Institute of Information Technology

University of Dhaka

BAKERY

MANAGEMENT SYSTEM

Software Requirements Specification and Analysis [SE-406]

Submitted By

GROUP - 4

Md. Siam (BSSE-1104)

Mushfiqur Rahman Chowdhury (BSSE1110)

Md. Muktar Hossain (BSSE-1116)

Saad Sakib Noor (BSSE-1122)

Tasmia Zerin (BSSE-1128)

Khalid Hasan Saad (BSSE-1133)

Submitted To

Dr. Kazi Muheymin-Us-Sakib

Professor,

Institute of Information Technology

University of Dhaka

30th September, 2021

Table of Contents

Contents **Page No.**

Introduction

There's a lot of competition in food businesses today. One wrong ingredient and you can end up making a sugary mess. The most common myth revolving around shifting to technology is that it's an expensive and exhausting exercise. However, that's not true if you know what technology to choose for your bakery. Once you have the right technology in place, you can manage your bakery like never before and increase sales overnight.

Again, direct communication and customer engagement are helpful in gaining the overall image of the organization. Bakery Management System (BMS) helps a bakery authority to automate, organize, balance and engage directly to its stakeholders. This document is intended to serve the documentation and provide an official means of communicating the user requirements to the developers and stakeholders.

Purpose

This documentation briefly describes the Software Requirement Specification of a Bakery Management System. It contains the functional, non-functional and the supporting requirements and establishes a requirement's baseline for the development of the system. The goal of this SRS is to serve as an official means of communicating user requirements to the developers and stakeholders' community. The SRS will evolve over time as users and developers work together to validate, clarify and expand its contents.

Intended Audience

This SRS report is intended for several audiences. We discussed with all the possible stakeholders and developed this SRS that gives a clear guideline and documentation to -

- The users and admin 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 a delivery date, and ensure that the developing team is on track during 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.

Conclusion

The analysis helped us figure out the stakeholders and requirements systematically. Moreover, it helped us to focus on the users who will be using our analysis. The document will be helpful to each and every stakeholder of the software to understand, relate and ensure convenient usage of the software. Developers and testers will have some clear ideas of what to do and development will be smoother. More and more communication between the stakeholders and developers will help evolve the software and enhance it further.

Inception of Bakery Management System

The project begins with the inception (a task that defines the scope and nature of the problem to be solved). In order to make this phase effective we took the following steps:

- Establishing a basic understanding of the problem
- Identifying the clients of the solution
- Identifying the stakeholders of the solution
- The nature of the solution that is desired
- Establishing preliminary communication and collaboration between the stakeholders and the software team

Establishing a Basic Understanding of the Problem

The basic goal of the project was to create a software for bakery management automation. For that, we firstly tried to gather the proper understanding of how

the existing system works. After a few conversations with the owners and the employees of the bakery, we found out the existing system works as follows-

1. Collecting raw materials
2. Making various food items (example: bread, bun, biscuit, cake, sweet etc.)
3. Collecting orders from retailers and customers
4. Delivering the products and collecting money
5. Management of expired products
6. Employee management

Collecting Raw Materials

Currently the raw materials are collected from the local markets without knowing the actual amount needed for the production. Also, the tracks of the quantity of previous days are not maintained properly in their records. As a result, the entire collection of raw materials is totally held upon some guesses and primary ideas of the employees. So sometimes, this causes a huge wastage of the raw products which concludes in causing profit deduction. Automating this can minimize the losses and maximize the profitability.

Making Various Food Items

This bakery currently makes various types of food items such as, bread, bun, biscuit, cookies, cakes and different varieties of sweets. Sometimes they supply orders according to the customers' desire. But as they don't know the amounts to produce for each item, many times some items are wasted and some items fall short which

causes great loss to the bakery. By automating, the popular products can be identified and a probable amount for all the products can be estimated.

Collecting Orders

The existing system contains employees for collecting orders from the retailers. As there is no online system for collecting orders, the employees have to go to the markets, retailers to collect the orders. Even though the employees try hard to get the orders and have the proper knowledge of how much in quantity the goods need to be produced, the bakery managerial team reported to us that only a tiny portion of the production is based on the previous orders. Most of the prepared goods are sold based on the assumptions of the employees. It is also reported that the inability to reach all the retailers at the right time and the retailers reporting that they get to know the exact amount of product that is needed at night. So, the instant reachability to the producers from the end of the retailers becomes important sometimes. Our automated system giving the opportunity of ordering anytime from anywhere will enable the retailers the exact amount that they need and it will give the bakery a better idea of the required quantity to be produced.

Delivering the Products and Collecting Money

The employees deliver the products without knowing the exact amount ordered by each retailer, so this sometimes causes shortages in some particular areas and overflows in others. Thus, the remainders are to be returned and while returning, sometimes the products get damaged. After delivering the goods, the employee collects the money. As there is no record of how much quantity is required and how

much is damaged/returned, it is very hard for the company to estimate the proper quantity of sales of that day, sometimes the trust can be compromised by the employees which stays untraceable by the company. As our automation will have the exact amount of the orders, the employees can manage and deliver these properly, which will minimize the damage of the products. Also, the exact quantity of sales everyday can be tracked by the system.

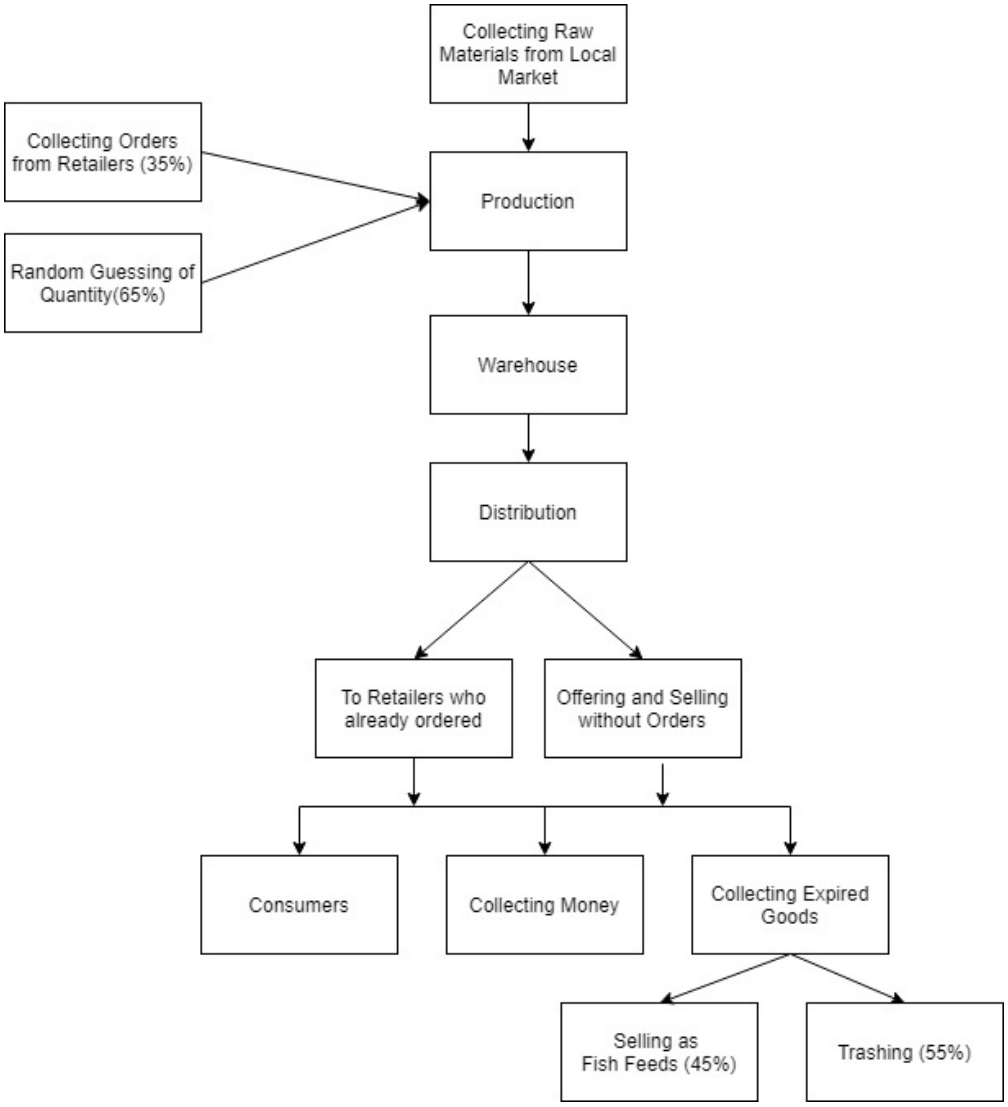
Management of Expired Products

The expired products are often trashed, sometimes some products are sold as fish feed at a cheaper price to fish producers. Our automation system will store the quantity of expired products and the fish producers can order the products from the system.

Employee Management

The manager distributes work among the employees (ex: assigning roles for taking order, dividing the area of service), tracks the working hours, gives salary and monitors all the employees. Our tool can help the manager handle all these tasks and keep track of performance and reward the best employee of the month which can motivate them to work better in the future.

Block Diagram for Bakery Management System (Existing)



Identifying the Clients of the Solution

Admin of the Hello Bakery can be identified as a client of our project.

Identifying the Stakeholders of the Solution

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 stakeholders of our solution are-

- Customers
- Employees
- Suppliers
- Retailers

The Nature of the Solution that is Desired

After communicating with the client and stakeholders, we identified the basic desires of them that the software should offer. The basic desires that the client and the stakeholders stated were as follows-

- Responsive and user-friendly interface
- It should automate the order taking procedures
- Online payment system should be integrated
- Manage the raw material information, and stocks
- Reliable supplier and retailer registration (multiple step authentication- ie. TIN, Address)

Establishing Preliminary Communication

We evaluated the problem with our proper solution and established a communication between the stakeholders and the software developer team. After getting the confirmation that the current solution doesn't compromise the user requirements and is feasible to be implemented by the dev team, we started the SRS of the project with the communicated schemes.

Elicitation Of Bakery Management System

Requirements elicitation (also called requirements gathering) combines elements of problem solving, elaboration, negotiation, and specification. In order to encourage a collaborative, team-oriented approach to requirements gathering, stakeholders work together to identify the problem, propose elements of the solution, negotiate different approaches, and specify a preliminary set of solution requirements. In our elicitation phase, we completed the following tasks –

- Collaborative Requirements Gathering
- Quality Function Deployment
- Usage Scenarios

Collaborative Requirements Gathering

We have conducted meetings with the stakeholders, such as, employees, retailers and customers. These meetings helped us to identify the problem, propose elements of the solution, negotiate different approaches, and specify a preliminary set of solution requirements.

Quality Function Deployment

Quality Function Deployment (QFD) is a structured approach of defining customer needs or requirements and translating them into specific plans to produce products to meet those needs. The "voice of the customer" is the term to describe these stated and unstated customer needs or requirements.

We visited the bakery “Hello Bakery” and talked to various level employees (manufacture, management, warehouse, owner). They maintain a completely manual procedure on all the stages, from collecting raw materials to delivery. For the automation of the Bakery System, the various level employees have emphasized their interest and area where they think the automation can add some convenience and maximize the profits. We also proposed some features we felt would be to them, and after discussing for a while, we came to the following requirements our automation tool will feature.

Normal Requirements: Normal requirements reflect objectives and goals stated for a product or system during meetings with the customer. Those are the basic requirements that fulfills client satisfaction. The Normal requirements we came up with are the following:

- Login and Sign Up: 7 User Types (Customer, Retailer, Supplier, Employee, Warehouse, Management, Admin)
- Customers and Retailers can sign up using their Name, Phone Number, Password, Address.
- Retailer and Supplier Sign Up should go through Admin Approval.
- All users will login using their Phone Number and Password
- Password recovery option will be present for the users who forgot their password.
- Users must be logged in before performing any operation.
- Admins will be given a predefined account.
- Users can order food online.
- Digital Memo will be given after the confirmation of the orders.

- Order details will be added into the virtual cart.
- Users can preorder the items for some specific days.
- Users can pay bills through virtual money transfer methods. (bKash, Nagad, Rocket) and cash on delivery.
- If any user cancels order before shipment, he/she will be refunded.
- Estimated date and time of food delivery will be prompt in the user's display Shipment status, Employee attendance, transaction information will be stored in the admin database.
- User information, order memo, transaction history will be stored in the user database.
- Admin can manage the everyday menu.
- If a food is out of stock, it will be marked as "Stock Out"

Expected Requirements: These requirements are so obvious that the customer need not explicitly state them. Their absence can create significant dissatisfaction. The expected requirements of our system are given below:

- The system will be secured and protected
- Interactive and user-friendly interface so that anyone can order an item without any difficulty.
- The delivery time of the orders will be estimated by GPS and shown in the user's display
- The stock number of every item will be updated regularly from the warehouse.
- Retailers can input the quantity of expired products to return for better manageability and track of the expired goods.

- Customers can place customized orders for different occasions with custom requirements.

Exciting Requirements: These requirements are beyond the user's expectations.

Following are the exciting requirements of our system:

- If a user orders an item frequently, similar items will be suggested to him/her.
- Artificial Intelligence will be used to estimate the number of items to be sold the next day, so there can be an efficient number of the items available.
- Users can design birthday cakes according to their choices and place an order.
- Users can give reviews and ratings of the food items and the software too.
- Customers will receive "Discount, Offers and Sales" notifications
- Artificial intelligence to figure out the best product to put on sales to maximize the profit and brand value.

USAGE SCENARIO

1) Account Management:

- **User Perspective:** User must create an account to enter into the system. Any user needs to provide following common information in order to create an account:
  - Full Name
  - Contact Number
  - Email Address
  - User Name
  - Password

A user can create only one account with the same phone number and email address. After providing the above information, the user has to provide user type. There will be 6 types of users: customer, retailer, supplier, employee and admin. Here is the required information for each type of user that they need to provide:

- **Customer:**
  - Delivery address
  - promotional mail option
- **Retailer & Supplier:**
  - Business id
  - Tax record (TIN)
  - Shop Address
  - Payment Information
  - Choose Type (Retailer / Supplier)

- **Employee:**
    - Employee ID
    - Payment Information
    - Designation
    - Present Address
    - Permanent Address
- **Admin:**
    - Bank account no.
    - Permanent Address
    - Admin id

For any customer sign up, an automated confirmation code will be given to the phone number by the system. But, **for any other type of user,** an admin will verify given information and send a confirmation code for the user to confirm in order to complete creation of a new account. A unique user id will also be given to every type of user upon registering into the system, the user id for the software will consist of 3 characters followed by 6-digit unique code. 3 characters at the beginning refers to the type of user. These would be **CUS/ RET/ SUP/ EMP/ ADM.** These will be followed by an automated 6-digit code.

- **Admin Perspective:** (Admin account for bakery administration will be handed to the bakery manager with predefined user id and password). Admins can restructure the distribution of employees by changing their designations in different areas, i.e., warehouse, management.

- **Update Account:** Any kind of user can update the data of the profile through the system.

- **Password Reset:** Any user can reset their password through using OTP which will be sent to the given contact number. If password resetting turns out to be a regular habit for some users, admins will be notified and after that admin can decide if he/she wants to intervene or not.

- **Log in:** A user can log into the system by providing user id and password.

2) Inventory:

Every raw material and bakery goods will be stored in the inventory. Suppliers supply the raw materials and the employees assigned for warehouse management stores these accordingly. Also, they update the database after arrival of each stock. It will notify immediately when the bakery is running short of any ingredient. The produced goods will be stored in another section. The expiration date of all the raw materials and produced goods will be tracked all time. These will avoid over production and manage efficient cost. The ordered items will be sent from the inventory.

3) Ordering:

Ordering can be of two types. First, the bakery can make an order for the raw materials from the suppliers. Second, the customers and retailers can order according to their desire. To order, users must be logged in to the system. There

will be a menu which will include price and stock. Users can choose to order food from the menu according to their choice and should also specify the quantity. After choosing, the food will be added to the virtual cart and the amount of bill to be paid will be shown. The delivery location of the user may be set using GPS map or manually set using user input. The delivery location must be within 100 square kilometer area of the bakery store. The user can pay the bill using any authorized payment system (Described in the following section "Payment") or can also avail Cash-On Delivery (COD) service. If a user chooses to pay using an online payment system, his order will be confirmed after payment. After order confirmation, a memo will be given to the user. Users ordering before 7 pm will get their products on the day they ordered. But orders after 7 pm will be delivered on the next day. Users can cancel their order before the product has been picked up from the bakery store for delivery. After the product has been picked up from the store for delivery, the purchase will become non-refundable. The estimated amount of time for the delivery will be shown after the order has been confirmed.

4) Delivery:

User perspective:

Estimated time of food delivery will be prompted in the user's display. If a retailer orders the goods by 8PM, they will be delivered within the same day depending on stock. Later orders will be delivered the next day. No additional delivery charge is added for the retailers.

Admin perspective:

After order confirmation admins will send the delivery persons to deliver the food. System will track the location from where the user has requested it, through GPS.

If the requested location is within 100 square kilometers of the Bakery, then the delivery person will be sent to deliver food. Otherwise, the order will be cancelled.

5) Payment:

Following payment methods will be provided for the users or customers: After each order from a regular user an admin will contact the customer for order confirmation. There will be various payment options available after the confirmation.

I. **Online Payment:** A system will be integrated for online payment methods. Any logged in customer or retailer will be able to accomplish payment procedure by various online payment systems such as **Bkash, uCash, Nagad and Rocket** through **SSLCommerz**. Also, users can pay through bank account. Notification email and SMS will be sent to the users after every transaction and every transaction information will be added to the database.

II. **On-Site Payment:** User can also pay on spot if he/she goes to the bakery by himself/herself. A memo will be provided including order number.

III. **Cash-On Delivery:** User can pay cash to the delivery man if he chooses Cash on Delivery. The delivery charge should be prepaid in this case. For cancellation before an hour, they will be refunded within 3-4 working hours.

If a user fails to pay within that time, his/her reservation will be cancelled automatically.

6) Memo:

After confirmation of each order, a memo will be prompted to the user including the following information-

- Order id
- Food item, quantity with price
- Total price
- Date and time
- Discounts
- Delivery charge
- In case of home delivery, a printed copy of memo will be provided to the user through the delivery man.
- In case of on spot payment, a printed copy of memo will be given to the user hand to hand after payment.
- Also, a digital memo will be sent to the user's account and database. The demo of the memo is given in the next page.

Bakery

Order Receipt

Name: Customer X
Email: cx@gmail.com
Phone Number: 0171*******

Order No: 1
Date: 22.04.21
Time: 12:05 PM

Ordered Items

	Food Items	Quantity	Price	Total
1	Chicken Sandwich	2	150.00	300.00
2	Marble Cake (400g)	1	300.00	300.00
3	Donut	4	50.00	400.00
4	Chocolate Chip Cookies (10pcs)	1	150.00	150.00

Sub Total	1150.00
Delivery Charge	100.00
Total	1250.00

Thank You

Demo of the Order Memo

7) Database:

The database will contain three tables, one for admin, one for user and one for inventory. Delivery staff numbers will be given input every day and this will be stored in the database. When the staff will go for the deliveries, the number of available staff will be updated by the admin. After each delivery, when the staff will come back to the bakery, the admin will update the database again.

Every incoming stock and produced goods will be added to the inventory in the database. Also, the expiry date of every product will be stored there, so the admin and employees can manage the expired goods. The ordered items will be subtracted from the database.

Every order, shipment and transactions will be stored distinctly in the table. Admin will be able to see the sum of transactions happening in a day. After payment of every order, the transaction detail will be added into the admin table.

Every order memo and transaction history of the user will be stored in the user table. A replica of user info from the admin table will also be stored in the user table for security concerns. For any user info update, both tables will be updated.

8) Additional Features:

Bakery Items Menu

- If a user orders certain items frequently, there will be a suggestion tool for showing similar items and they will be prompted in the user's homepage.
- The suggestion tool will also show all the relevant food items according to the user's taste.
- After analyzing the orders of previous months from the users, the system will show the mostly ordered food items in the admin's homepage.

- An artificial intelligence tool for estimating the amount to be sold the next day and equations to maximize the profit and brand value.
- A design tool plugin for the customers to design their birthday or any occasional cakes.
- A recommendation tool for food review, rating and ongoing discounts.
- Customers will receive “Discount, Offers and Sales” notifications.
- The retailers can add the information for the expired goods for the bakery to collect the next day and replace.
- Information of reselling of the expired goods to the locals as fish feeds will be stored in the database. The buyers will receive a printed memo afterwards.

9) Refund Policy:

I. If customer doesn’t get fresh item as promised, bakery will refund the order.
II. In case of mistakes during the delivery, such as, delivering the wrong item, the bakery will again deliver the correct order.
III. Bakery will not refund or exchange in case of customer’s change of mind.
IV. The delivery charge won’t be refunded in any case.

10) Administrative Management:

1) Menu:

The menu will be fixed for a day. If any change is required, then the admin will update the menu. The availability of food items will be displayed when the users want to order.

After every order is confirmed (Online/Offline), the menu will be updated and the quantity of items confirmed will be deducted from the availability count. If any order is cancelled, the availability of the food items will also be updated.

2) Inventory Management:

Administrator will input the count of his/her stored grocery items. He/she will update counts every day. By this, he/she will be able to track which item is going to be stocked out or which item is abundantly stocked.

3) Campaign and Offers:

Admins can run some special market campaigns based on the past sales record of the database to put up sales volume and popularity of the bakery. An automated monthly campaign suggestion will be prompted to the admins where they can take the ideas from.

Requirements Modeling

At a technical level, software engineering begins with a series of modeling tasks that lead to a specification of requirements and a design representation for the software to be built. The requirements model, actually a set of models—is the first technical representation of a system. Requirements modeling uses a combination of text and diagrammatic forms to depict requirements in a way that is relatively easy to understand, and more important, straightforward to review for correctness, completeness, and consistency.

Scenario Based Modeling

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, the software team will be better able to properly characterize requirements and build meaningful analysis and design models. Hence, requirements modeling with UML begins with the creation of scenarios in the form of use cases, activity diagrams, and swimlane diagrams.

USE CASE DIAGRAM

A **use case** is a list of actions or event steps typically defining the interactions between a role (actor) and a system to achieve a goal. The actor can be a human or other external system. In this modelling, **use case diagram** is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. Use case diagrams are a blueprint for the system. Due to their simplistic nature, use case diagrams can be a good communication tool for stakeholders. The drawings attempt to mimic the real world and provide a view for the stakeholder to understand how the system is going to be designed. Use case diagrams consist of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system.

Primary Actor

Primary actors interact 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.

