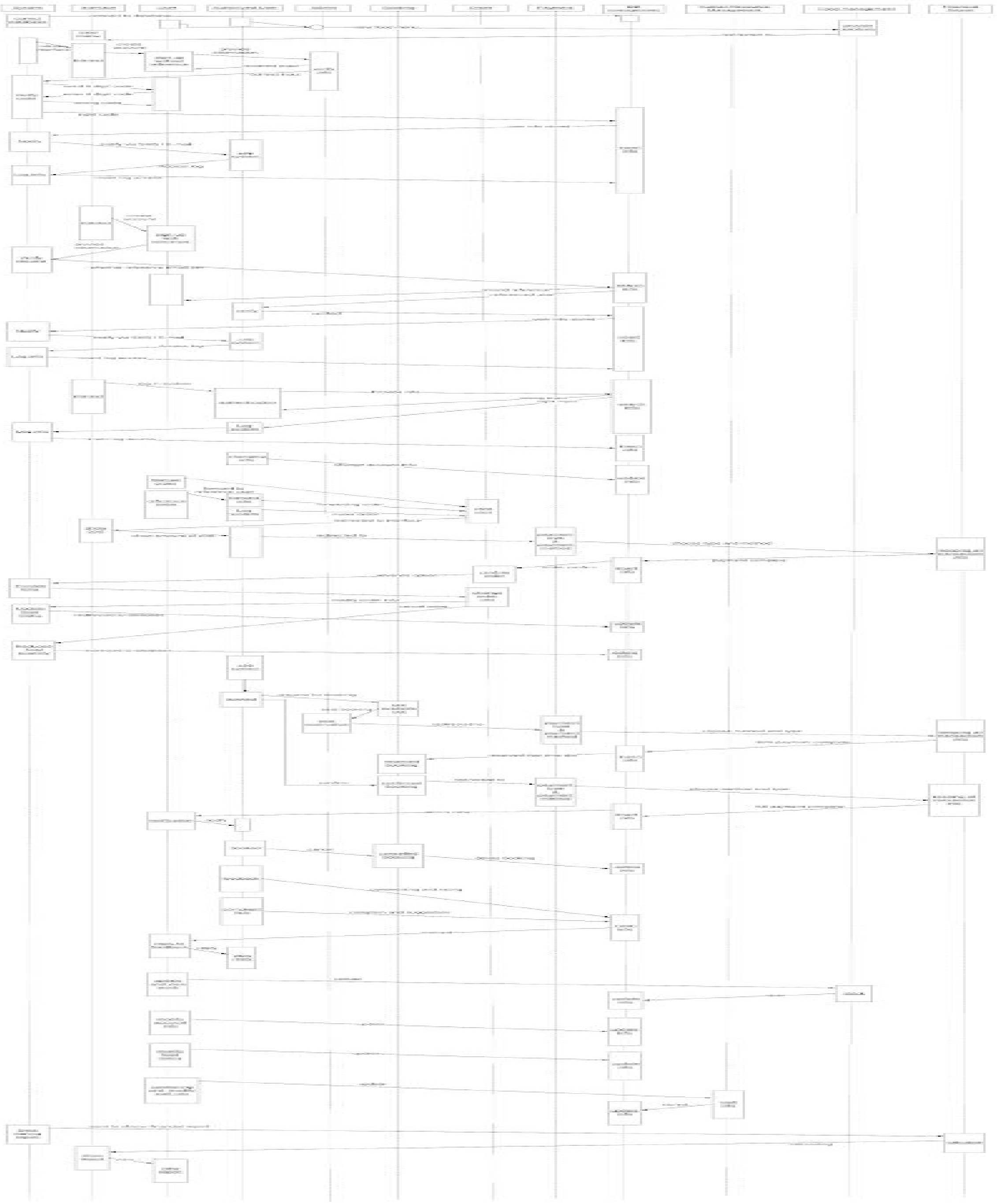
HR MANAGEMENT



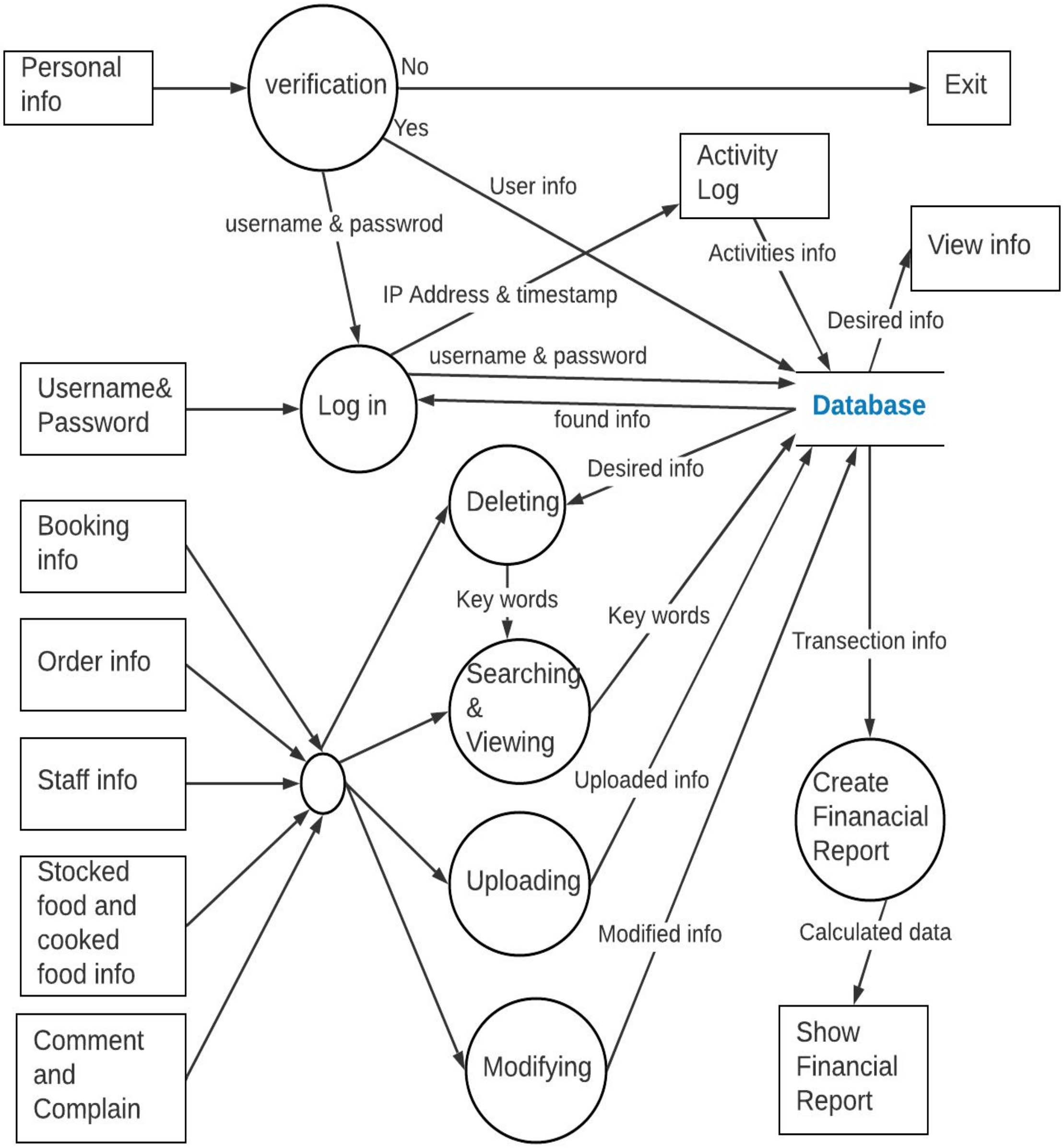
FINANCIAL REPORT



7.2 SEQUENCE DIAGRAM



7.3 DATA FLOW DIAGRAM:


Personal info
verification
No
Exit
Yes
User info
Activity Log
username & passwrod
Activities info
View info
IP Address & timestamp
Desired info
username & password
Username& Password
Log in
Database
found info
Desired info
Deleting
Booking info
Key words
Key words
Order info
Searching & Viewing
Transection info
Staff info
Uploaded info
Create Finanacial Report
Stocked food and cooked food info
Uploading
Modified info
Calculated data
Comment and Complain
Modifying
Show Financial Report