Level: 0

Name: Bakery Management System
Primary Actor: User, Admin
Secondary Actor: SSLCommerz, Email, UI, GPS, SMS, Bank

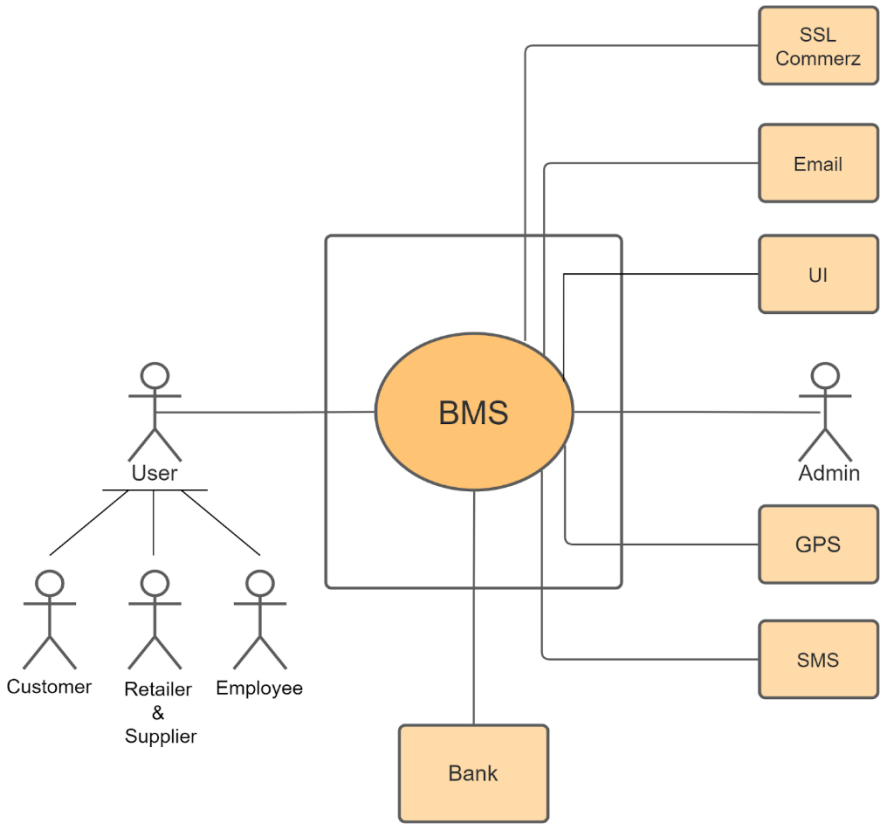


Figure 1 : Bakery Management System

Level: 1

Name: Bakery Management System (Detailed)
Primary Actor: Admin, Employee, Customer, Retailer & Supplier
Secondary Actor: SSLCommerz, Email, UI, GPS, SMS, Bank

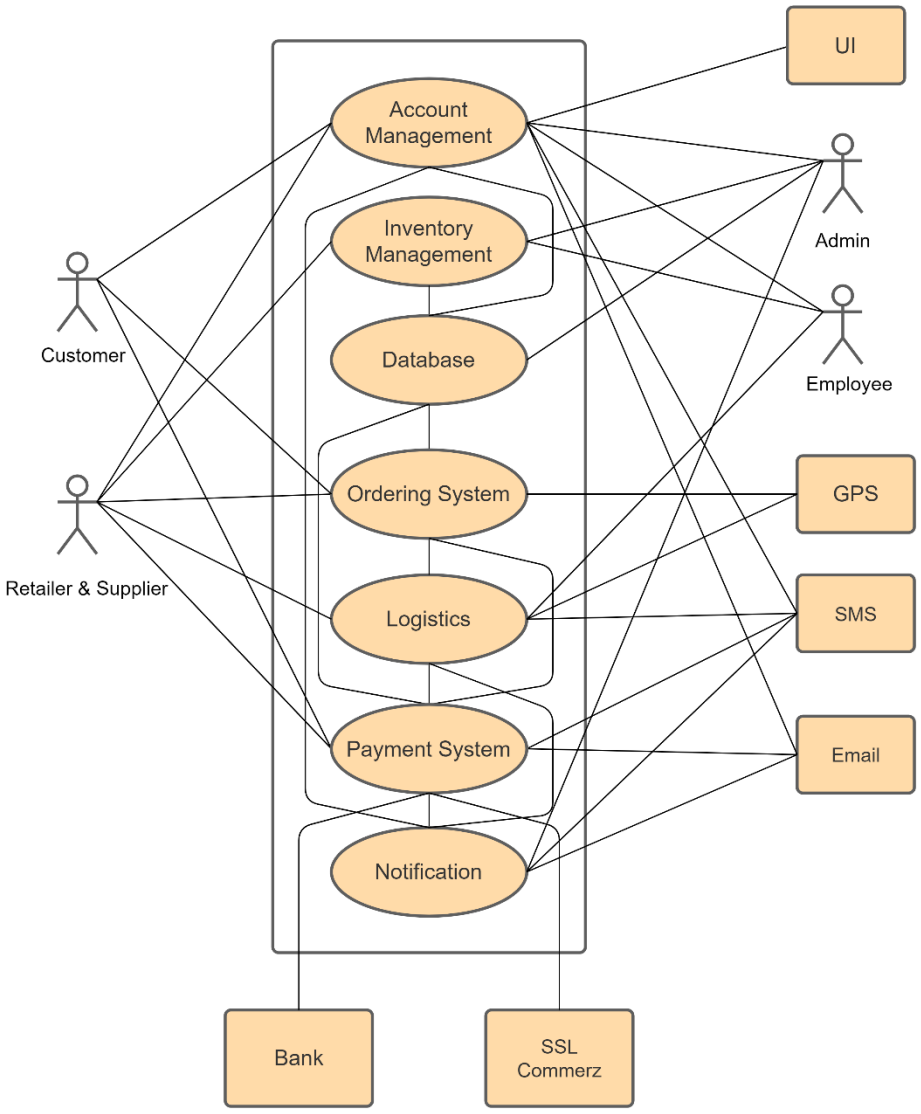


Figure 2 : Bakery Management System (Detailed)

Description of Use Case Diagram Level 1:

1. **Account Management:** A user must create an account and be logged in to enter the system. He/she can update his/her profile, reset the forgotten password. Admin will verify the credentials of the users.
2. **Inventory Management:** The raw materials for the bakery items and produced items will be stored in the inventory. Suppliers will deliver the raw materials according to the need of the bakery. The ordered items will be delivered from the inventory. Employees will be assigned for managing the stocks, tracking the expiry dates and taking care of the expired goods.
3. **Database:** Every information will be stored in the database. Inventory stocks, available employees, order, shipment, daily transactions, everything will be updated regularly. Account details of every user will also be stored in the database.
4. **Ordering System:** Customers, retailers & suppliers can login to the system and order any item from the bakery, the items will be added to their virtual cart. Suppliers can get orders from the bakery for necessary raw materials. After finishing the order, the user will proceed to the delivery and payment options. Also, anyone can cancel the order if needed.
5. **Logistics:** The ordered items will be delivered to the users. Delivery persons will be assigned for delivering every day. However, if the order is outside 100 kilometers, it will be cancelled. After delivering, a memo will be prompted to the receiver.
6. **Payment System:** User can pay through banks, or any other digital financial services such as bKash, Nagad, uCash, Rocket through SSLCommerz. A user

can also pay on spot if he/she goes to the bakery by himself/herself. Also, a user can pay cash to the delivery man if he/she chooses Cash on Delivery.

7. **Notification:** All the users will get any kind of notification by SMS and email. Customers will get verification, confirmation messages and other offers. The admin and employees will get all the updates of stock and order.

Level: 1.1

Name: Account Management
Primary Actor: Admin, Employee, Customer, Retailer & Supplier
Secondary Actor: UI, SMS

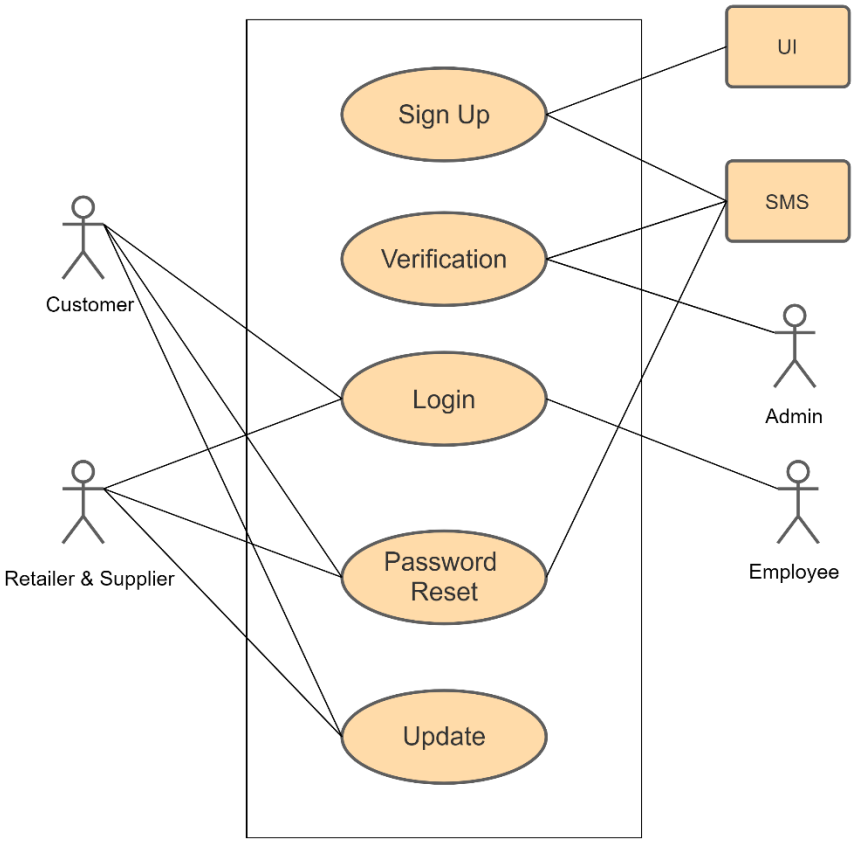


Figure 3 : Account Management

Description of Use Case Diagram level 1.1:

1. **Sign Up:** For creating an account, a user needs to provide the following common information in order to create an account: Full Name, Contact Number, Email Address, User Name, Password. After providing these, the user has to provide user type. For any customer sign up, an automated confirmation code will be given to the phone number by the system. A unique user id will also be given to every type of user upon registering into the system. Admin account for bakery administration will be handed to the bakery manager with predefined user id and password.
2. **Verification:** For any other type of user except customer, an admin will verify given information and send a confirmation code for the user to confirm in order to complete creation of a new account.
3. **Login:** User will log in to the system providing his/her user ID and password.
4. **Password Reset:** Any user can reset his/her password through using OTP which will be sent to the given contact number.
5. **Update:** Any kind of user can update the data of the profile through the system.

Action and Reply:

Action: User provides necessary information to the system.

Reply: By verifying the information given by the user, the system will create an account for his/her and give a user ID for logging in.

Action: User provides his/her unique ID and password to log into the system.

Reply: User enters the system and gains access to the features of it. For each type of user, the accessibilities will be different.

Action: User forgets his password for logging in and requests to reset the password.

Reply: The system sends an OTP to the user's contact number for resetting password.

Action: User provides new information for updating the account.

Reply: System updates the account of the user using the information.

Level: 1.1.2

Name: User Verification
Primary Actor: Admin
Secondary Actor: UI, SMS

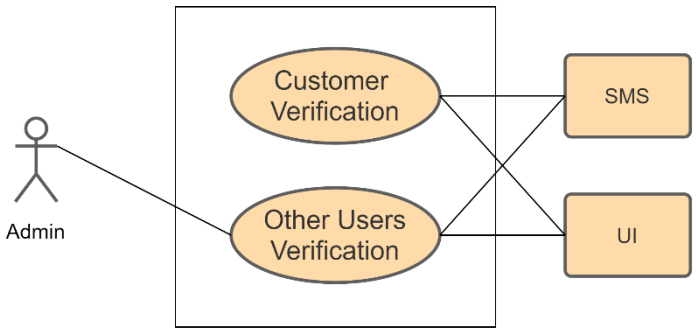


Figure 4 : User Verification

Description of Use Case Diagram level 1.1.2:

1. **Customer Verify:** The users signing up as customers will be given an OTP via SMS. By submitting this, the user will be verified.
2. **Other Users Verify:** For other users, such as employee, supplier and retailer, admin will verify their details and send an OTP to them.

Action and Reply:

Action: Users providing the user type as “Customer” receives an OTP in his/her contact number. He/she uses it for verifying the account.

Reply: The system will verify the user’s account if the OTP is correct. After verification, the account will be created.

Action: Users providing the other user types except “Customer” will be verified by the admin. Admin will verify the credentials such as, TAN, employee ID. After this, an OTP will be sent to the user for completing the verification process.

Reply: The system will verify the account by checking the OTP. Thus, the account will be created after this.

Level: 1.2

Name: Inventory Management
Primary Actor: Admin, Employee
Secondary Actor: None

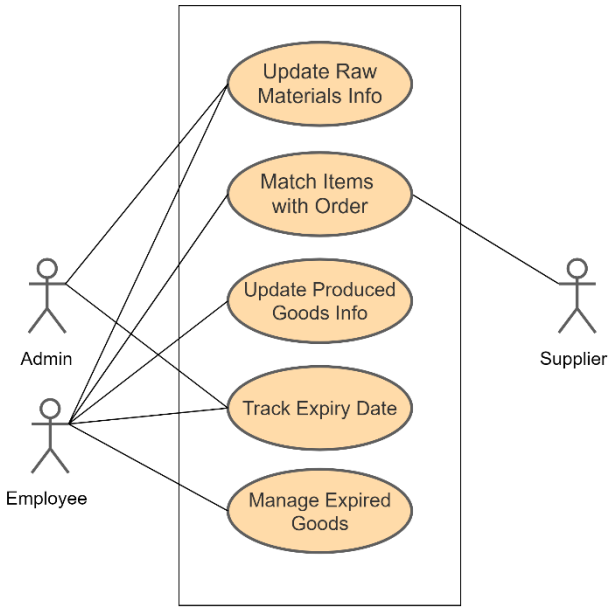


Figure 5 : Inventory Management

Description of Use Case Diagram Level 1.2:

1. **Update Raw Materials Info:** To update raw materials info, an employee will log in to the system and ask for permission to update info. The admin will grant access to that specific admin to update the raw materials info and can inspect the updated info of the raw materials. The admin can deny access to that employee for updating whenever necessary.
2. **Update Produced Goods Info:** To update produced goods info, an employee will update the number of produced goods everyday just after the items are ready for sale.

3. **Track Expiry Date:** The admin and the employees will be able to see the expiry date in the system and also the exact time left for the item to expire (in days, months etc.). When the expiry date is nearby, the admin and the employees will be notified so that they can manage it.
4. **Manage Expired goods:** Once an item is expired, the employees will be notified of the expiry and they will manage that item. After this, they will update the system and the expired product will no longer be shown.
5. **Send ordered items for delivery:** Before sending out an ordered item for delivery, an employee will mark that item sent for delivery and the inventory will be updated.

Action and Reply:

Action: After getting the supplies of the raw materials, admin and employees will add the data to the inventory.

Reply: The details of the raw materials will be updated.

Action: When the items are ready for sale, the number of produced goods will be added to the inventory.

Reply: The details of the produced goods will be updated.

Action: Track the expiry date of all the items.

Reply: The system will notify when the expiry date of any item is nearby.

Action: Admin and the employees will be able to see the expired goods list.

Reply: The employees will manage the expired goods, either by selling as fish feeds, or by discarding them.

Level: 1.3

Name: Database
Primary Actor: Admin, Employee
Secondary Actor: None

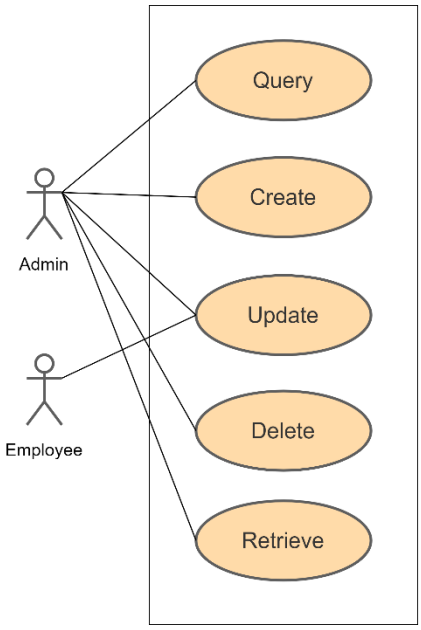


Figure 6 : Database

Description of Use Case Diagram Level 1.3:

1. **Admin Table:** The admin table will contain all the data related to the admin which will be processed for the admin.
2. **User Table:** The user table will contain all information related to the user.
3. **Inventory Table:** The inventory table will contain all the data of the inventory.

Level: 1.3.1

Name: Query
Primary Actor: Admin
Secondary Actor: None

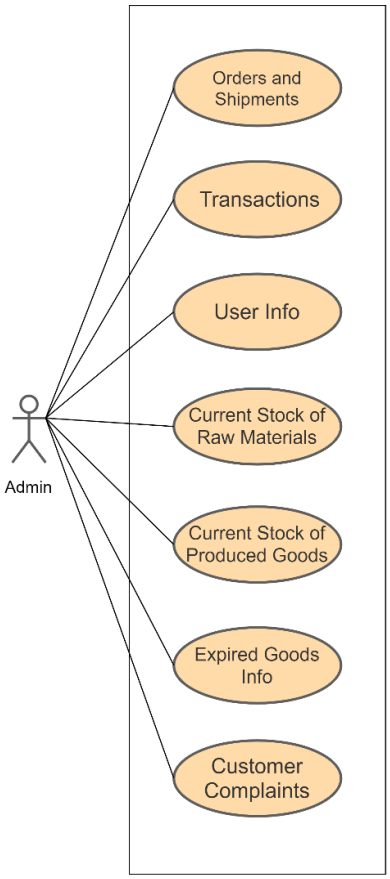


Figure 7 : Query

Description of Use Case Diagram Level 1.3.1:

1. **Add Admin:** Any one of current admins can add another admin to the system. While adding an admin, the info of the new admin will be filled by taking input from the old admin.

2. **Remove Admin:** Any one of the current admins can remove an admin from the system. After removing, the info of the removed admin will be deleted from the system.
3. **Update On-Duty Employees Info:** The employee table will contain the data of on-duty employees and only the admin will be able to see the data. The employees can only fill out or update their own data.
4. **Track Orders and Shipments:** The admin table will contain data of all the orders including the delivered orders, location of the order, order status (delivered or pending), payment amount and customer name. The admin will supervise the orders, the employees will be able to access all the pending orders and the customer will be able to track his/her own orders.
5. **Handle Daily Transactions:** The admin table will contain information of all the transactions of the business such as transactions with the retailer and supplier and with the customers. Each transaction will contain the information of products, bill amount and name of the user. The admin can access all the transaction information and the customers or the retailer and supplier can only access their own transaction. Admin will be able to see the sum of transactions happening in a day. After payment of every order, the transaction detail will be added into the admin table.
6. **User Info:** The information of all kinds of users will be stored in the admin table. The user info will include user name, user type (customer, employee, retailer and supplier), a unique user id, date of birth of the user and phone number. A user will fill out his/her info and can update the provided info whenever necessary.

7. **Customer Complaints:** All the complaints from the customer will be stored in customer complaints with customer id and complaint statements. Only the admin will be able to access all these complaints. Each customer will have access to his/her complaint he/she filed (Updating the status of the complaint or marking it as solved, necessary communication with the admin to resolve the complaint).

Action and Reply:

Action: Existing admin will add information about the new admin for adding him.
Reply: The database will be updated and an admin will be added to the system.

Action: Any admin can remove another admin if necessary.
Reply: The database will be updated by removing the admin.

Action: Admin can see the data of the employees and assign them accordingly. Employees can update their info after completing any task.
Reply: The database will be updated by assigning the employees, available employees and on-duty employees.

Action: The admin will supervise the orders. The employees will be able to access all the pending orders and the customer will be able to track his/her own orders.
Reply: The system will show all the order status. On the customer accounts, the orders made by them will be shown too.

Action: Admin can view the previous transactions, daily transactions with all the details.
Reply: After billing and payment of every order, the transaction detail will be added into the admin table.

Action: Users can insert and update their info in their account.
Reply: All the information of the users will be updated in the database.

Action: Customers can file any complaint if they want.
Reply: The complaints will be added to the database and the admin will be able resolve them.

Level: 1.3.2

Name: Create
Primary Actor: Admin
Secondary Actor: None

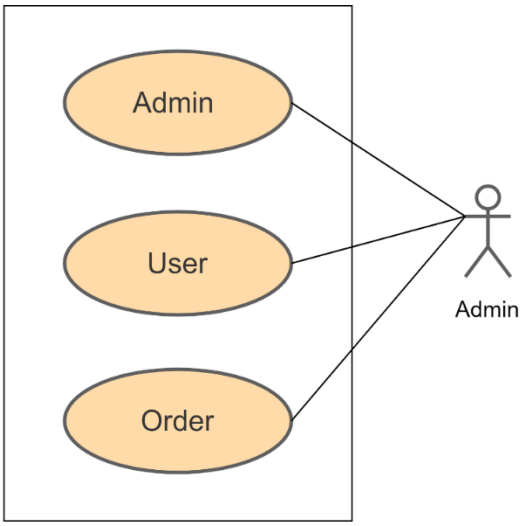


Figure 8 : Create

Description of Use Case Diagram Level 1.3.2:

1. **Order Memo:** Every order memo and transaction history of the user will be stored in the user table. Each user will be able to see all their order memos.
2. **Transaction History:** A copy of all the transactions from the admin table will be stored in the user table and each user will see his own transaction history which contains all his past transactions. For any new transactions, the transaction information will be added to both the tables.
3. **User Info:** A replica of user info from the admin table will also be stored in the user table for security concerns. For any user info update, both tables will be updated.
4. **Order history:** All order information including ordered items, order bill, user name and user id will be stored in the order history. A user will only see all his own orders.
5. **Order Status:** The order status of each user will be stored. The order status may be any one of the following: no orders, order pending, waiting for delivery. The order status for the user will be updated when he makes an order, then the system accepts it and finally it is delivered.

Action and Reply:

Action: Every order memo and transaction history will be added to the database.
Reply: Every user can see their orders, with the memo and transaction details.

Action: The status of the orders will be stored in the database.
Reply: Users can see the status of all the orders, and track the ongoing and pending orders easily.

Level: 1.3.3

Name: Update
Primary actor: Admin, Employee
Secondary actor: None

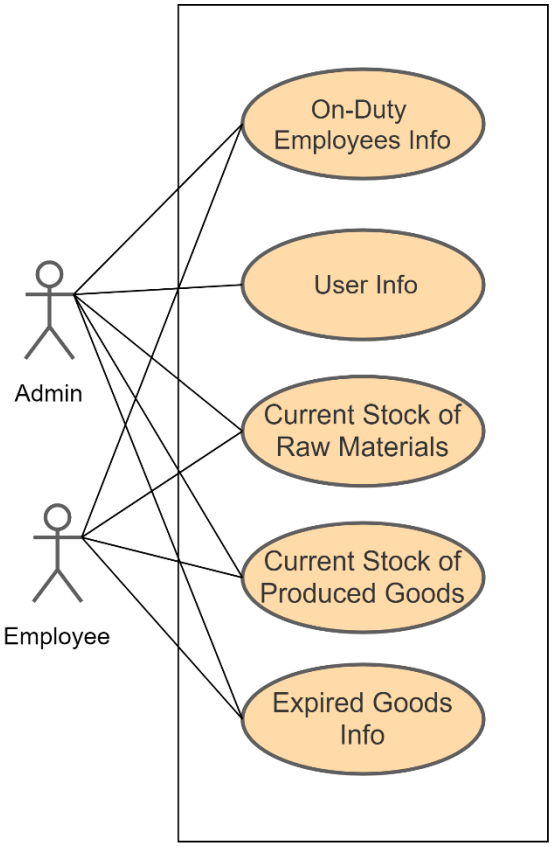


Figure 9 : Update

Description of Use Case Diagram level 1.3.3:

1. **Current Stock of Raw Materials:** Admins and authorized employees update raw materials stock with every shipment and keep track of raw material stock.

2. **Current Stock of Produced Goods:** Customers can see which products are available and in stock in a given time. Customers can make purchase decisions based on given product stocks. Admins and authorized employees can see and update stock of produced goods.
3. **Expired Goods Info:** Admins and Employees have access to expired goods info, so they can track and differentiate products that are not suitable for sale anymore.

Action and Reply:

Action: Suppliers supply raw materials to the bakery and employees add these to the inventory.

Reply: The inventory table will be updated after each incoming stock and show available stock.

Action: After producing the goods, the admin and the employees will update the database.

Reply: System will show all the details of the produced goods, currently available goods.

Action: Admins and employees can access the details of the expired goods for managing them.

Reply: System will show the number of expired goods. After discarding them, the database will be updated.

Level: 1.4

Name: Ordering System
Primary actor: Admin, Supplier, User
Secondary actor: GPS

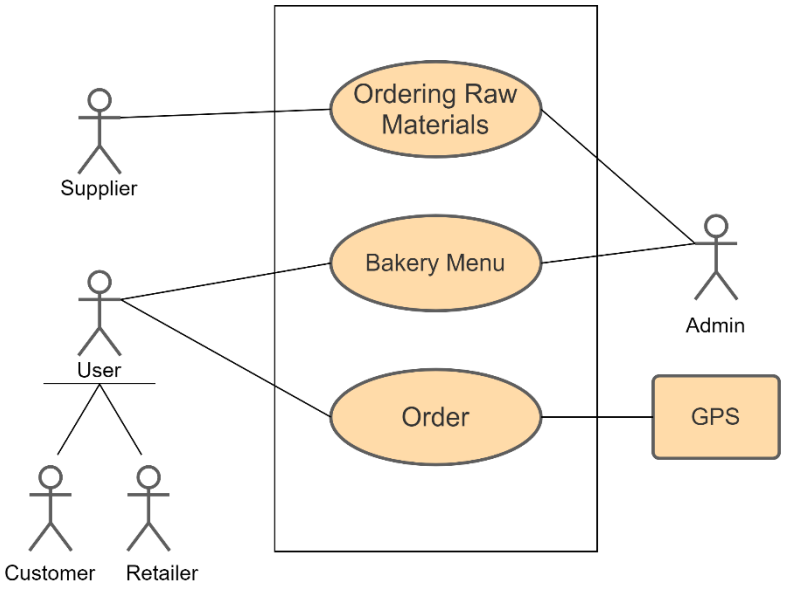


Figure 10 : Ordering System

Description of Use Case Diagram level 1.4:

1. **Order Raw Material:** Admin can order raw materials and track orders. Supplier will get order through the system and supply.
2. **Bakery Menu:** Users can navigate through the system using the bakery menu.
3. **Choose Items and Quantity:** Users can choose any desired items in desired quantity as long as there's enough stock.

4. **Cancel Order:** In case of change of mind, users can cancel the order as long as the order has not been shipped. Admins can approve or decline cancellation requests.
5. **Set Delivery Location:** Users have to set a delivery location in case of home delivery, the GPS will track their location and save it, in case the user wants the GPS to save location.

Action and Reply:

Action: Admin will order raw materials from the suppliers.

Reply: Order will be sent to the supplier for delivering the raw materials.

Action: A bakery menu will be available for the users for choosing the items for ordering.

Reply: Bakery menu will always be updated according to the availability of the products.

Action: Users can choose the desired items and their quantity.

Reply: The items with their quantity will be added to the virtual cart.

Action: User requests cancellation of the order.

Reply: If the order has not been shipped yet, it will be cancelled, otherwise not.

Action: Users set a delivery location using GPS for home delivery.

Reply: The GPS will track the location and proceed for delivering.

Level: 1.5

Name: Logistics
Primary actor: Admin, Customer, Retailer, Supplier, Employee
Secondary actor: GPS, SMS

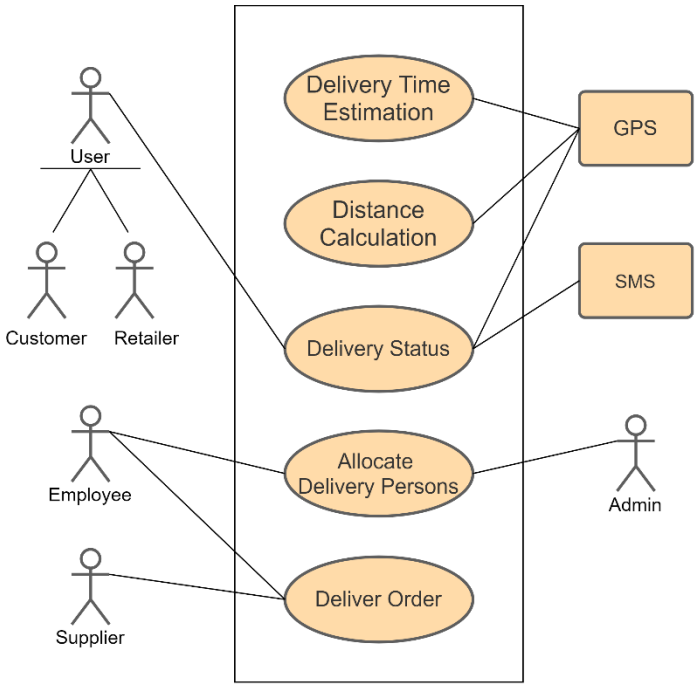


Figure 11 : Logistics

Description of Use Case Diagram level 1.5:

1. **Delivery Time Estimation:** GPS will calculate estimated delivery time based on user location and traffic in the area and show that to the user.
2. **Distance Calculation:** GPS will calculate distance between user and bakery.
3. **Delivery Status:** Once a user places an order, the user will be sent an SMS confirming the order and the user can check delivery status on his/her account, GPS will keep track of the delivery.

4. **Allocate Delivery Person:** After an order is placed by a user, an admin will allocate suitable delivery personnel for the delivery**.**
5. **Deliver Order:** Supplier will deliver the required raw material on a daily basis and the supply history will be kept. Employees will deliver the orders to the customers according to their location.

Action and Reply:

Action: User will set the location using the GPS.

Reply: GPS will show the estimated time and distance for delivery.

Action: User will confirm the order.

Reply: The system will show the delivery status and GPS will keep track of the delivery.

Action: Admin will allocate delivery personnel.

Reply: Employees will be allocated for delivering the orders.

Action: Suppliers will get orders from the bakery, customers and retailers will order according to their desire.

Reply: Suppliers will deliver the raw materials and employees will deliver the orders to the destinations.

Level: 1.6

Name: Payment
Primary actor: Admin, Customer, Retailer, Supplier, Employee
Secondary actor: SMS, GPS, SSLCommerz, Bank

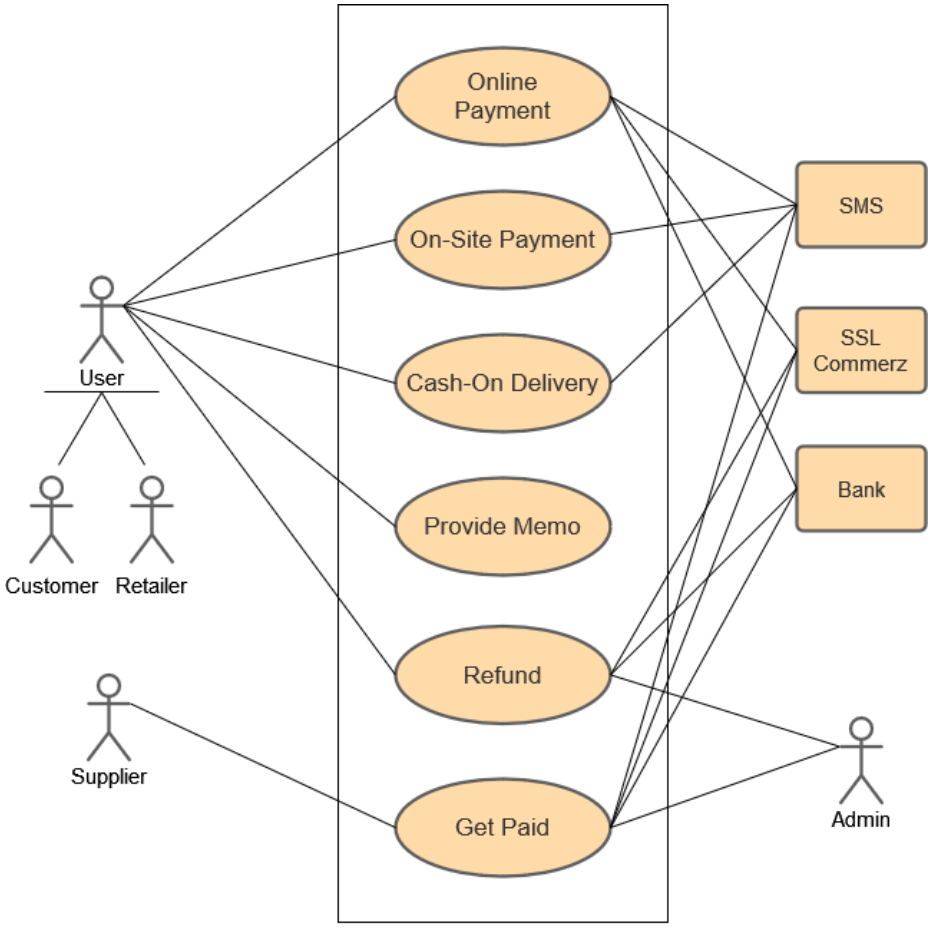


Figure 12 : Payment

Description of Use Case Diagram level 1.6:

1. **Online Payment:** Users can choose to pay online on any order via SSLCommerz. Users will be sent an SMS to notify about the transaction in

that case. Relevant banks will also be notified about the transaction in order to update users account details.

2. **On-Site Payment:** If a user wishes to pay on site, after payment the employee will input the information to the system and an SMS will notify the full transaction.
3. **Cash-On Delivery:** In case of home delivery, users can choose “Cash-On Delivery”. Users will be sent an SMS right after the order has been placed, notifying them about the exact amount the user owes. Another SMS will be sent when the delivery is complete and the payment has been done.
4. **Provide Memo:** A transaction receipt or memo will be provided after every transaction, online, on-site or cash on delivery.
5. **Refund:** In cases where necessary, users can apply for refund for their product. Refund requests will go through admins and an admin must approve the refund first. Once the request has been approved, the user will be notified by an SMS. The bakery will transfer the refunded money to the user's bank account through SSLCommerz.
6. **Get Paid:** Suppliers will receive payment after delivery of the materials are done. Admin will approve the payment. The money owed will be sent to the suppliers' accounts through SSLCommerz. Suppliers will be notified that they have received their payment once it’s done.

Level: 1.6.1

Name: Online Payment
Primary actor: Customer, Retailer
Secondary actor: SMS, SSLcommerz, Bank

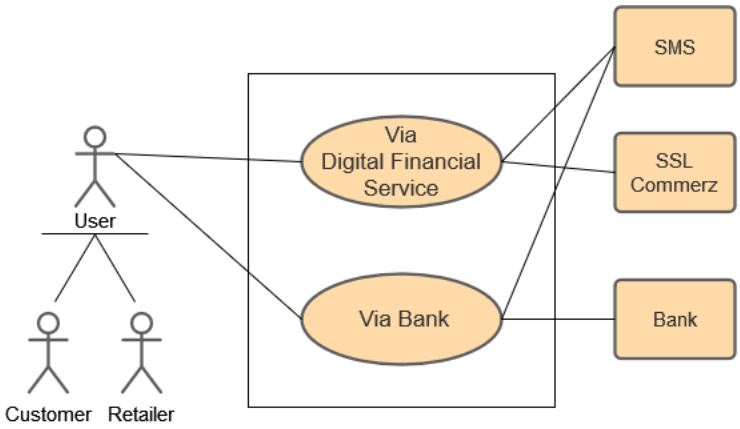


Figure 13 : Online Payment

Description of use case diagram level 1.6.1:

1. **By Digital Financial Service:** Any logged in customer or retailer will be able to accomplish payment procedure by various digital financial services such as Bkash, uCash, Nagad and Rocket. A notification SMS will be sent to the users after every transaction and every transaction information will be added to the admin database.
2. **By Bank:** Any logged in customer or retailer will also be able to accomplish payment procedure by various online banking systems. A notification SMS will be sent to the users after every transaction and every transaction information will be added to the admin database.

Action and Reply:

Action: User will choose any of the digital financial services for online payment.

Reply: The transaction will be done by the account chosen by the user.

Action: User will choose to pay via an online banking system.

Reply: The payment will be completed by deducting from the bank account.

Level: 1.6.5

Name: Refund
Primary actor: Admin, Customer, Retailer
Secondary actor: SMS, SSLcommerz, Bank

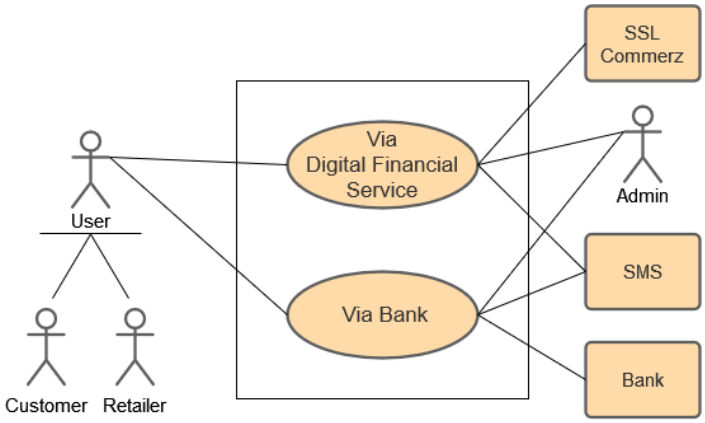


Figure 14 : Refund

Description of use case diagram level 1.6.5:

1. **By Digital Financial Service:** In case of any order cancellation, the customer or retailer will be refunded by various digital financial services such as bKash, uCash, Nagad and Rocket. Admin will approve the payment. A notification

SMS will be sent to the users after every transaction and every transaction information will be added to the database.

2. **By Bank:** In case of any order cancellation, the customer or retailer can also be refunded by various online banking systems. Admin will approve the payment. A notification SMS will be sent to the users after every transaction and every transaction information will be added to the database.

Action and Reply:

Action: Users will cancel the order if they desire.

Reply: Users will be refunded via the chosen digital financial service or bank account.

Level: 1.6.6

Name: Get Paid
Primary actor: Admin, Supplier
Secondary actor: SMS, SSLcommerz, Bank

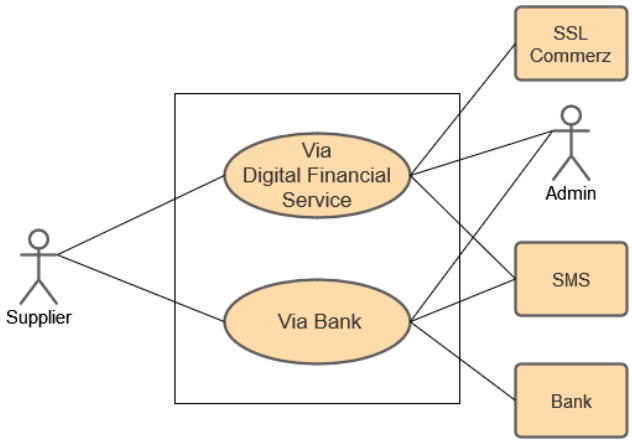


Figure 15 : Get Payment

Description of use case diagram level 1.6.6:

1. **By Digital Financial Service:** After delivery of the material has been done, suppliers will receive payment by any of the digital financial services such as bKash, uCash, Nagad and Rocket. Admin will approve the payment. A notification SMS will be sent to the supplier after the payment is done and every transaction information will be added to the database.

2. **By Bank:** After delivery of the material has been done, suppliers will receive payment by online banking system. Admin will approve the payment. A notification SMS will be sent to the supplier after the payment is done and every transaction information will be added to the database.

Action and Reply:

Action: Users will cancel the order if they desire.

Reply: Users will be refunded via the chosen digital financial service or bank account.

Level: 1.7

Name: Notification
Primary actor: Admin
Secondary actor: SMS, Email, User

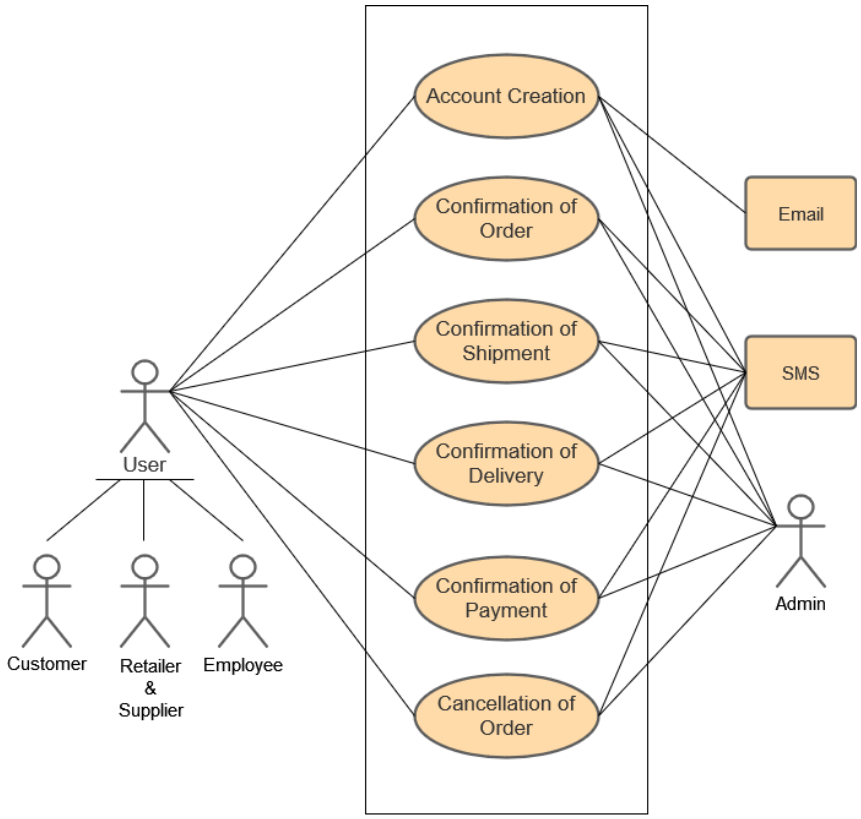


Figure 16 : Notification

Description of use case diagram level 1.7:

1. **Account Creation:** After successful creation of a user account, the user will be notified about this by SMS and Email.
2. **Confirmation of Order:** When user will confirm an order, a SMS will be sent regarding this.

3. **Confirmation of Shipment:** When the home deliveries will be shipped, the associated users will be notified.
4. **Confirmation of Delivery:** After delivering the order, users will be notified.
5. **Confirmation of Payment:** After any type of payment, the user will receive an SMS.
6. **Cancellation of Order:** If any user cancels the order, he/she will be notified.

Action and Reply:

Action: User will create an account by providing credentials.

Reply: User will receive an OTP and a confirmation message after account creation.

Action: User will make an order.

Reply: A confirmation message will be sent after making an order.

Action: User will make a Home Delivery.

Reply: When the order of the user will be shipped, a message regarding this will be sent to him/her.

Action: User will make a Home Delivery and wait for it.

Reply: When the order will be delivered, a completion message will be sent to the user.

Action: User will pay for the order.

Reply: A message will be sent after the confirmation of the payment.

Action: User can cancel order if needed.

Reply: User will get a message regarding the cancellation.

Activity Diagram

Activity diagram is an important behavioral diagram in UML diagram to describe dynamic aspects of the system. Activity diagram is essentially an advanced version of flow chart that models the flow from one activity to another activity.

Level: 1

Name: Bakery Management System

Reference: Use Case Diagram Level – 1

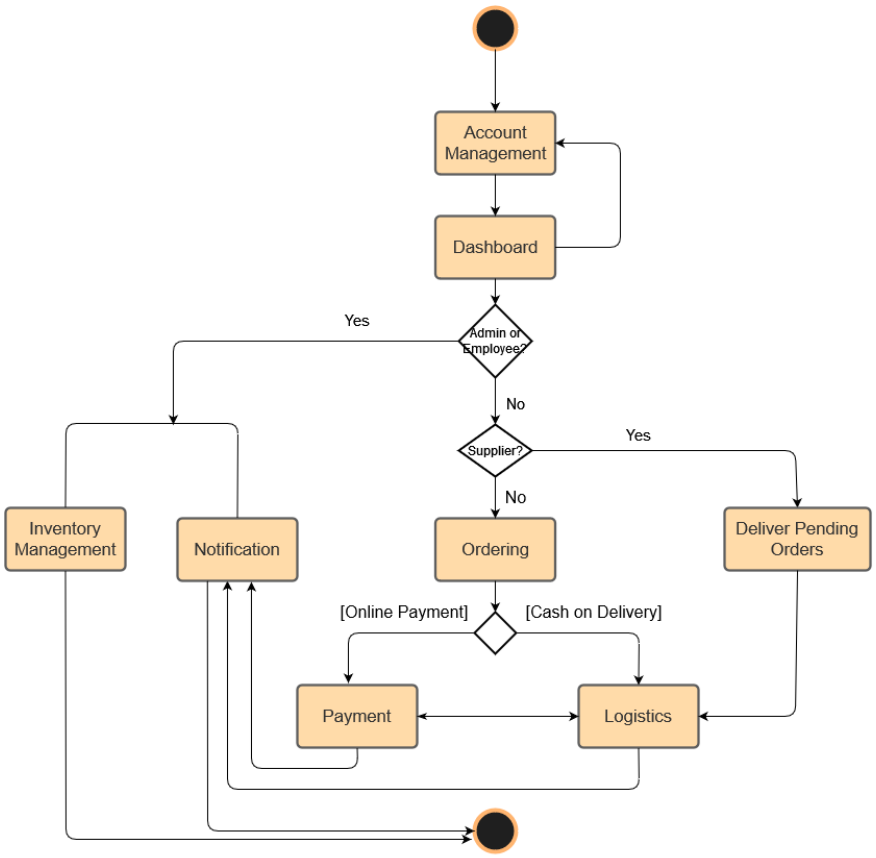


Figure 17 : Bakery Management System (Use Case Level - 1)

Description of activity diagram level 1:

1. **Account Management:** Any user can login if already has an account, otherwise create a new account by providing the credentials. Then those will be verified and an account will be created.
2. **Dashboard:** Logged in user will be directed to dashboard. He/she can log out from there. If the user is admin or employee, he/she will go to the inventory management section, if he/she is a supplier, then view the pending orders. Lastly, the customers can order from dashboard.
3. **Inventory Management:** Admin and employees will keep record of the stocks and raw materials here.
4. **Ordering:** If the customer chooses online payment, then he/she will have to pay first. Then the order will be proceeded to logistics section.
5. **Deliver Pending Orders:** Supplier will supply the pending orders to the bakery by logistics.
6. **Payment:** User will pay through any online payment system (e.g., bKash, Nagad).
7. **Logistics:** All the orders will be proceeded for delivery in this section.

Level: 1.1

Name: Account Management

Reference: Use Case Diagram Level – 1.1

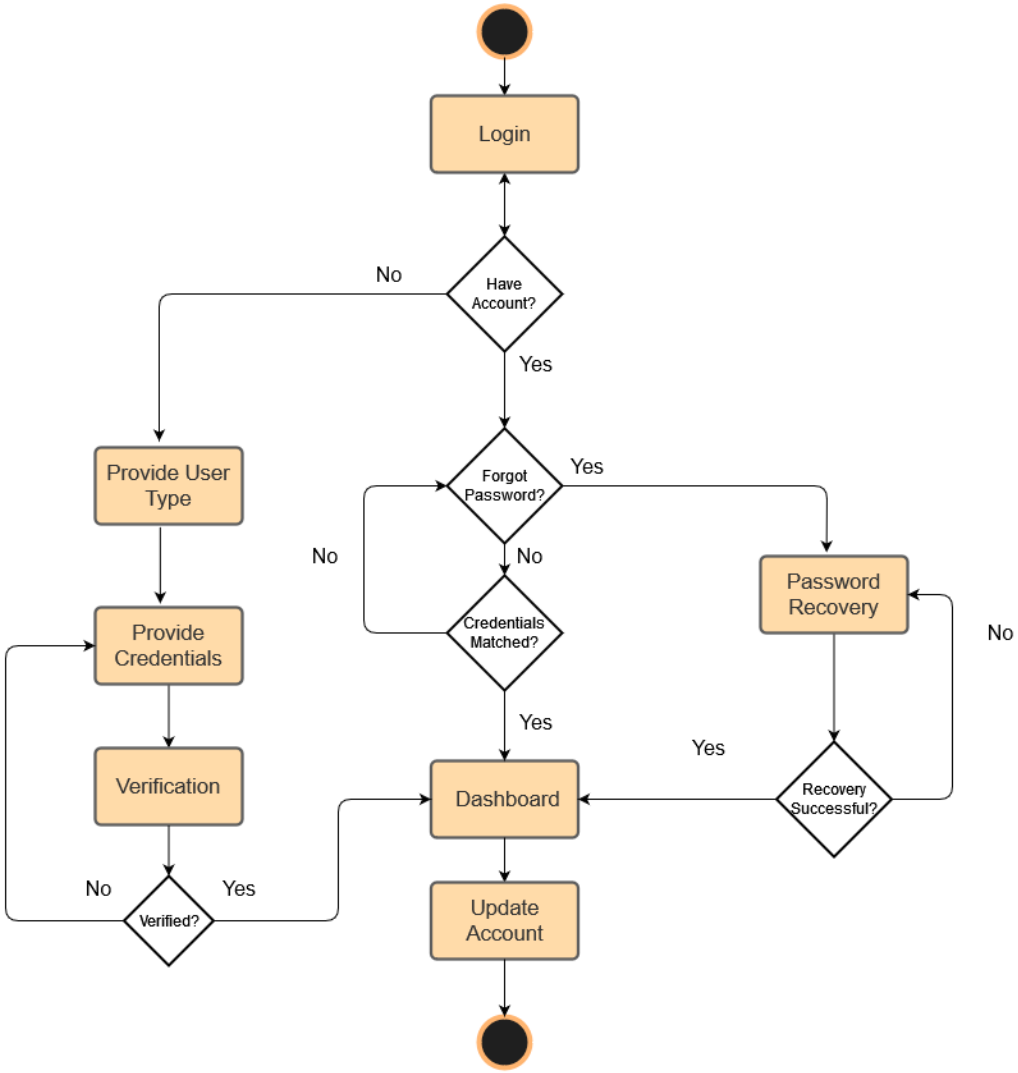


Figure 18 : Account Management (Swim Level - 1.1)

Description of activity diagram level 1.1:

1. **Login:** The users having an account can login by providing the correct credentials.
2. **Password Recovery:** If the user forgets his/her password, then an OTP will be sent for resetting the password. The user can login with that afterwards.
3. **Provide User Type:** If a user wants to create an account, firstly he/she has to provide the user type (e.g., Admin, Employee, Customer).
4. **Provide Credentials:** After entering user type, he/she has to provide the required credentials.
5. **Verification:** The credentials will be verified, customers will get an OTP and the information of retailers and suppliers will be verified by admin.
6. **Dashboard:** Once the user is verified, he/she will be directed to the dashboard. But if the user isn't verified, he/she again has to provide the credentials.
7. **Update Account:** Any logged in user can update the account.

Level: 1.1.2

Name: User Verification

Reference: Use Case Diagram Level – 1.1.2

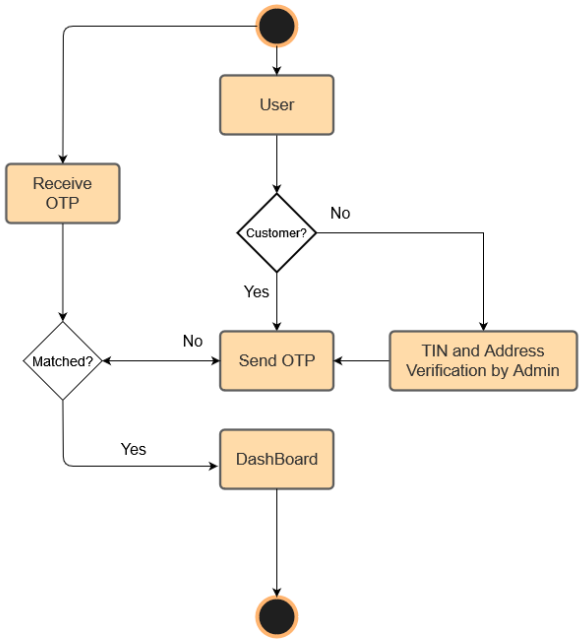


Figure 19 : User Verification (Use Case Level - 1.1.2)

Description of activity diagram level 1.1.2:

1. **User:** Any type of user will come to the verification section.
2. **Send OTP:** If the user is customer, an OTP will be sent to him.
3. **TIN and address Verification:** The credentials will be verified by admin.
4. **Receive OTP:** The user has to input the OTP provided to him.
5. **Dashboard:** If the OTP matches, user will be sent to dashboard. Otherwise, another OTP will be given.

Level: 1.2

Name: Inventory Management
Reference: Use Case Diagram Level – 1.2

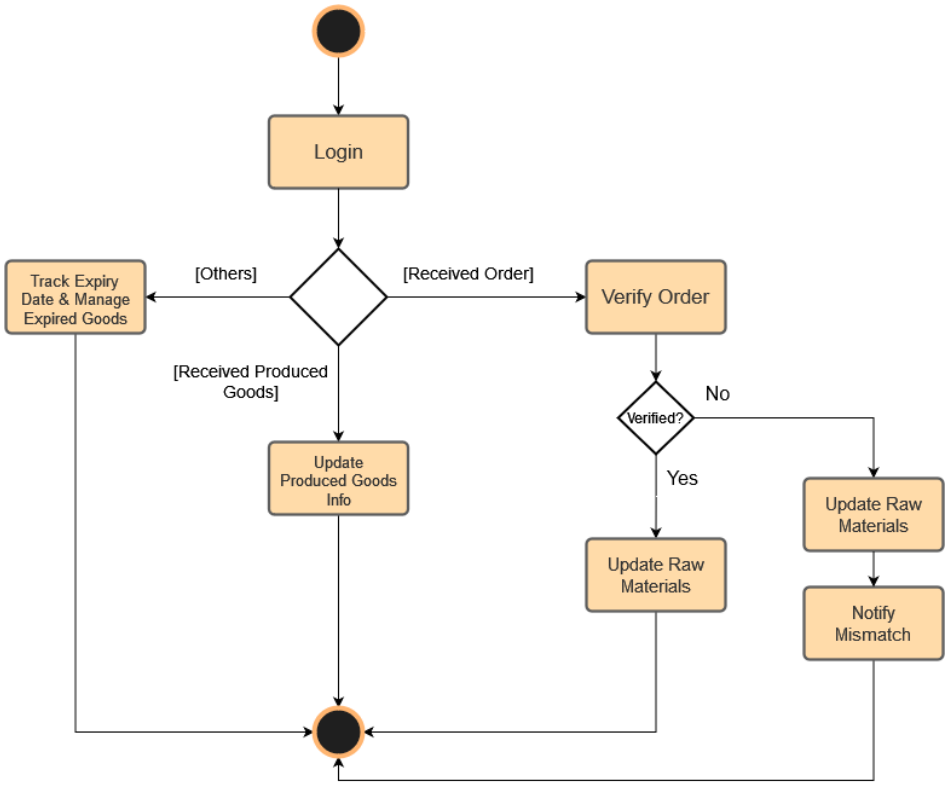


Figure 20 : Inventory Management (Use Case Level - 1.2)

Description of activity diagram level 1.2:

1. **Verify Order:** For any received order, the employees will verify it.
2. **Notify Mismatch:** If the supplied materials have a mismatch, the supplier will be notified about it.
3. **Update Raw Materials:** Once the order is verified or has a mismatch, the number and information of the raw materials in the inventory will be updated.

4. **Update Produced Goods Info:** Every time the bakery produces goods, the information related to those will be updated.
5. **Track Expiry Date and Manage Expired Goods:** The admin and employees will track the expiry dates of the goods stored in the inventory. Also, manage the goods that are expired.

Level: 1.3

Name: Database
Reference: Use Case Diagram Level – 1.3

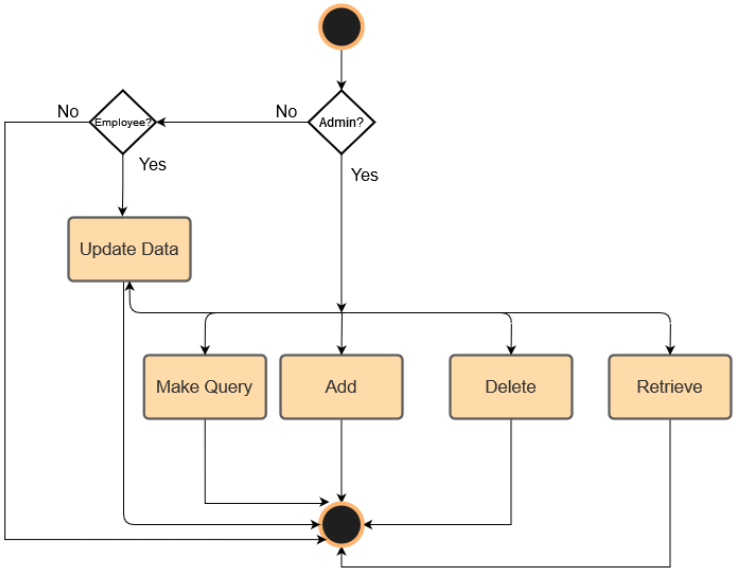


Figure 21 : Database (Use Case Level - 1.3)

Description of activity diagram level 1.3:

1. **Make Query:** Admin can make query for getting any type of information.
2. **Add:** The admin can add new data to the database.
3. **Delete:** The admin can delete unnecessary data from the database.

4. **Retrieve:** Admin will have an option to retrieve the lost or mistakenly deleted data.
5. **Update Data:** Employees only have access for updating the data. So, the employees and admin can update the stock and material information from time to time.

Level: 1.3.1

Name: Query
Reference: Use Case Diagram Level – 1.3.1

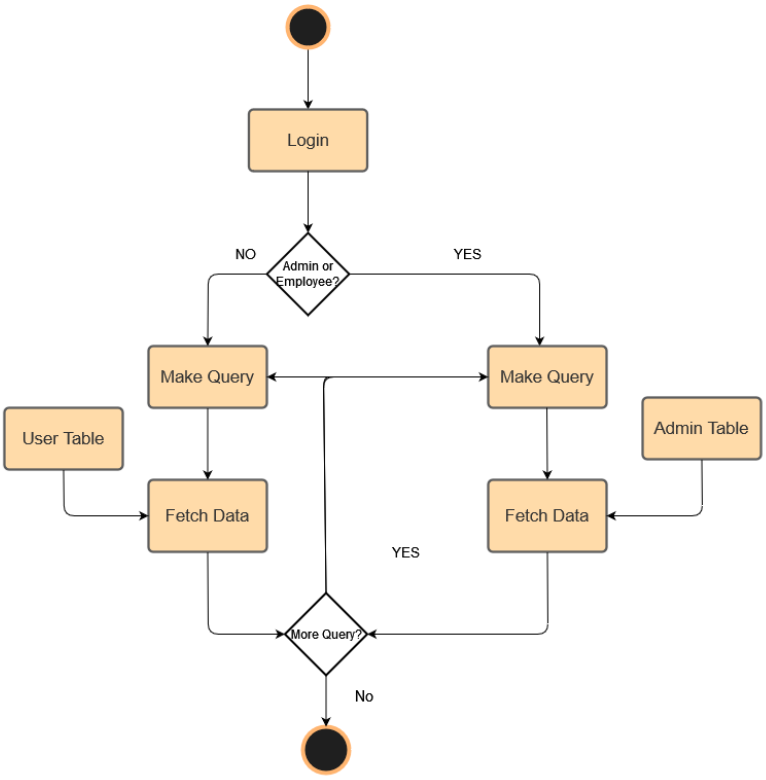


Figure 22 : Query (Use Case Level - 1.3.1)

Description of activity diagram level 1.3.1:

1. **Make Query:** An admin or employee can make query on the database for knowing any type of information.
2. **Fetch Data:** The data based on the query will be fetched.

3. **Admin Table:** The data requested by admin will be sent from admin table.
4. **Employee Table:** The data requested by admin will be sent from employee table.

Level: 1.3.2

Name: Create
Reference: Use Case Diagram Level – 1.3.2

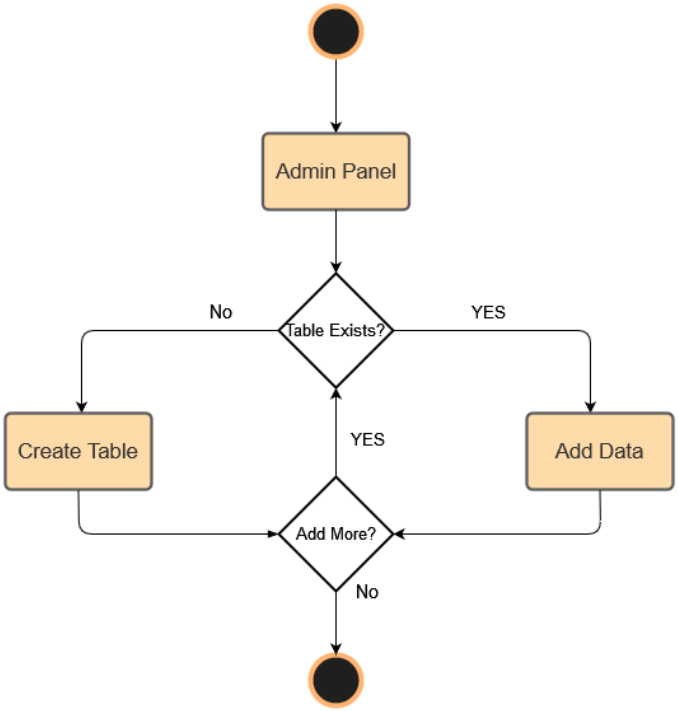


Figure 23 : Create (Use Case Level - 1.3.2)

Description of activity diagram level 1.3.2:

1. **Admin Panel:** Only the admin can create table or add data.
2. **Add Data:** The admin can add new data to the database every time.
3. **Create Table:** If the table doesn't exist, the admin can create a new table.

Level: 1.3.3

Name: Update

Reference: Use Case Diagram Level – 1.3.3

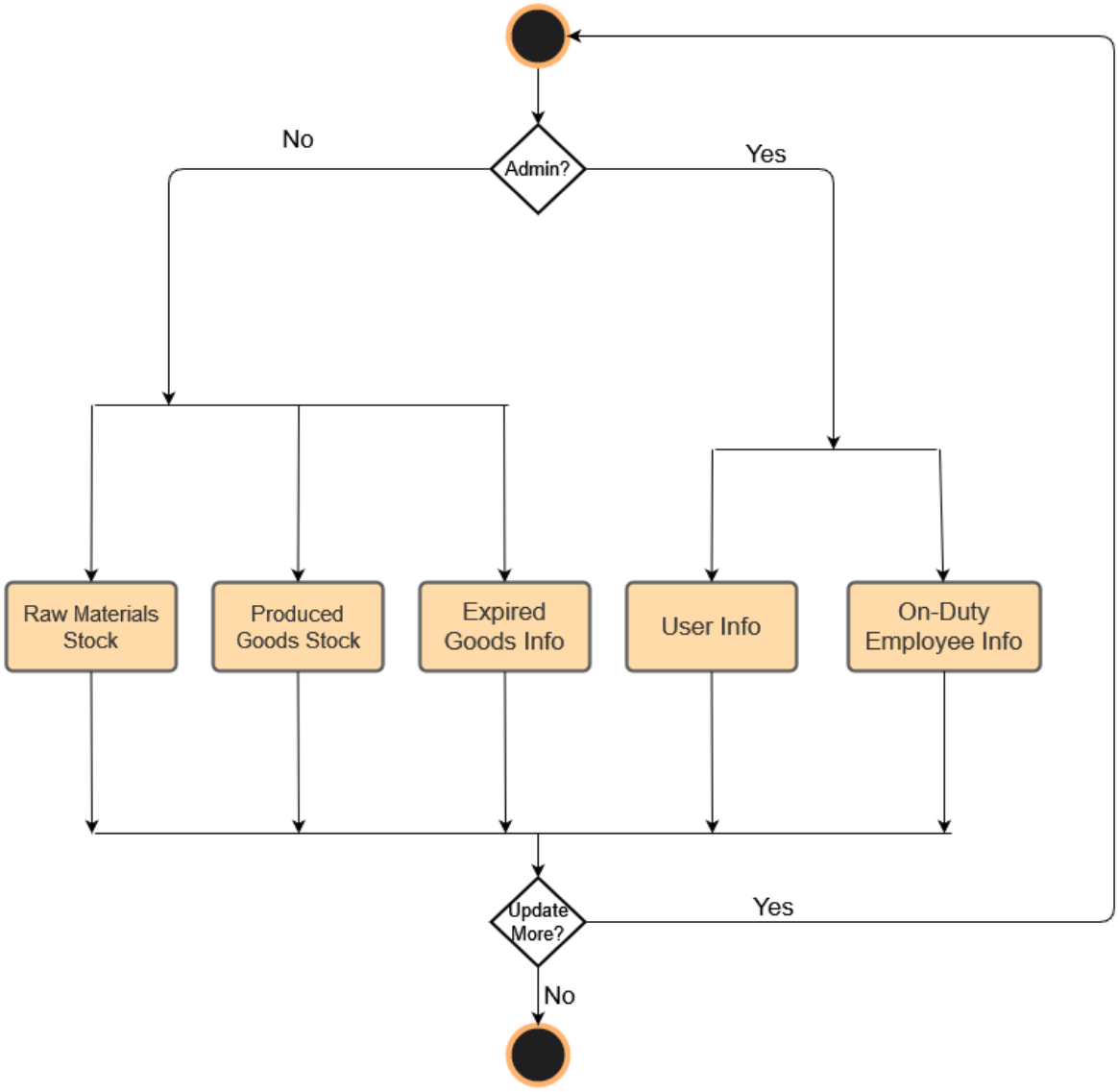


Figure 24 : Update (Use Case Level - 1.3.3)

Description of activity diagram level 1.3.3:

1. **User Info:** Admin can update the user information if any user creates or updates account.
2. **On-Duty Employee Info:** The admin can update the responsibilities of the employees every day.
3. **Raw Materials Stock:** The employees can update the number of raw materials after using or getting supply from the suppliers.
4. **Produced Goods Stock:** Every time a customer orders, or the bakery produces the items, the employees will update the stock in the inventory table of the database.
5. **Expired Goods Info:** The employees will always check if any good is expired and update the information regarding those.

Level: 1.4

Name: Ordering System

Reference: Use Case Diagram Level – 1.4

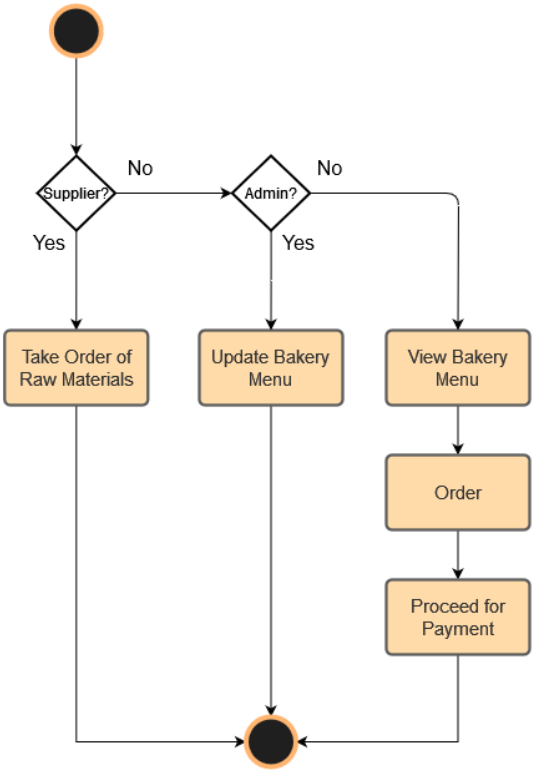


Figure 25 : Ordering System (Use Case Level - 1.4)

Description of activity diagram level 1.4:

1. **Take Order of Raw Materials:** Suppliers can take the orders of raw materials given by the admin.
2. **Update Bakery Menu:** The admin can update the bakery menu every day.
3. **View Bakery Menu:** Other users can view the menu to order items.
4. **Order:** Users can order from the menu.
5. **Proceed for Payment:** After ordering, the users will proceed for making payment.

Level: 1.5

Name: Logistics

Reference: Use Case Diagram Level – 1.5

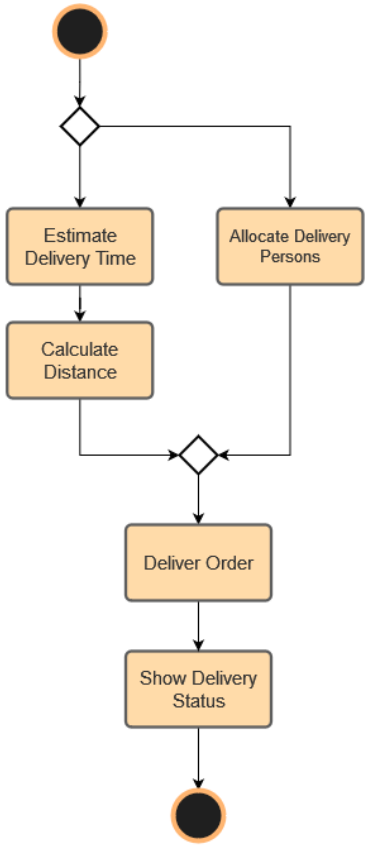


Figure 26 : Logistics (Use Case Level - 1.5)

Description of activity diagram level 1.4:

1. **Estimate Delivery Time:** Estimate time by getting the delivery location.
2. **Allocate Delivery Persons:** The admin allocates delivery persons every day.
3. **Calculate Distance:** Calculate distance from bakery to the location.
4. **Deliver Order:** After the above steps, the orders will be sent to deliver.
5. **Show Delivery Status:** Show the status for shipment, location and completion.

Level: 1.6

Name: Payment

Reference: Use Case Diagram Level – 1.6

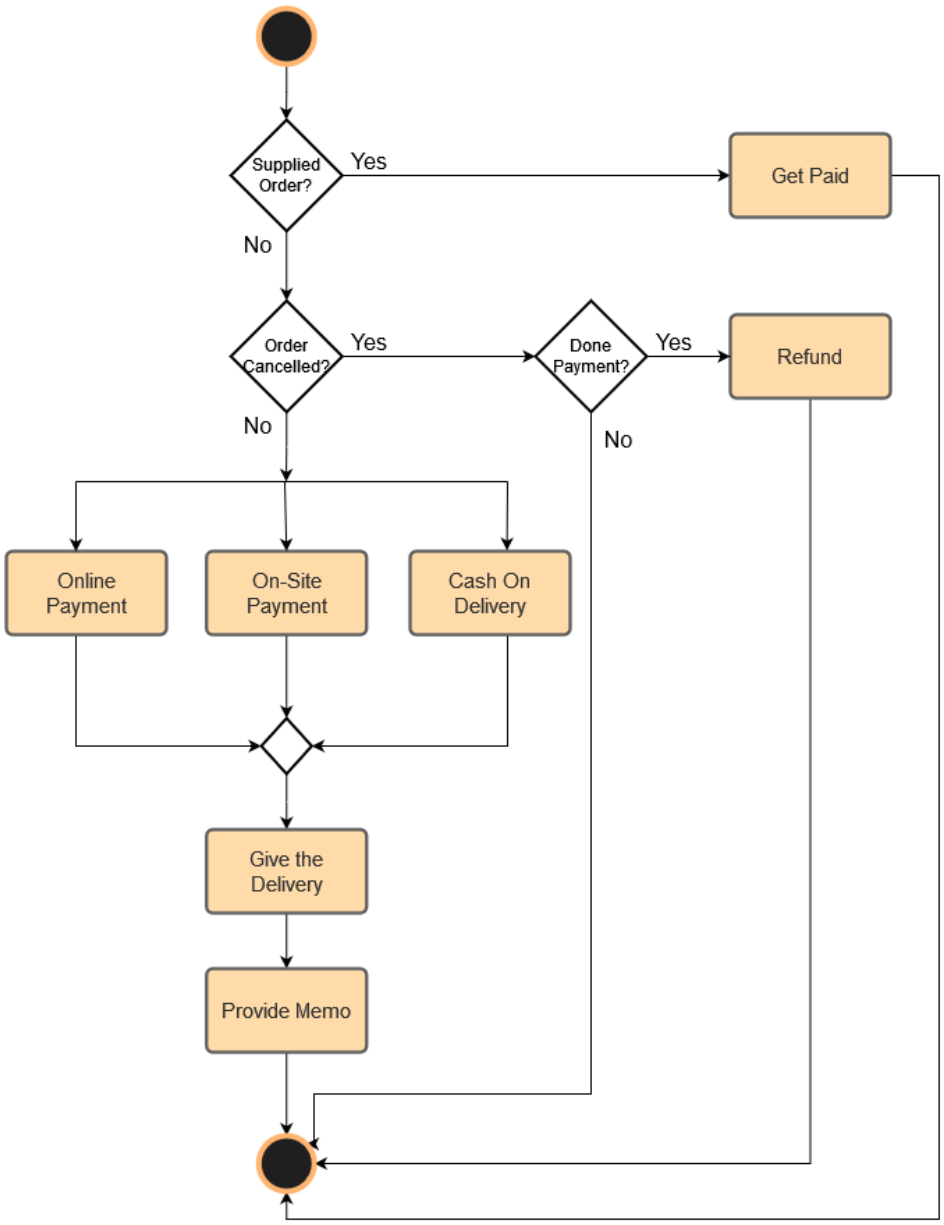


Figure 27 : Payment (Use Case Level - 1.6)

Description of activity diagram level 1.6:

1. **Get Paid:** Suppliers will receive payment after delivering any order.
2. **Refund:** Any user who ordered and paid for it, but cancelled afterwards will be refunded.
3. **Online Payment:** The user has to first pay by any payment method, then receive the order.
4. **On-Site Payment:** The customer/retailer will come to the bakery and pay for the order.
5. **Cash On Delivery:** If the user chooses COD, he/she will wait for the order to be delivered.
6. **Give the Delivery:** The delivery person will deliver the order.
7. **Provide Memo:** After the delivery and payment, a memo will be provided to him/her.

Level: 1.6.1

Name: Online Payment

Reference: Use Case Diagram Level – 1.6.1

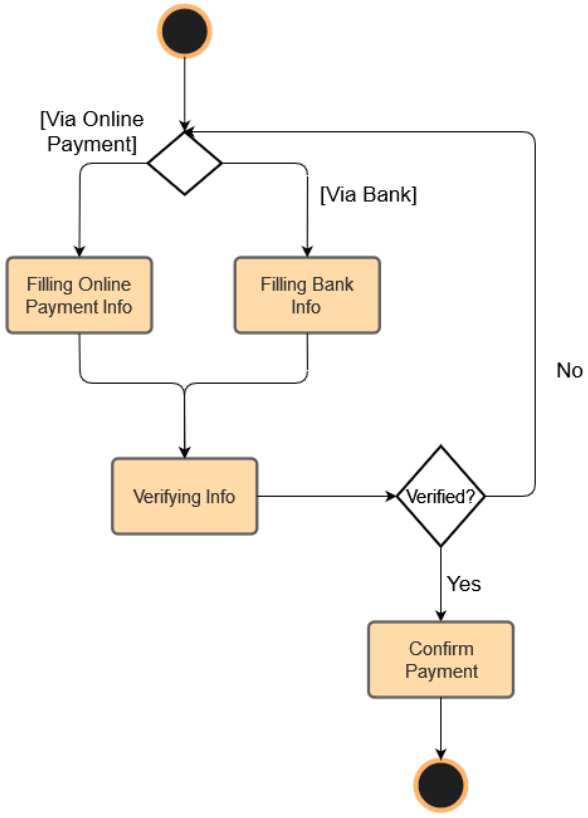


Figure 28 : Online Payment (Use Case Level - 1.6.1)

Description of activity diagram level 1.6.1:

1. **Filling Online Payment Info:** User has to give the information regarding the selected online payment method.
2. **Filling Bank Info:** If the user wants to pay through bank, he/she has to give information regarding it.
3. **Verifying Info:** The information will be verified before transaction.
4. **Confirm Payment:** After verifying, the payment will be confirmed. Otherwise, the user has to give information again.

Level: 1.6.5

Name: Refund

Reference: Use Case Diagram Level – 1.6.5

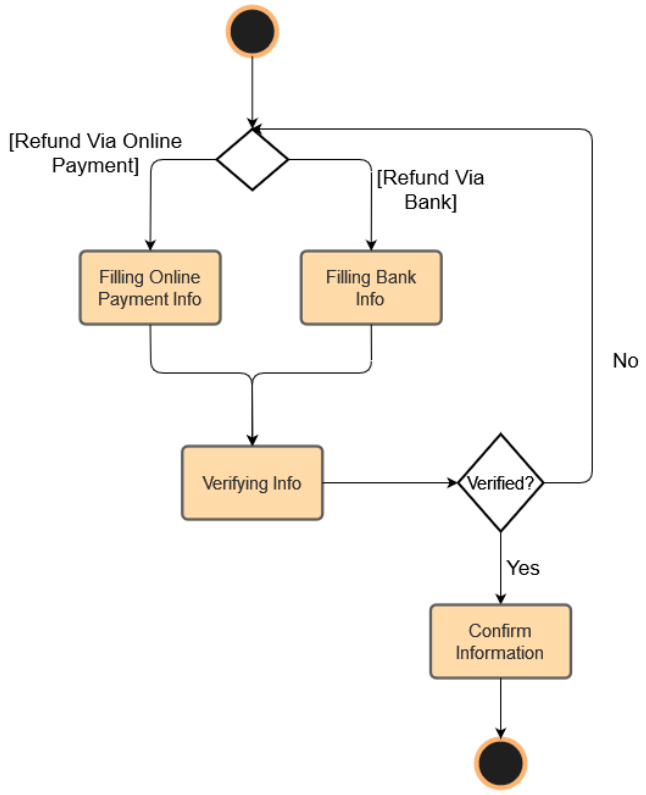


Figure 29 : Refund (Use Case Level - 1.6.5)

Description of activity diagram level 1.6.5:

1. **Filling Online Payment Info:** User has to give the information regarding the selected online payment method.
2. **Filling Bank Info:** If the user wants to get refunded through bank, he/she has to give information regarding it.
3. **Verifying Info:** The information will be verified before transaction.
4. **Confirm Information:** After verifying, the user will be refunded. Otherwise, he/she has to give information again.

Level: 1.6.6

Name: Get Paid

Reference: Use Case Diagram Level – 1.6.6

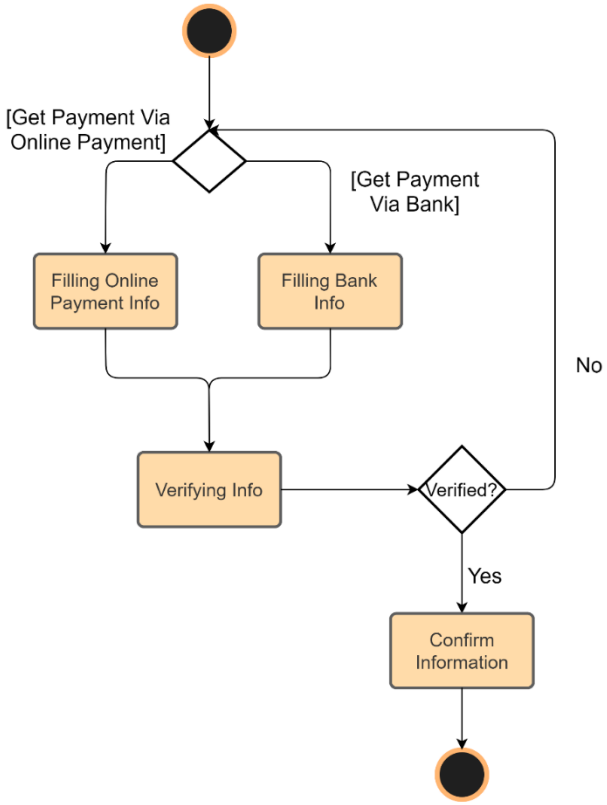


Figure 30 : Get Paid (Use Case Level - 1.6.6)

Description of activity diagram level 1.6.6:

1. **Filling Online Payment Info:** Supplier has to give the information regarding the selected online payment method.
2. **Filling Bank Info:** If the supplier wants to get paid through bank, he/she has to give information regarding it.
3. **Verifying Info:** The information will be verified before transaction.
4. **Confirm Information:** After verifying, the supplier will receive payment. Otherwise, he/she has to give information again.

Level: 1.7

Name: Notification
Reference: Use Case Diagram Level – 1.7

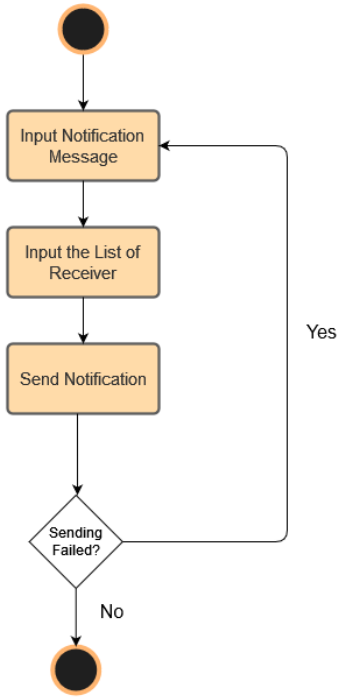


Figure 31 : Notification (Use Case Level - 1.7)

Description of activity diagram level 1.7:

1. **Input Notification Message:** System has to input the message which will be sent to notify.
2. **Input the List of Receiver:** The receivers will be listed for sending the message.
3. **Send Notification:** The notification will be sent to the listed receivers.

Swimlane Diagram

A swimlane diagram is a type of flowchart. Like a flowchart, it diagrams a process from start to finish, but it also divides these steps into categories to help distinguish which departments or employees are responsible for each set of actions. It is based on the analogy of lanes in a pool, as it places process steps within the horizontal or vertical "swimlanes" of a particular department, work group or employee, thus ensuring clarity and accountability.

Level: 1.1

Name: Account Management
Reference: Use Case & Activity Diagram Level – 1.1

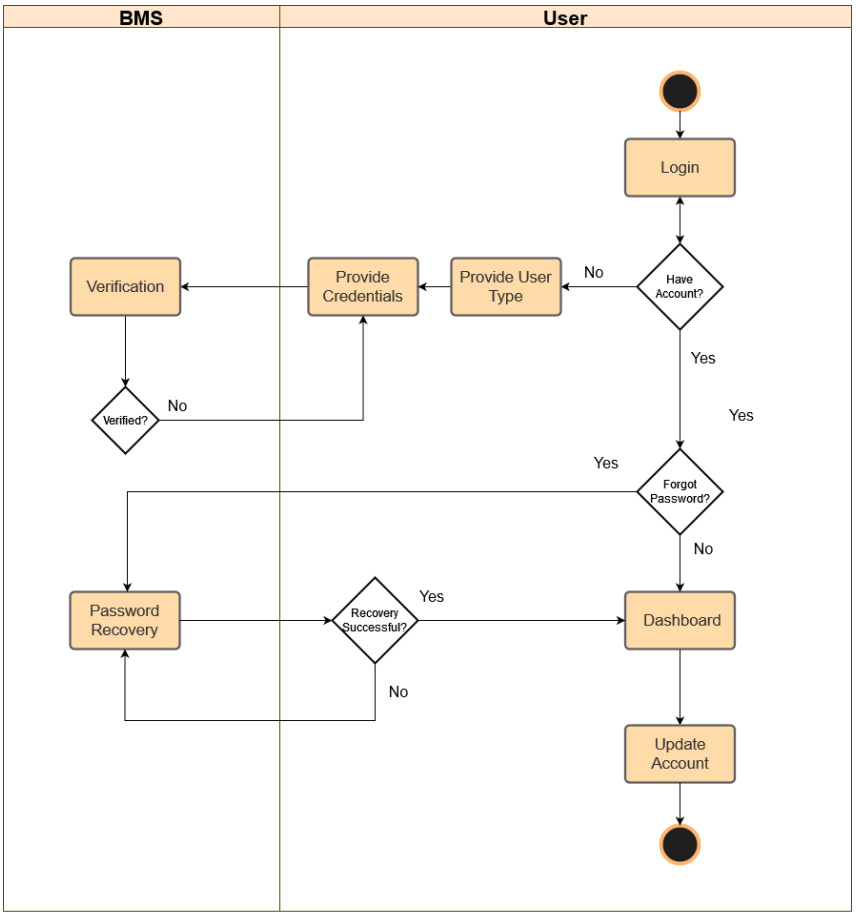


Figure 32: Account Management (Swimlane Diagram Level - 1.1)

Level: 1.1.2

Name: User Verification

Reference: Use Case & Activity Diagram Level – 1.1.2

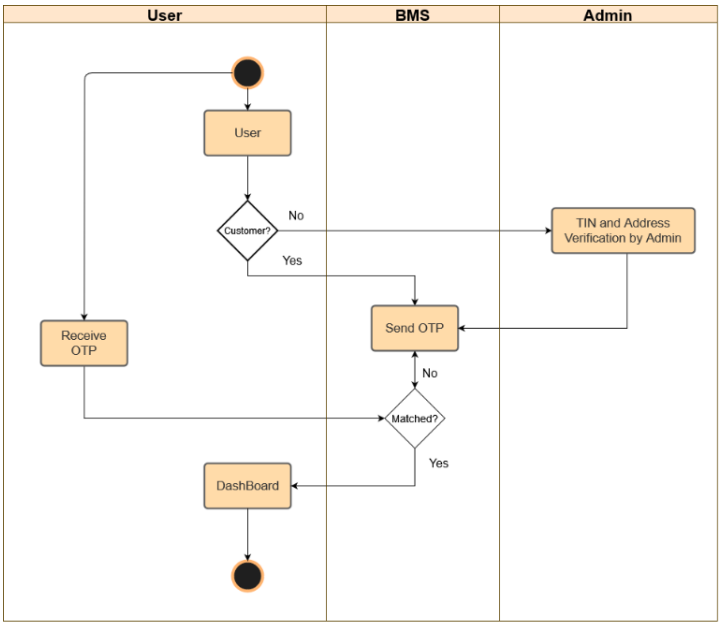


Figure 33 : User Verification (Swimlane Diagram Level - 1.1.2)

Level: 1.2

Name: Inventory Management

Reference: Use Case & Activity Diagram Level – 1.2

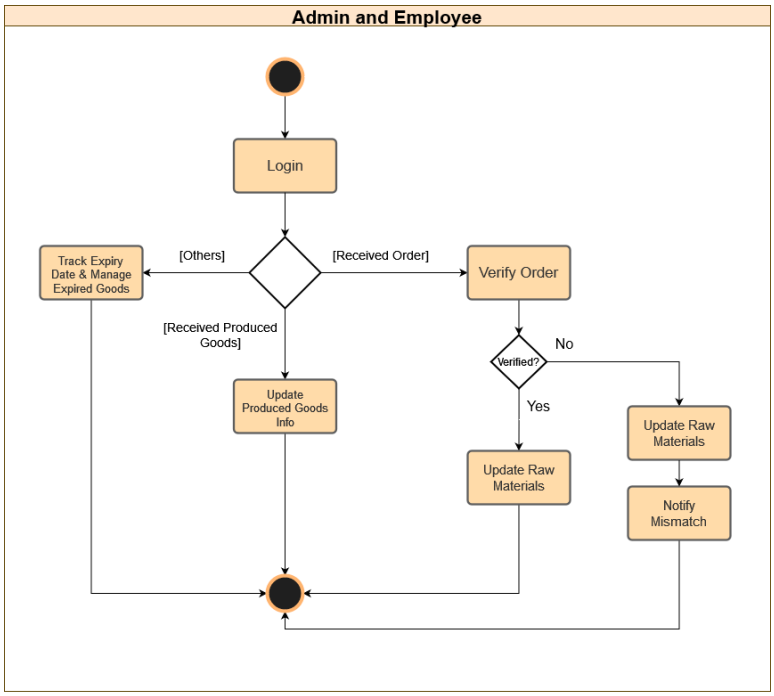


Figure 34 : Inventory Management (Swimlane Diagram Level - 1.2)

Level: 1.3

Name: Database

Reference: Use Case & Activity Diagram Level – 1.3

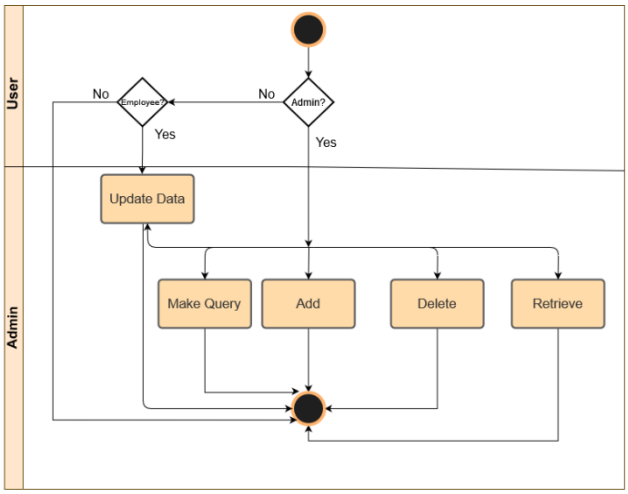


Figure 35 : Database (Swimlane Diagram Level - 1.3)

Level: 1.3.1

Name: Query

Reference: Use Case & Activity Diagram Level – 1.3.1

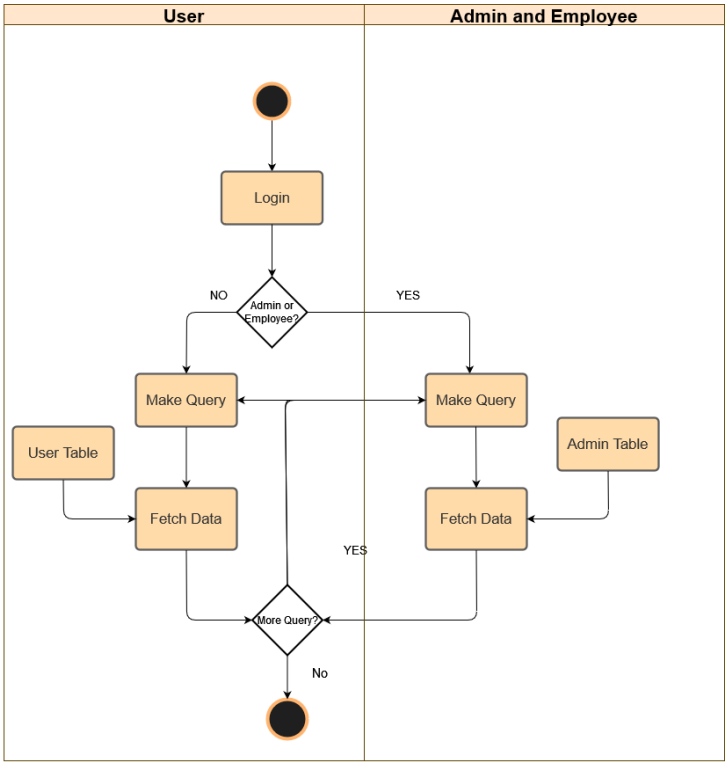


Figure 36: Query (Swimlane Diagram Level - 1.3.1)

Level: 1.3.2

Name: Create

Reference: Use Case & Activity Diagram Level – 1.3.2

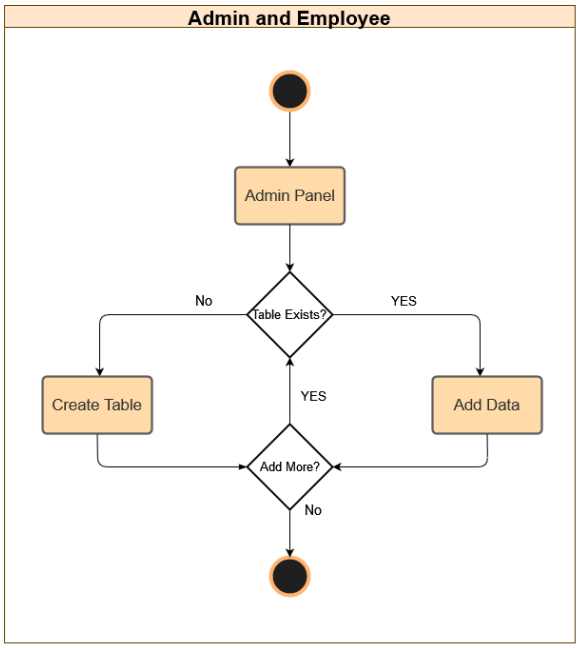


Figure 37 : Create (Swimlane Diagram Level - 1.3.2)

Level: 1.3.3

Name: Update

Reference: Use Case & Activity Diagram Level – 1.3.3

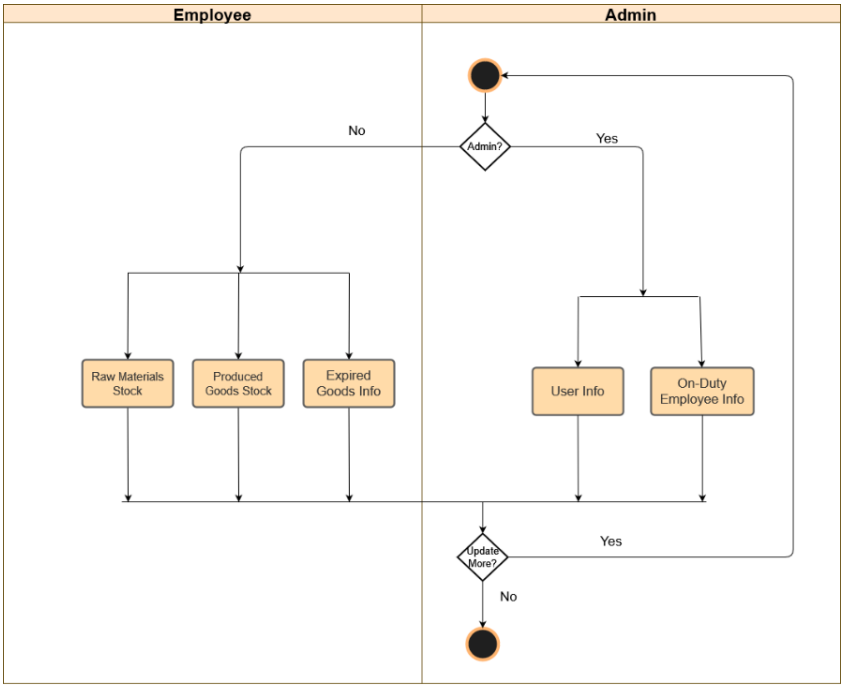


Figure 38 : Update (Swimlane Diagram Level - 1.3.3)

Level: 1.4

Name: Ordering System

Reference: Use Case & Activity Diagram Level – 1.4

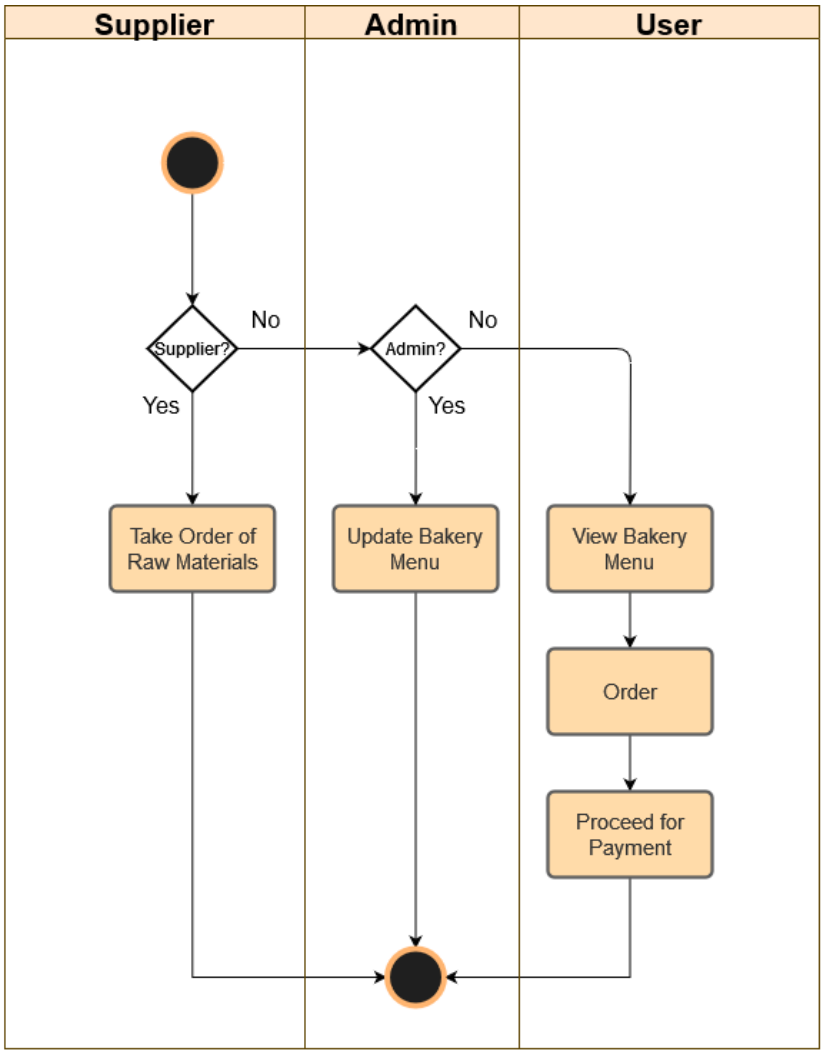


Figure 39 : Ordering System (Swimlane Diagram Level - 1.4)

Level: 1.5

Name: Logistics

Reference: Use Case & Activity Diagram Level – 1.5

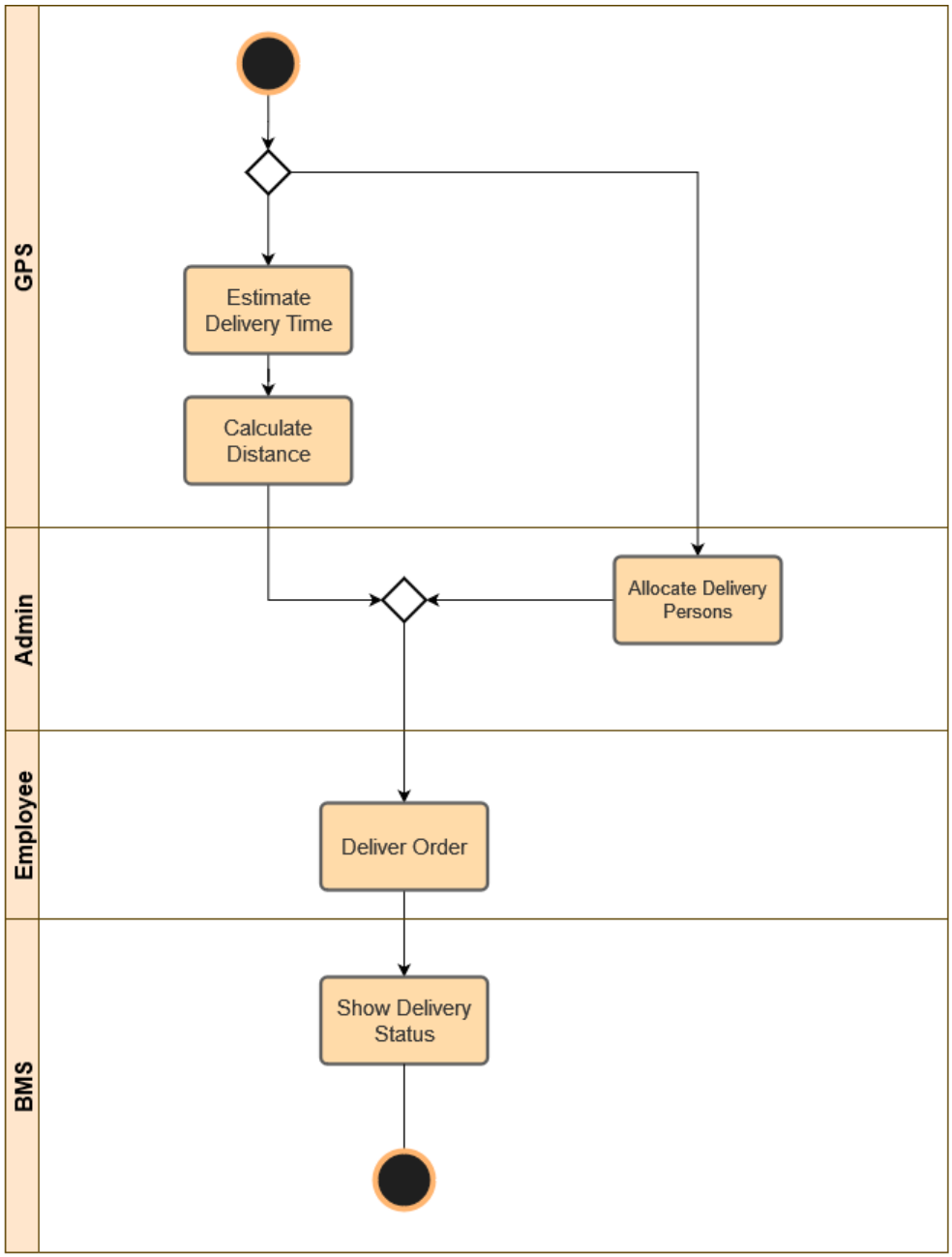


Figure 40 : Logistics (Swimlane Diagram Level - 1.5)

Level: 1.6

Name: Payment

Reference: Use Case & Activity Diagram Level – 1.6

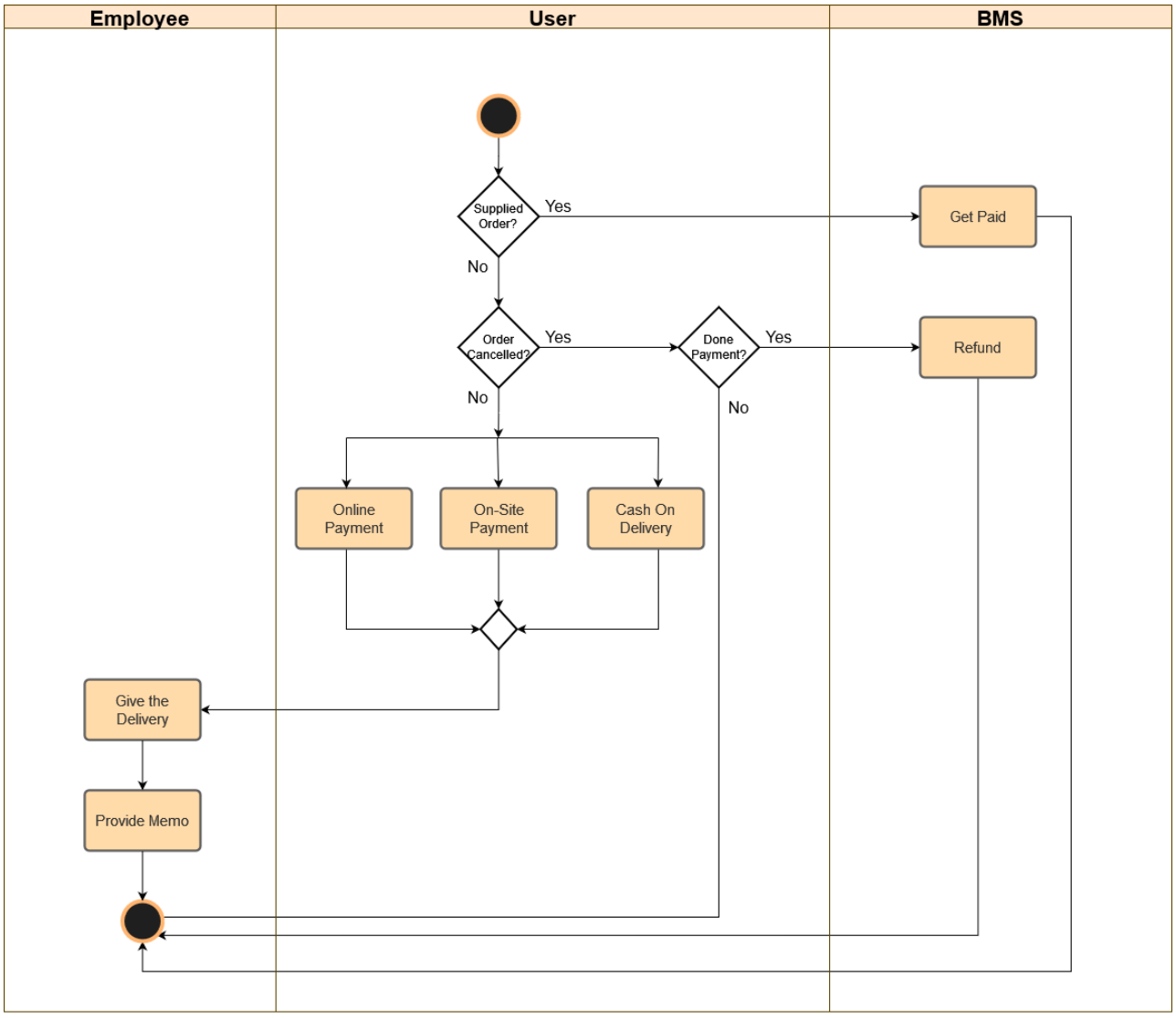


Figure 41 : Payment (Swimlane Diagram Level - 1.6)

Level: 1.6.1

Name: Online Payment

Reference: Use Case & Activity Diagram Level – 1.6.1

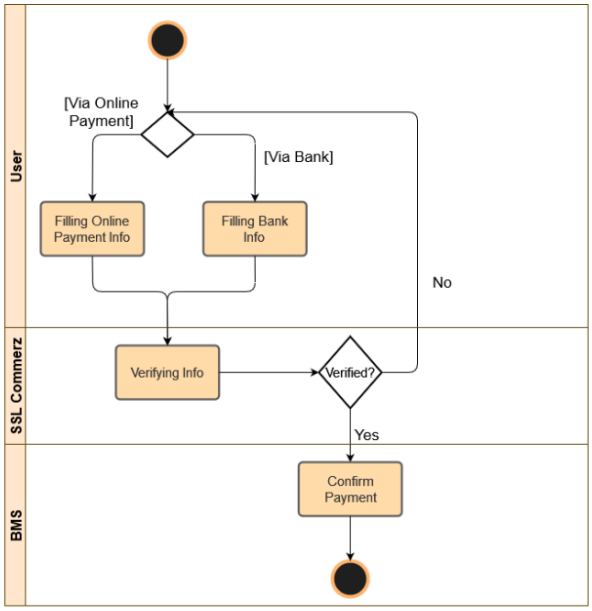


Figure 42 : Online Payment (Swimlane Diagram Level - 1.6.1)

Level: 1.6.5

Name: Refund

Reference: Use Case & Activity Diagram Level – 1.6.5

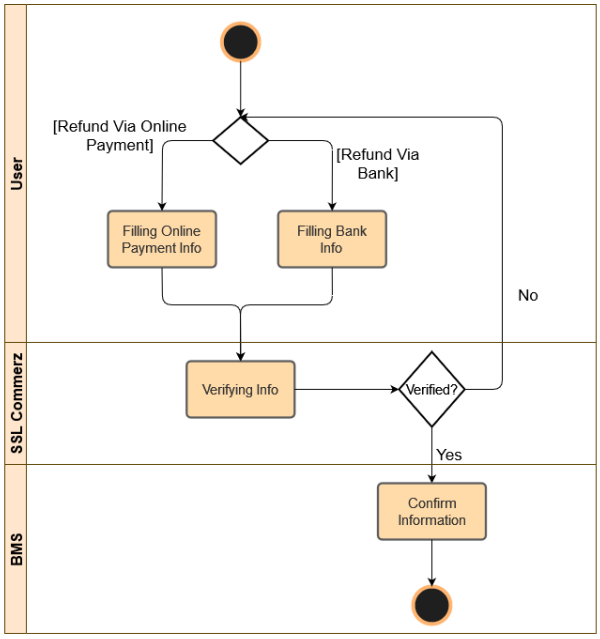


Figure 43 : Refund (Swimlane Diagram Level - 1.6.5)

Level: 1.6.6

Name: Get Paid

Reference: Use Case & Activity Diagram Level – 1.6.6

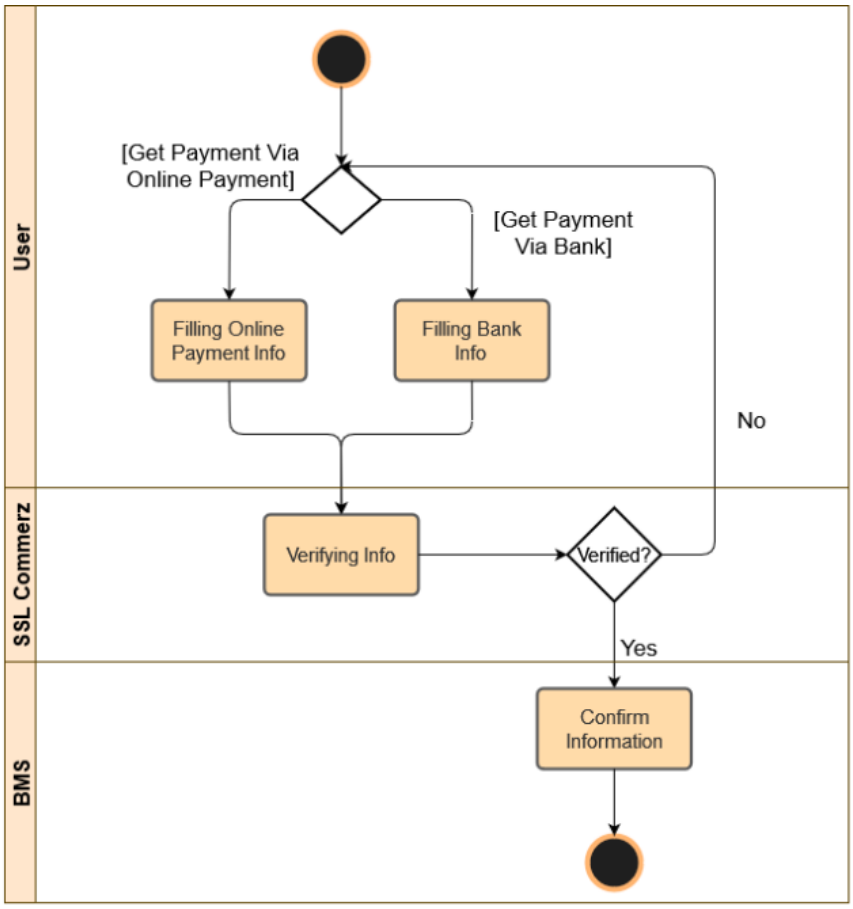


Figure 44: Get Paid (Swimlane Diagram Level - 1.6.6)

Level: 1.7

Name: Notification

Reference: Use Case & Activity Diagram Level – 1.7

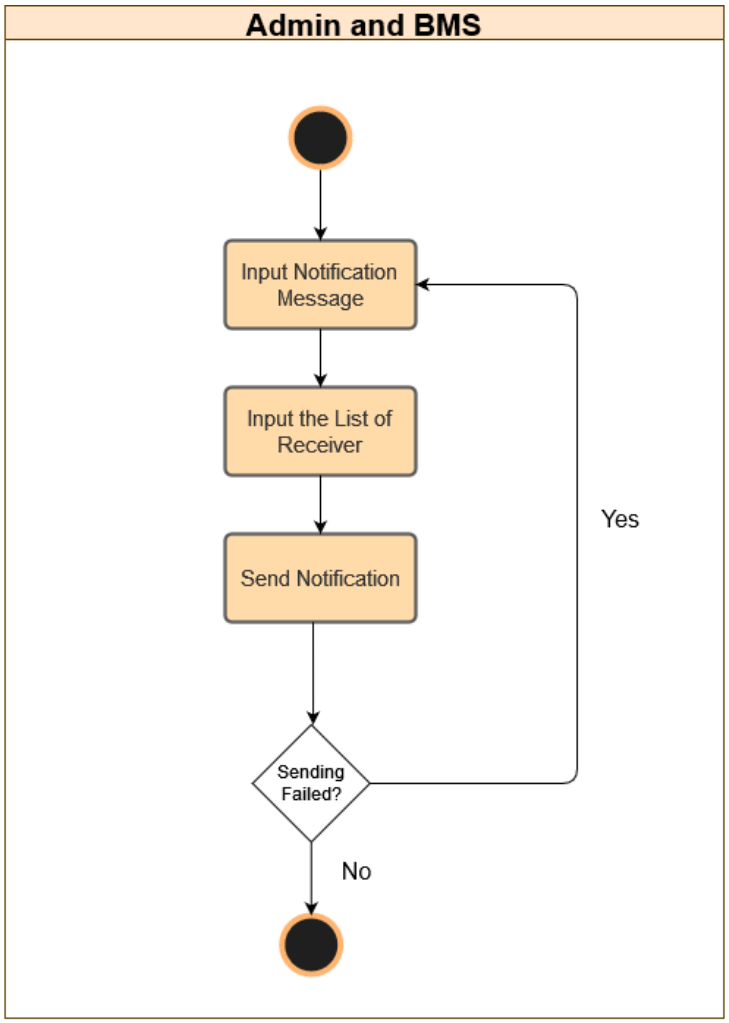


Figure 45 : Notification (Swimlane Diagram Level - 1.7)

DATA BASED MODELING

Data modeling, sometimes also called information modeling, is the process of visually representing what data the application or system will use and how it will flow. The resulting diagram or other visual representation is meant to be designed in a way that is as easy to understand as possible. The fundamental elements that a data model needs to include and describe are the data objects, more frequently called ''entities''; the attributes of those objects or entities; and the relationships between the objects or entities. While formulating real-world scenario into the database model, the ER (Entity – Relationship) Model creates entity set, relationship set, general attributes and constraints. ER Model is best used for the conceptual design of a database.

Data Objects

A data object is a region of storage that contains a value or group of values. Each value can be accessed using its identifier or a more complex expression that refers to the object. In addition, each object has a unique data type. The data type of an object determines the storage allocation for that object and the interpretation of the values during subsequent access. It is also used in any type checking operations. Both the identifier and data type of an object are established in the object declaration.

Data Object Identification

Serial	Noun	Problem(p)/ Solution(s) space	Attribute
1	BMS	s	
2	Hello Bakery	p	
3	Admin	s	85, 90, 93
4	Supplier	s	16,17, 90, 93
5	Order	s	19,33,34,37,38,39,49,61,62,79,82
6	Retailer	s	16,17,37,38,39,74,79,80,81,82,90,93
7	Account	p	
8	Information	p	
9	Full name	s	
10	Mobile number	s	
11	Email address	s	
12	Customer	s	90
13	Employee	s	85, 90
14	Password	s	
15	Employee ID	s	

16	Shop Address	s	
17	TIN number	s	
18	Verification code	s	
19	Username	s	
20	Credentials	p	
21	Database	p	
22	Brahmanbaria	p	
23	On spot order	p	
24	Grocery item	p	
25	SSLCommerz	s	
26	Bakery's transaction account	s	
27	Expired goods	p	
28	Managerial team	p	
29	Notification	s	88,89,91
30	User	s	9,10,11,14,19, 87, 90
31	Payment	p	
32	Raw materials	p	
33	Memo	s	34,37,38,39,49

34	Order id	s	
35	Delivery man	p	15,16,90
36	Cash on delivery	p	
37	Food item	s	
38	Quantity	s	
39	Total price	s	
40	Bread	p	
41	Cancellation	p	
42	Transaction	s	34,80,83,84
43	Profile	p	
44	Recovery link	s	
45	SMS	s	
46	Menu	s	37,38,49
47	Stock	s	
48	Virtual cart	s	
49	Price	s	
50	Delivery	p	
51	Bun	p	
52	Occasion	p	

53	Ceremony	p	
54	Event	p	
55	Biscuit	p	
56	Sweet	p	
57	Cake	p	
58	Records	p	
59	Money	p	
60	Service	p	
61	Delivery charge	s	
62	Estimated time	s	
63	User’s display	p	
64	Fish Feed	p	
65	GPS	s	
66	System	p	
67	Admin database	s	
68	User database	s	
69	Delivery staff number	s	
70	Availability count	s	

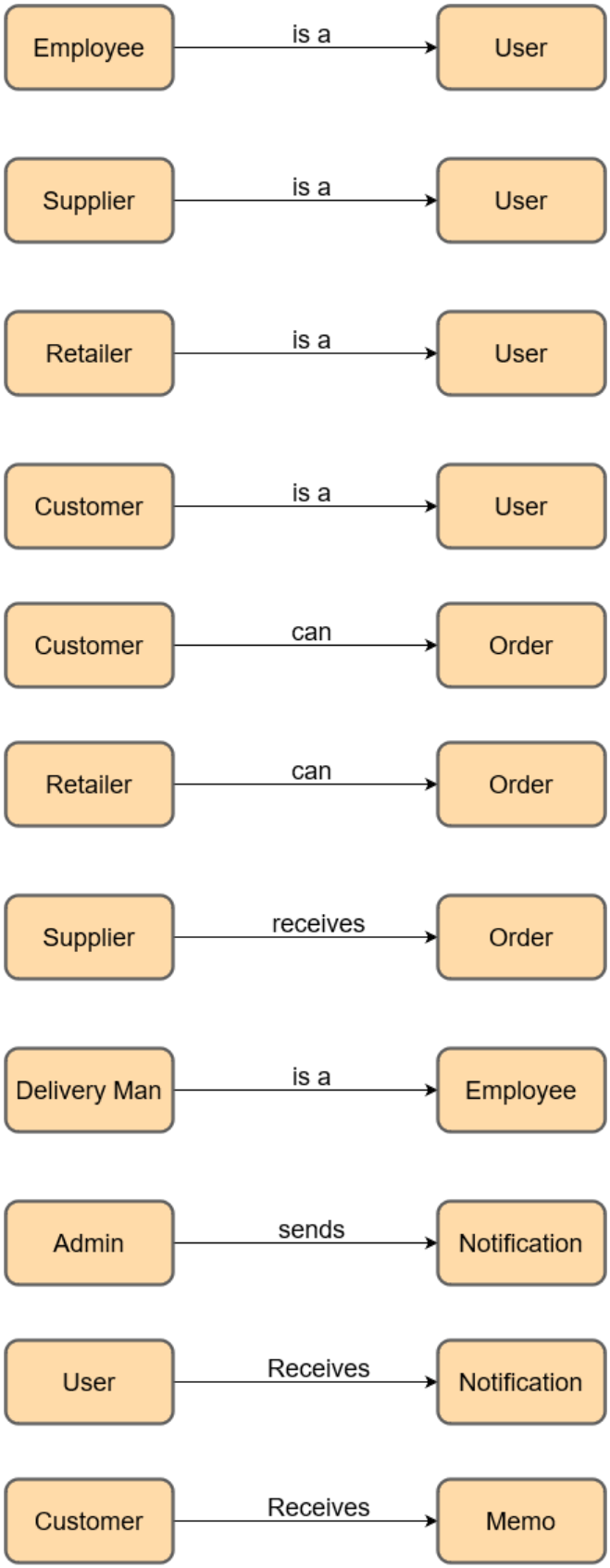
71	Sum of transactions	s	
72	Transaction history	p	
73	Replica of user info	p	
74	Quantity of Customer	s	
75	Administrative management	s	
76	Warehouse	s	16, 24,38
77	verification	s	
78	Business id	s	
79	Confirmation Time	s	
80	Reservation id	s	
81	Event time	s	
82	Request time	s	
83	Transaction id	s	
84	Transaction amount	s	
85	Designation	s	

86	Terms and Conditions	s	
87	User Type	s	
88	Notification ID	s	
89	Notification Content	s	
90	User ID	s	
91	Receiver	p	
92	Present Address	s	
93	Permanent Address	s	
94	UI	s	94

Final Data Objects

1. Admin
2. Supplier
3. Order
4. Retailer
5. Customer
6. Employee
7. Notification
8. User
9. Memo
10. Delivery Man
11. Transaction
12. Menu
13. Warehouse

Relationship Between Data Objects



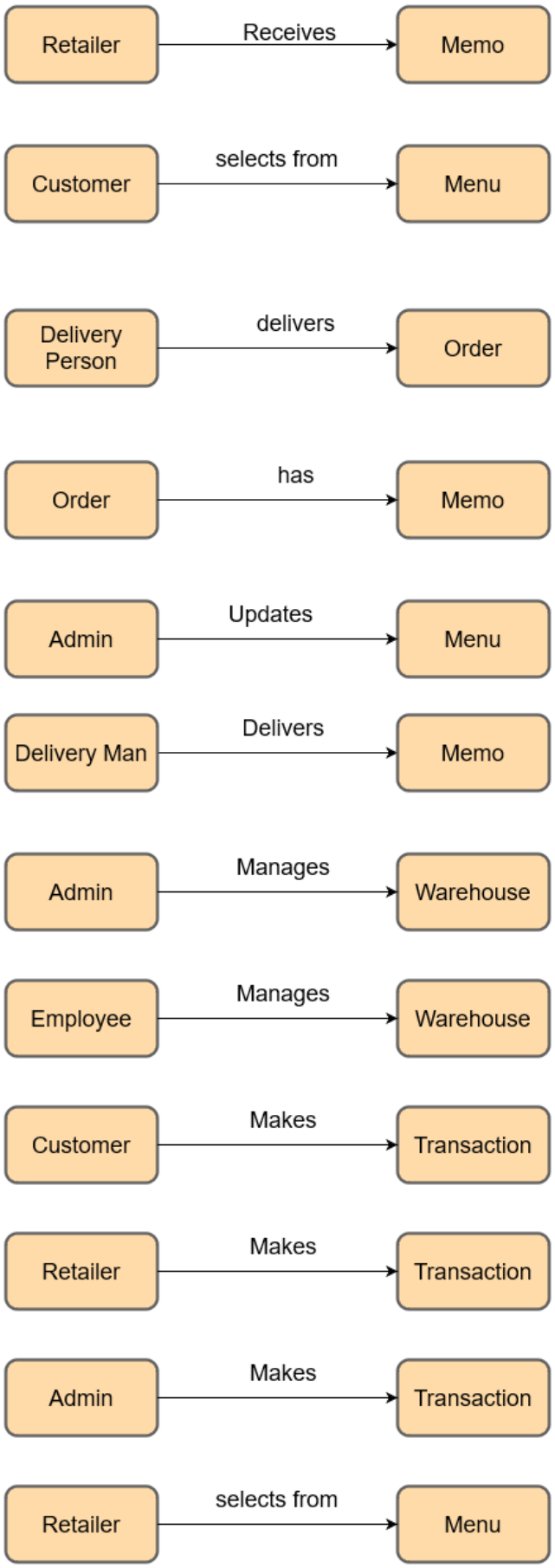


Figure 46 : Relationship between Data Objects

ER Diagram

An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set.

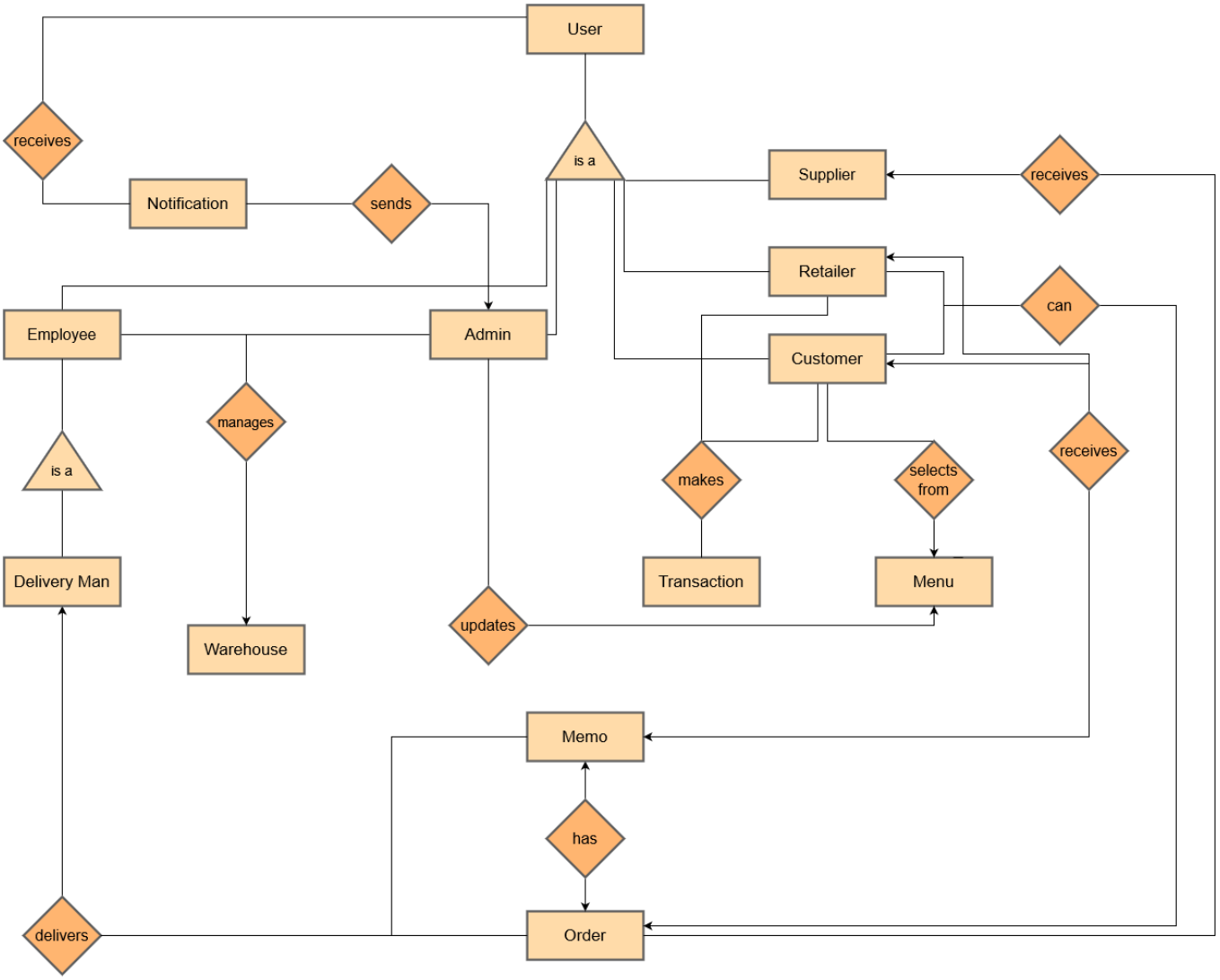


Figure 47 : ER Diagram of the Data Objects

Schema Diagram

Data Object	Attribute	Type	Size
Admin	- Designation	Varchar	40
	- User ID	Varchar	40
	- Present Address	Varchar	40
Supplier	- Shop Address	Varchar	40
	- Tin Number	Varchar	40
	- User ID	Varchar	40
	- Present Address	Varchar	40
	- Permanent Address	Varchar	40
Order	- Username	Varchar	40
	- Memo	Varchar	
	- Order ID	Varchar	40
	- Food Item	Set	
	- Quantity	Number	4
	- Total Price	Number	4
	- Price	Number	4
	- Delivery charge	Number	4
	- Estimated time	Datetime	
	- Confirmation time	Datetime	
		Datetime	

	- Request time		
Retailer	- Shop address - TIN number - Food item - Quantity - Total price - Bun - Quantity of customer - Confirmation time - Reservation ID - Event time - Request time - User ID - Present address - Permanent address	Varchar Varchar Set Number Number Number Number Datetime Varchar Datetime Datetime Varchar Varchar Varchar	40 40 4 (6,2) 4 4 40 40 40 40
Customer	- User ID	Varchar	40
Employee	- Designation - User ID	Varchar Varchar	40 40
Notification	- Notification ID - Notification content - Receiver	Varchar Varchar Varchar	40 40 40

User	- Full name - Mobile number - Email address - Password - Username	Varchar Varchar Varchar Varchar Varchar	40 40 40 40 40
Memo	- Order ID - Food item - Quantity - Total price - Price	Varchar Set Number Number Number	40 4 (6,2) (6,2)
Delivery man	- Employee ID - Shop address - User id	Varchar Varchar Varchar	40 40 40
Transaction	- Order ID - Reservation ID - Transaction ID - Transaction amount	Varchar Varchar Varchar Number	40 40 40 (6,2)
Menu	- Food item - Quantity - Price	Set Number Number	 4 (6,2)
Warehouse	- Shop address - Grocery item - Quantity	Varchar Set Number	40 4

Class-Based Modeling

Class-based modeling represents the objects that the system will manipulate, the operations (also called methods or services) that will be applied to the objects to effect the manipulation, relationships (some hierarchical) between the objects, and the collaborations that occur between the classes that are defined. The elements of a class-based model include classes and objects, attributes, operations, class-responsibility-collaborator (CRC)models, collaboration diagrams, and packages.

List of Nouns in Bakery Management System

Serial No	Noun	Serial No	Noun
1	BMS	19	Username
2	Hello Bakery	20	Credentials
3	Admin	21	Database
4	Supplier	22	Brahmanbaria
5	Order	23	On spot order
6	Retailer	24	Grocery item
7	Account	25	SSLCommerz
8	Information	26	Bakery's transaction account
9	Full name	27	Expired goods
10	Mobile number	28	Managerial team
11	Email address	29	Notification
12	Customer	30	User
13	Employee	31	Payment
14	Password	32	Raw materials
15	Employee ID	33	Memo
16	Shop Address	34	Order id
17	TIN number	35	Delivery man
18	Verification code	36	Cash on delivery

37	Food item	65	GPS
38	Quantity	66	System
39	Total price	67	Database
40	Bread	68	Delivery staff number
41	Cancellation	69	Availability count
42	Transaction	70	Sum of transactions
43	Profile	71	Transaction history
44	Recovery link	72	Replica of user info
45	SMS	73	Quantity of Customer
46	Menu	74	Administrative management
47	Stock	75	Warehouse
48	Virtual cart	76	verification
49	Price	77	Business id
50	Delivery	78	Confirmation Time
51	Bun	79	Reservation id
52	Occasion	80	Event time
53	Ceremony	81	Request time
54	Event	82	Transaction id
55	Biscuit	83	Transaction amount
56	Sweet	84	Designation
57	Cake	85	Terms and Conditions
58	Records	86	User Type
59	Money	87	Notification ID
60	Service	88	Notification Content
61	Delivery charge	89	User ID
62	Estimated time	90	Receiver
63	User’s display	91	Present Address
64	Fish Feed		

List of Verbs in Bakery Management System

Serial No	Verb	Serial No	Verb
1	Create Account	28	Order (to bakery)
2	Provide Information	29	Include (price and stock)
3	Provide User Type	30	Specify (quantity)
4	Verify Information	31	Add item (cart)
5	Send Confirmation Code	32	Set location (user)
6	Confirm Sign up	33	Pay (bill)
7	Give unique User ID	34	Confirm (Order)
8	Restructure employee distribution	35	Deliver (Products)
9	Change Designation(employee's)	36	Cancel (Order)
10	Update profile data	37	Show (estimated delivery time)
11	Reset password	38	Track (order location)
12	Send OTP (password reset)	39	Contact (Customer)
13	Notify Admins	40	Send (Notification mail and SMS)
14	Login	41	Add (Transaction Info)
15	Store (Raw material)	42	Provide Memo (Sale)
16	Supply (raw material)	43	Refund (to customer)
17	Update (database)	44	Cancel (Reservation)
18	Notify (Stock Shortage)	45	Estimate (Next day sales)
19	Store (produced goods)	46	Fix (Menu Items)
20	Prompt (memo)	47	Display (Product Availability)
21	Input (Delivery staff number)	48	Update Menu
22	Update (Number of available staff to database)	49	Store (Order details to database)
23	Add (Incoming stocks, produced goods to database)	50	Show (Suggested items)
24	Subtract (ordered items from database)	51	Show (Most ordered item)
25	Track (expiry date)	52	Run (marketing campaign)
26	Send (ordered items)	53	Prompt (monthly campaign suggestion)
27	Order (from bakery)		

General Classification

Candidate classes are categorized based on the seven general classification. The analysis classes manifest themselves in one of the following ways:

1. External entities
2. Things
3. Events
4. Roles
5. Organizational units
6. Places
7. Structures

A candidate class is selected for special classification if it fulfills three or more characteristics.

Serial No	Solution Space Nouns	General Classification
1	UI	2,3,7
2	Admin	4,5,7
3	Supplier	4,5,7
3	Order	2,3,7
4	Retailer	4,5,7
5	Full Name	2
6	Mobile Number	2
7	Email Address	2
8	Customer	4,5,7
9	Employee	4,5,7
10	Password	2
11	Employee ID	2
12	Shop Address	2
13	TIN Number	2
14	Verification Code	2
15	Username	2
16	SSLCommerz	1
17	Transaction Amount	2

18	Notification	2,3,7
19	User	4,5,7
20	Memo	2,7
21	Order ID	2
22	Food Item	2
23	Quantity	2
24	Total Price	2
25	Transaction	1,3
26	Recovery Link	2
27	SMS	1,2
28	Menu	2,7
29	Stock	2,7
30	Virtual Cart	2
31	Price	2
32	Delivery Charge	2
33	Estimated Time	2
34	GPS	1
35	Database	2,7
36	Delivery Staff Number	2
37	Availability Count	2
38	Sum of Transactions	2
39	Quantity of Customer	2
40	Administrative Management	3
41	Warehouse	7
42	Verification	3
43	Business ID	2
44	Confirmation Time	2
45	Reservation ID	2
46	Event Time	2
47	Request Time	2
48	Transaction ID	2
49	Designation	2
50	Terms and Conditions	2
51	User Type	2
52	Notification ID	2
53	Notification Content	2
54	User ID	2
55	Present Address	2
56	Permanent Address	2

Potential To Be Classes

- UI
- Admin
- Supplier
- Retailer
- Customer
- Employee
- SSLCommerz
- User
- Transaction
- SMS
- GPS
- Order

Selection Criteria

The candidate classes are then selected as classes by six Selection Criteria:

1. Retain information

2. Needed services

3. Multiple attributes

4. Common attributes

5. Common operations

6. Essential requirements

A candidate class generally becomes a class when it fulfills around three characteristics.

Serial No	Noun	Selection Criteria
1	Admin	1,2,3,4,5
2	Supplier	1,2,3,4,5
3	Retailer	1,2,3,4,5
4	Customer	1,2,3,4,5
5	Employee	1,2,3,4,5
6	SSLCommerz	6
7	User	1,2,3,4,5
8	Transaction	2,3,6
9	SMS	6
10	GPS	6
11	UI	1,2,3
12	Order	1,2,3

Selected Classes

- UI
- Admin
- Supplier
- Retailer
- Customer
- Employee
- SSLCommerz
- User
- Transaction
- SMS
- GPS
- Order

Attribute and Method Identification

Class Name	Attribute	Method
UI		+verify_info() +notify_user() +send_otp() +update_user_database() +provide_menu() +display_stock() +notify_after_transaction() +prompt_memo() +update_item_count() +display_most_ consumed_food() +display_relevant_food() +analyze_order()
User	-Full Name -Mobile Number -Email Address -Password -Username -User Type -User ID	+login() +select_user_type() +create_account() -recover_password() +verify_info() +receive_otp() +update_account() +provide_credentials()

Admin	-Designation -User ID -Permanent Address	+send_notification() +verify_credentials() +update_menu() +update_stock() -update_user_database() +update_user_info() +add_user_info() +handle_complaints() +update_stock() +check_stock_over() +update_raw_materials +add_incoming_stock() +add_produced_goods() +track_expiery_date() +substract_ordered_items() +assign_delivery_preson() +order_from_supplier() +cancel_order_from_supplier() +handle_customer_complain() +notify_supplier_if_mismatch() +pay_employee() +pay_supplier() +get_payment()
Supplier	-User ID	+receive_order()

	-Shop Address -TIN Number -Permanent Address	+get_payment() +deliver_order() +accept_or_deny_order()
Retailer	-User ID -Shop Address -Permanent Address -TIN Number -Food Item -Quantity -Total Price -Quantity of Customer -Confirmation Time -Reservation ID -Event Time -Request Time	+make_order() +make_payment() +cancel_order() +add_to_cart() +provide_feedback() +schedule_a_order() +custom_order() -update_order_history()
Customer	-User ID	+make_order() +make_payment() +cancel_order() +add_to_cart() +provide_feedback() +schedule_a_order() +custom_order() -update_order_history()
Employee	-User ID	+get_payment()

	-Designation	+update_stock() -update_user_database() +update_user_info() +add_user_info() +handle_complaints() +update_stock() +check_stock_over() +update_raw_materials +add_incoming_stock() +add_produced_goods() +track_expiery_date() +substract_ordered_items() +notify_supplier_if_mismatch()
SSLCommerz	-isTransaction_completed -isNotified	+make_transaction() +notify_admin() +add_payment_to_account()
Transaction	-Order ID -Reservation ID -Transaction ID -Transaction Amount	+input_payment_method() +input_amount() +input_recipient_info() +notify_after_payment()
SMS	-isVerified -isCodeSent -mobileNumber	+send_OTP() -create_OTP() +validate_input_mobile_number() +verify()

		+send_confirmation()
GPS	-distance -estimated_time	+get_location() +display_distance_time() +calculate_estimated_arrival() -calculate_distance()
Order	-Username -Memo -Order ID -Food Item -Quantity -Price -Total Price -Delivery Charge -Estimated Time -Confirmation Time -Request Time	+add_to_cart() +confirm_items_of_cart() +add_shipping_information() +prompt_order_details +store_order_details() +generate_memo() +refund_user() +cancel_order() -validate_order_cancellation() +on_spot_order()

Class Cards

Class: UI	
Responsibilities	Collaborators
Verifying information	Admin
Notifying user	User
Sending OTP and recovery link	User
Updating Database	
Providing menu	Customer, Retailer
Displaying stock	Supplier
Notifying after transaction	User
Prompting memo	Order
Deducting item	Order
Displaying mostly consumed food	Admin
Analyzing order of previous month	Admin
Getting estimated delivery time and distance from GPS	GPS
Updating stock and responsibilities	Employee

Class: User	
Responsibilities	Collaborators
Login any type of user	UI
Creating account by providing credentials	UI
Recovering password	UI

Verifying credentials	UI
Receiving OTP	SMS
Updating account	UI

Class: Admin	
Responsibilities	Collaborators
Verifying Supplier and Retailer Information	UI
Notifying the Recent Promotional Offers	Customer, Retailer
Ordering Raw Materials from Supplier	Supplier
Dealing with Customer Complains	Customer
Selecting the Menu and Quantity	UI
Managing the warehouse	
Assigning delivery person	Employee
Updating database	UI, Order, Transaction, User

Class: Supplier	
Responsibilities	Collaborators
Supplying the necessary raw materials as per order	Admin
Receiving payment	Transaction
Cancelling orders if necessary	Admin
Dealing with delivery mismatches	

Class: Retailer	
Responsibilities	Collaborators
Selecting items from menu	UI
Ordering	UI
Providing feedback	Admin
Making payment	Transaction
Cancelling order	UI
Requesting a refund	Admin
Scheduling a custom order for occasion	UI
Receiving notification of the recent campaigns for retailers	UI, Admin

Class: Customer	
Responsibilities	Collaborators
Selecting items from menu	UI
Ordering	UI
Providing feedback	Admin
Making payment	Transaction
Cancelling order	UI
Requesting a refund	Admin
Scheduling a custom order for occasion	UI
Receiving notification	UI, Admin

Class: Employee	
Responsibilities	Collaborators
Getting payment	Transaction
Updating stock information	UI
Updating Database	User, Order, Supplier, Transaction
Notifying supplier if mismatch occurs	Supplier
Getting assigned for tasks	Admin

Class: SSLCommerz	
Responsibilities	Collaborators
Making transaction	Transaction
Notifying admin about payment completion	UI
Adding paid amount to bakery's transaction account	Transaction

Class: Transaction	
Responsibilities	Collaborators
Getting the amount and recipient information	Order
Performing Transaction	SSLCommerz
Notifying the success or failure of the transaction	UI

Class: SMS	
Responsibilities	Collaborators
Getting the phone number	UI
Sending OTP	User
Getting the code as input	
Verifying the codes	UI

Class: GPS	
Responsibilities	Collaborators
Getting the location and distancing	Order
Display distance and time	UI
Calculating estimated arrival	
Sends estimated time and distance	UI

Class: Order	
Responsibilities	Collaborators
Adding items to cart	UI
Adding shipping information	UI
Prompting and storing order details	UI
Generating memo	
Refunding the user	Transaction
Validating and cancelling order	GPS
Making On Spot order	

CRC Diagram

ID: 1

Name: UI

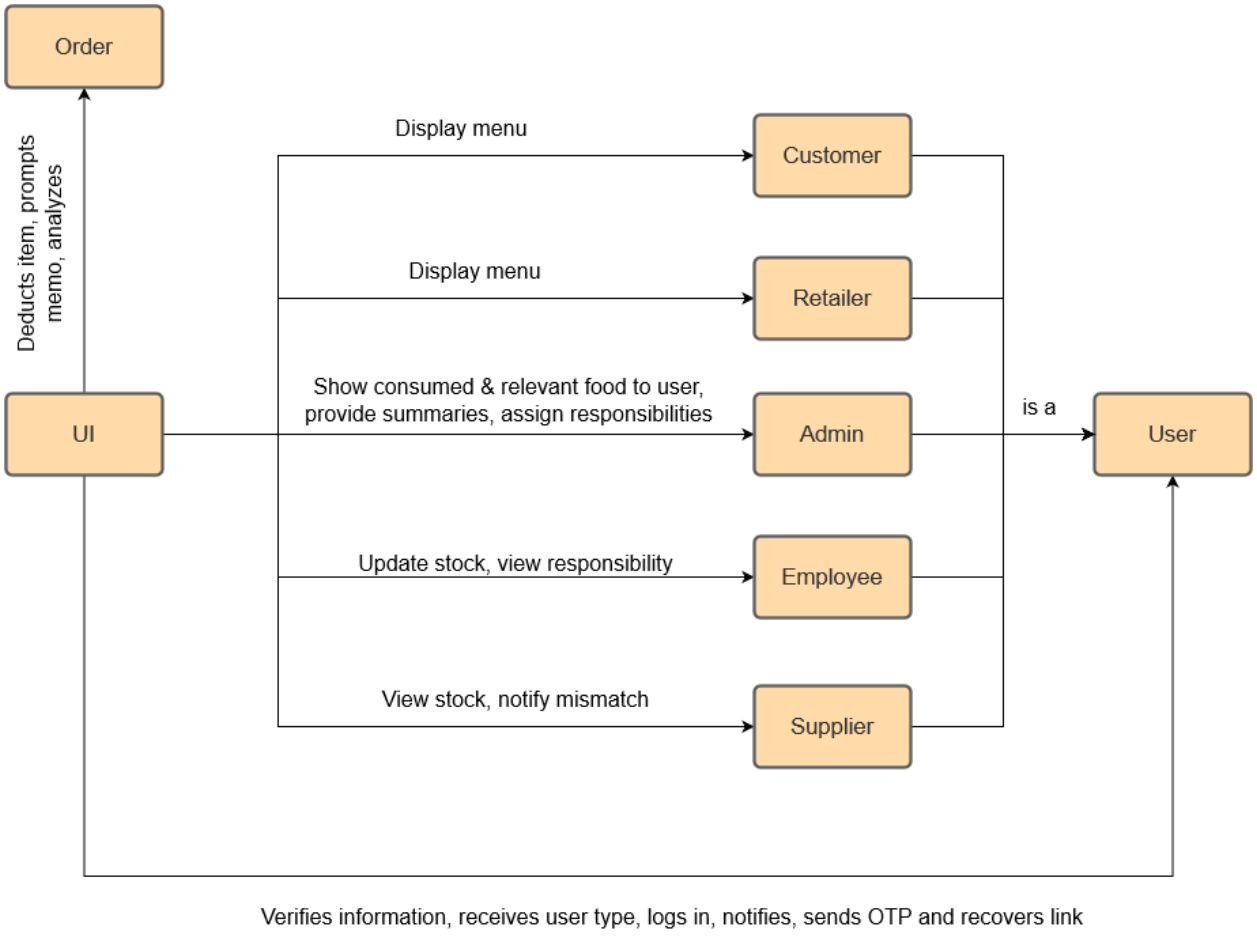


Figure 48 : CRC Diagram for UI Class

ID: 2

Name: User

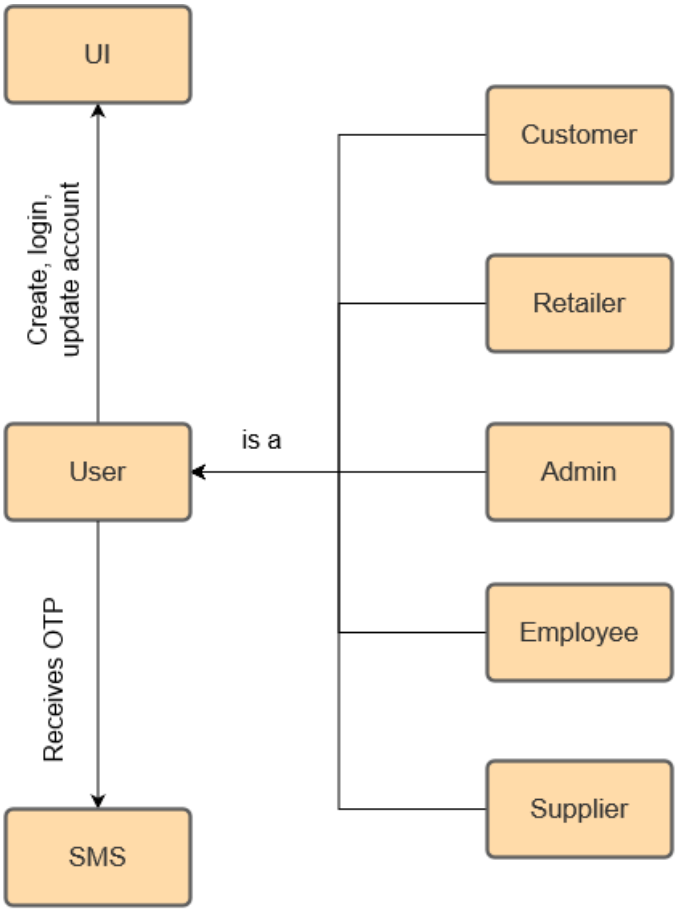


Figure 49 : CRC Diagram for User Class

ID: 3

Name: Admin

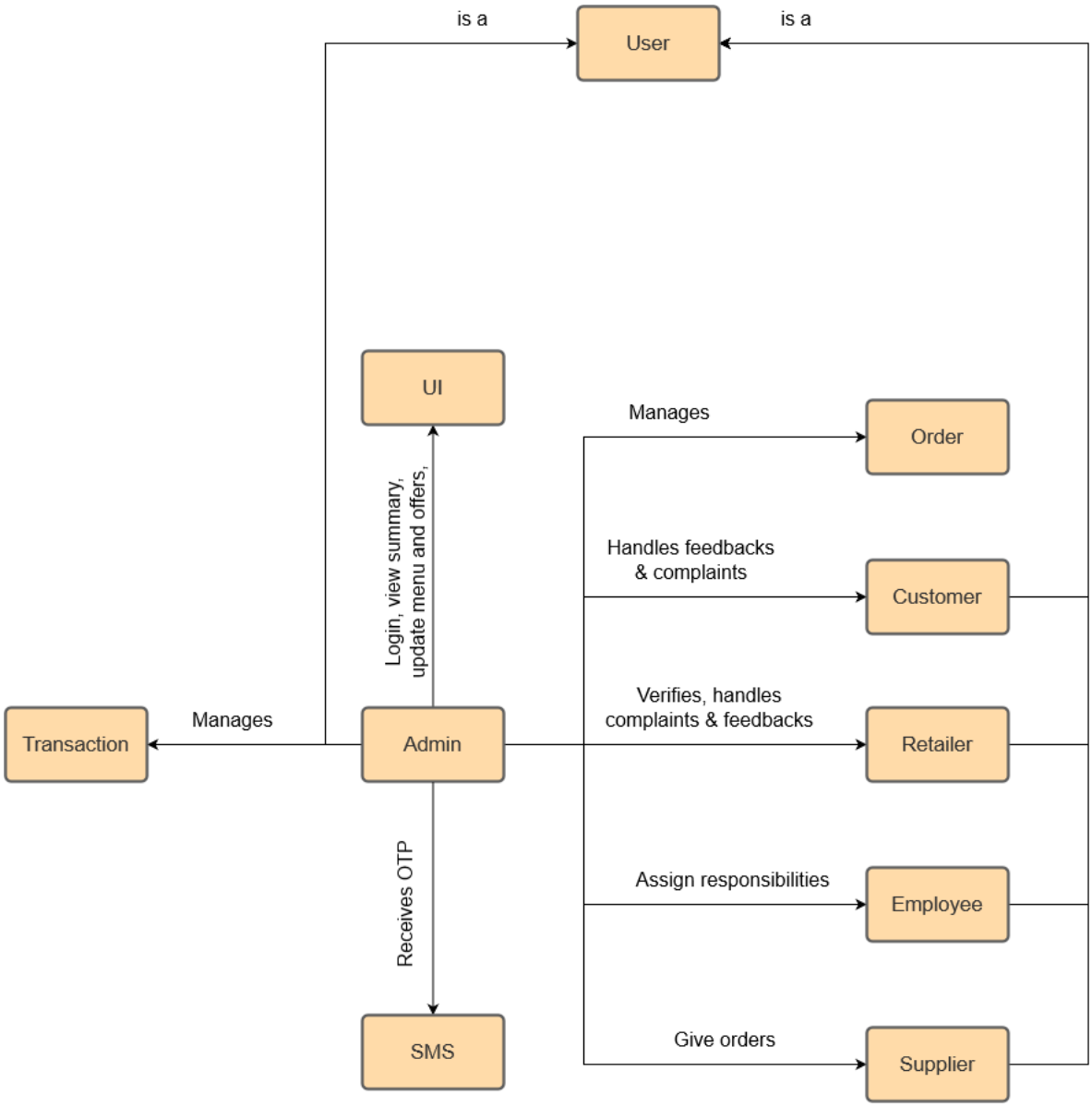


Figure 50 : CRC Diagram for Admin Class

ID: 4

Name: Supplier

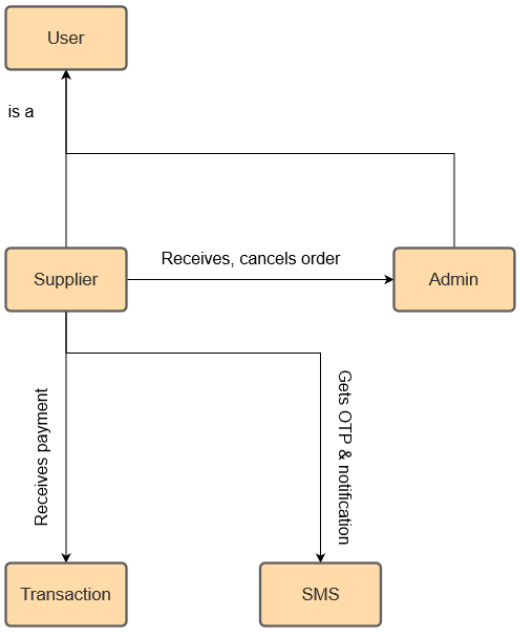


Figure 51 : CRC Diagram for Supplier Class

ID: 5

Name: Retailer

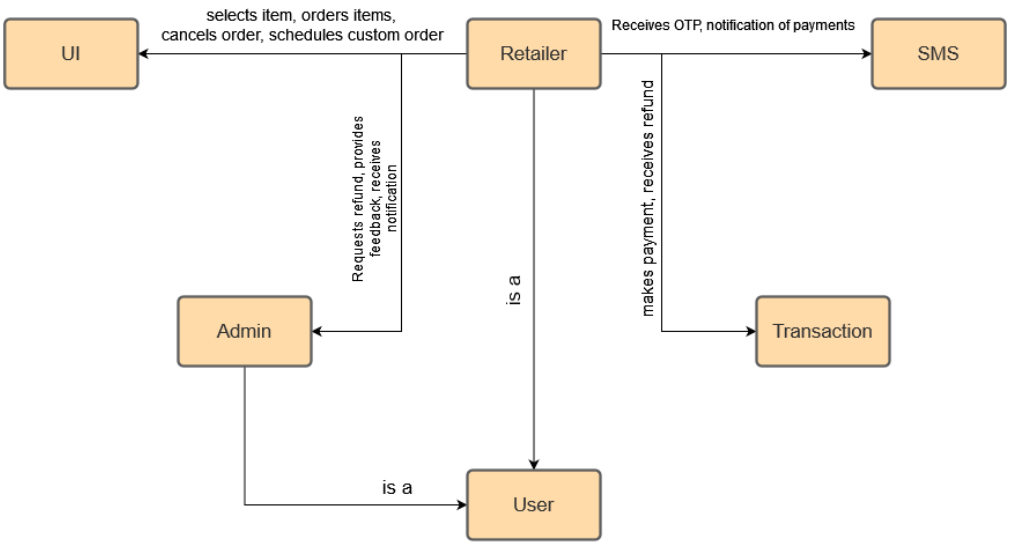


Figure 52 : CRC Diagram for Retailer Class

ID: 6

Name: Customer

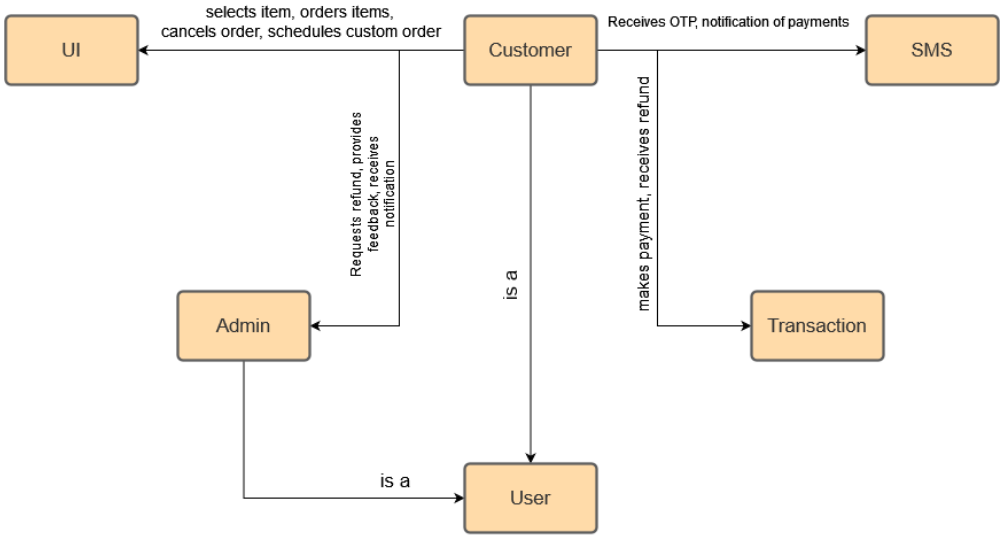


Figure 53 : CRC Diagram for Customer Class

ID: 7

Name: Employee

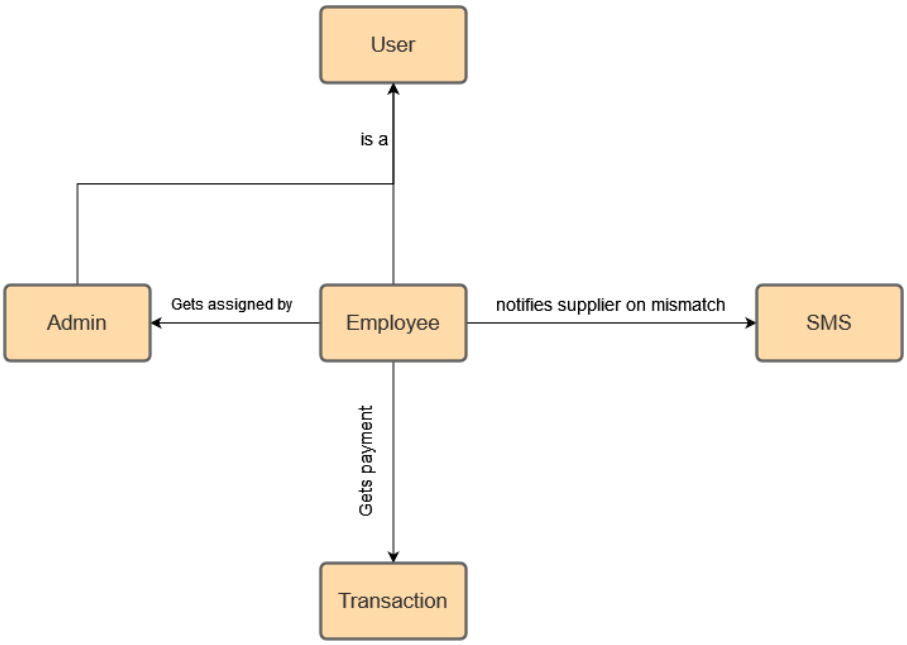


Figure 54 : CRC Diagram for Employee Class

ID: 8

Name: SSLCommerz

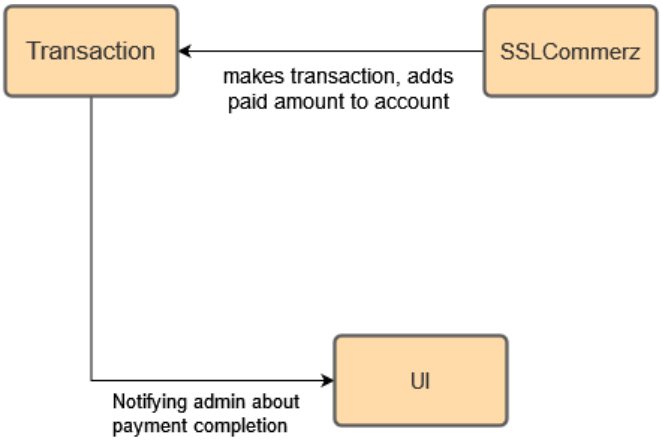


Figure 55 : CRC Diagram for SSLCommerz Class

ID: 9

Name: Transaction

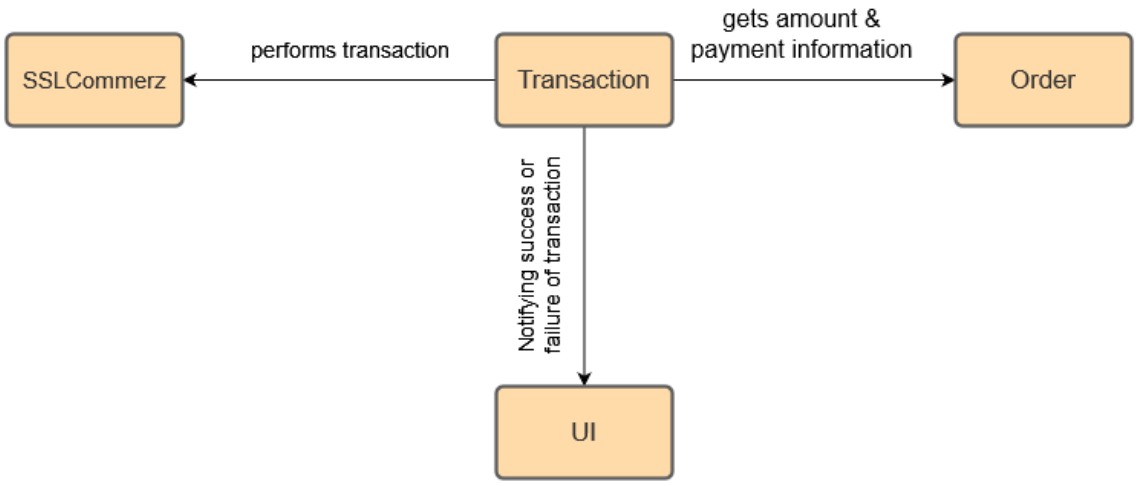


Figure 56 : CRC Diagram for Transaction Class

ID: 10

Name: SMS

Figure 57 : CRC Diagram for SMS Class

ID: 11

Name: GPS

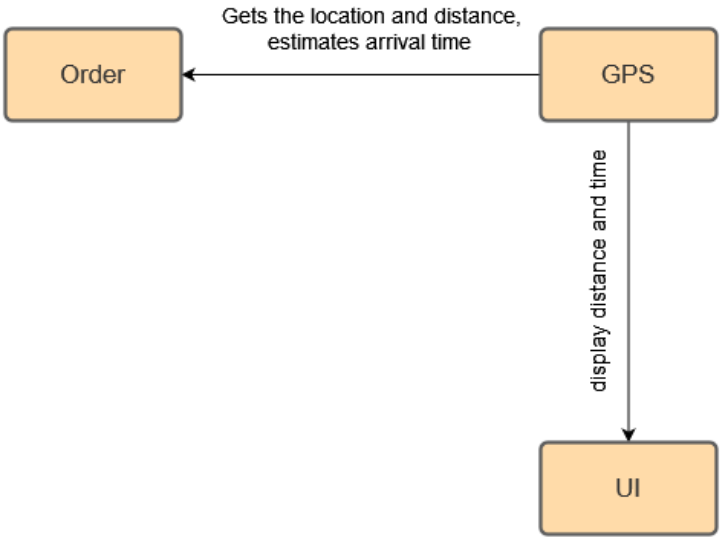


Figure 58 : CRC Diagram for GPS Class

ID: 12

Name: Order

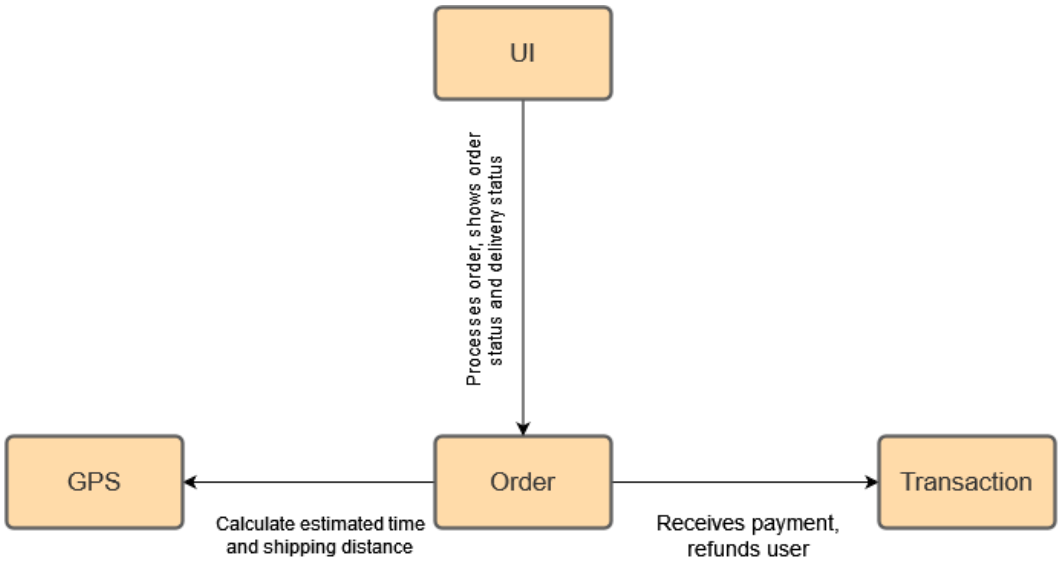


Figure 59 : CRC Diagram for Order Class

Behavioral Modeling

The behavioral model indicates how software will respond to external events or stimuli. In the context of behavioral modeling, two different characterizations of states must be considered: (1) the state of each class as the system performs its function and (2) the state of the system as observed from the outside as the system performs its function.

State Transition Diagram

One component of a behavioral model is a UML state diagram that represents active states for each class and the events (triggers) that cause changes between these active states.

Event Table

Serial No	Event	Event Name	Initiator	Collaborator	Associated Methods
1.	App Launch	Launch_APP	UI	UI, User	+launch()
2.	Choose to Login	Login_To_System	User	User, UI	+login() +get_user_type() +get_credentials() +verify_credentials() +launch_dashboard() +notify_credential_mismatch()
3.	Choose to Sign Up	Sign_up_to_System	User	User, UI	+signup() +get_user_type() +get_credentials() +verify_credentials()

					+update_user_database() +launch_dashboard()
4.	Sign up or log in	Launch_Dashboard	User	User, UI	+launch_dashboard() +display_stock() +display_offer() +display_available_items()
5.	Forget Password	Password_Recovery	User	User, UI, SMS	+send_code() +receive_code() +validate_input_mobile_number() +verify() +send_confirmation()
6.	Order Food	Order_Food	Customer, Retailer	Customer, Retailer, UI, Order	+order() +display_menu() +choose_item() +select_quantity() +select_delivery_address()
7.	Verify Information	Verify_Info	Admin		+verify_credentials()
8.	Notify Users	Notify_User	UI, SMS	User	+notify_user()
9.	Send OTP	Send_OTP	UI, SMS		+send_otp()
10.	Update User Table	Update_User_Table	Admin, Employee	UI	+update_user_database()
11.	Update Admin Table	Update_User_Table	Admin, Employee	Admin, Customer, Supplier, Retailer	+update_user_info() +add_user_info() +handle_complaints()

12.	Update Inventory Table	Update_Inventory_Table	Admin, Employee	Supplier, Order	+update_stock() +check_stock_over() +update_raw_materials +add_incoming_stock() +add_produced_goods() +track_expiery_date() +substract_ordered_items()
13.	Display Stock	Display_Stock	UI	Admin	+display_stock()
14.	Update Menu	Update_Menu	Admin	User, UI	+update_menu()
15.	Update Stock	Update_Stock	Employee	Admin	+update_stock()
16.	Notify after Transaction	Notify_after_Transaction	Transaction	SMS	+notify_after_payment()
17.	Prompt Memo	Prompt_Memo	UI	Order	+prompt_memo()
18.	Generate Memo	Generate_Memo	Order		+generate_memo()
19.	Update Item Count	Update_Item_Count	UI	Employee	+update_item_count()
20.	Display Most Consumed Food	Display_Most_Consumed_Food	UI	Admin	+display_most_consumed_food()
21.	Display Relevant Food	Display_Relevant_Food	UI	Admin	+display_relevant_food()
22.	Analyze Order	Analyze_Order	UI	Admin	+analyze_order()
23.	Select User Type	Select_User_Type	User	UI	+select_user_type()
24.	Receive OTP	Receive_OTP	SMS	UI	+receive_code()
25.	Update Account	Update_Account	User	UI	+update_account()
27.	Assign Staffs	Assign_Staffs	Admin	Admin, Employee,	+assign_delivery_person() +update_available_staff()

				Delivery Person	
28.	Order Raw Materials	Order_Materials	Admin	Supplier	+order_from_supplier()
29.	Cancel Order from Supplier	Cancel_Order_from_Supplier	Admin	Supplier	+cancel_order_from_supplier()
30.	Handle Complaints	Handle_Complaints	Admin	Customer, Retailer	+handle_customer_complain()
31.	Notify Mismatch	Notify_Mismatch	Admin, Employee	SMS	+notify_supplier_if_mismatch()
32.	Pay Employees	Pay_Employees	Admin	Employee	+pay_employee()
33.	Pay Suppliers	Pay_Suppliers	Admin	Supplier	+pay_supplier()
34.	Accept or deny Order of supply	Accept_or_deny_Order_of_supply	Supplier	Admin	+accept_or_deny_order()
35.	Supply Raw Materials	Supply_Raw_Materials	Supplier	Admin	+deliver_order()
36.	Make Payment of Order	Make_Payment_of_Order	Customer, Retailer	Transaction	+make_payment()
37.	Cancel Order	Cancel_Order	Customer, Retailer	UI	+cancel_order()
38.	Add Items to Cart	Add_Items_to_Cart	Customer, Retailer	UI	+add_to_cart()
39.	Provide Feedbacks	Provide_Feedbacks	Customer, Retailer	Admin	+provide_feedback()
40.	Schedule Order	Schedule_Order	Customer, Retailer	UI	+schedule_a_order()
41.	Custom Order	Custom_Order	Customer, Retailer	UI	+custom_order()

42.	Update Order History	Updat_Order_History	UI		-update_order_history()
43.	Complete Transactions	Complete_Transactions	SSLCommerz	Transaction	+make_transaction()
44.	Add Payment to Account	Add_Payment_to_Account	SSLCommerz	Transaction	+add_payment_to_account()
45.	Take Payment Method	Take_Payment_Method	Transaction	Order	+input_payment_method()
46.	Take Amount	Take_Amount	Transaction	Order	+input_amount()
47.	Take Recipient Information	Take_Recipient_Information	Transaction	Order	+input_recipient_info()
48.	Verify Given Mobile Number	Verify_Given_Mobile_Number	SMS		+validate_input_mobile_number() +verify()
49.	Send Confirmation	Send_Confirmation	SMS	User	+send_confirmation()
50.	Get Delivery Location	Get_Location	GPS	Order	+get_location()
51.	Calculate Estimated Time	Calculate_Time	GPS	Order	+calculate_estimated_arrival
52.	Calculate Distance	Calculate_Distance	GPS		-calculate_distance()
53.	Show Distance and Time	Show_Distance_Time	GPS	UI	+display_distance_time
54.	Confirm Items of Cart	Confirm_Items	Order	UI	+confirm_items_of_cart()
55.	Add Shipping information	Add_Shipping_Info	Order	UI	+add_shipping_information()

56.	Prompt Order Details	Prompt_Order	Order	UI	+prompt_order_details
57.	Store Order Details	Store_Order	Order		+store_order_details()
58.	Refund	Refund	Order	User	+refund_user()
59.	Validate Order Cancellation	Validate_Order_Cancellation	Admin		-validate_order_cancellation()
60.	On Spot Order	OnSpot_Order	Customer, Retailer	UI	+on_spot_order()
61.	View Pending Info to be Verified	View_Pending_Info	Admin	Admin, User	-verify_refund() -verify_credentials()
62.	Provide Menu	Provide_Menu	UI	Customer, Retailer	+provide_menu()
63.	Request Payment	Request_Payment	Supplier	Admin	+get_payment()
64.	Check if location within range	Check_Within_Range	GPS	Order	+check_if_within_range()
65.	Proceed for Ordering	Proceed_For_Ordering	UI	Order	+go_to_order()

State Transition Diagram

ID: 1

Name: UI

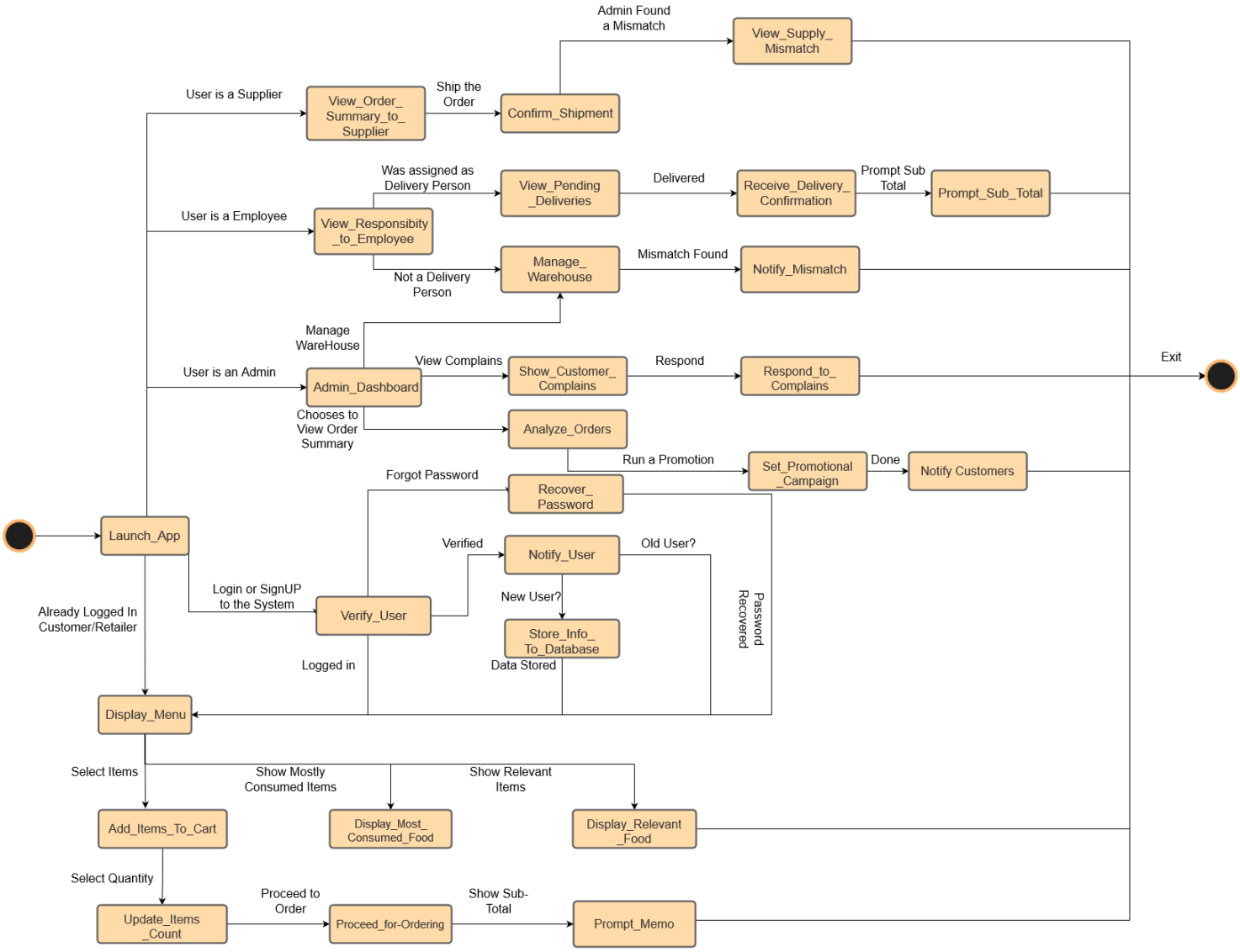


Figure 60 : State Transition Diagram for UI Class

ID: 2

Name: User

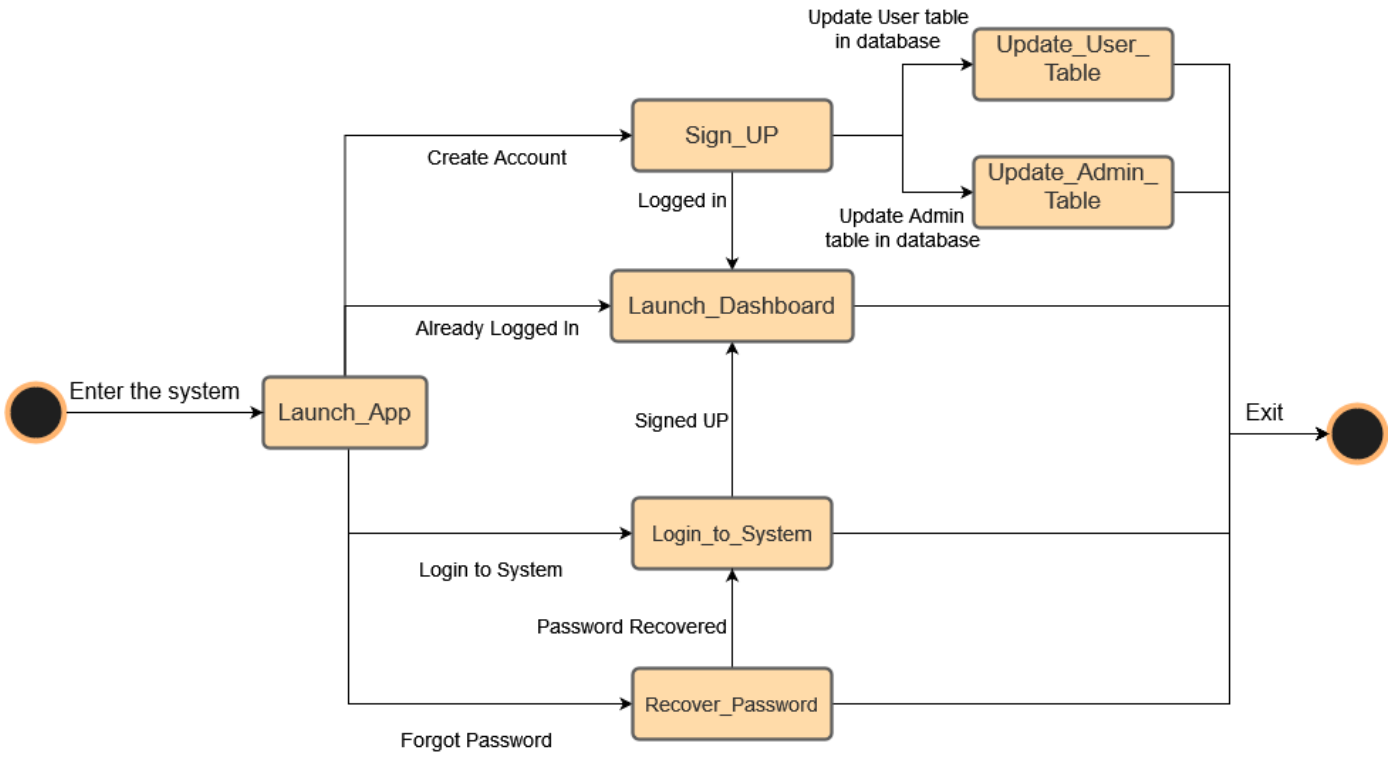


Figure 61 : State Transition Diagram for User Class

ID: 3

Name: Admin

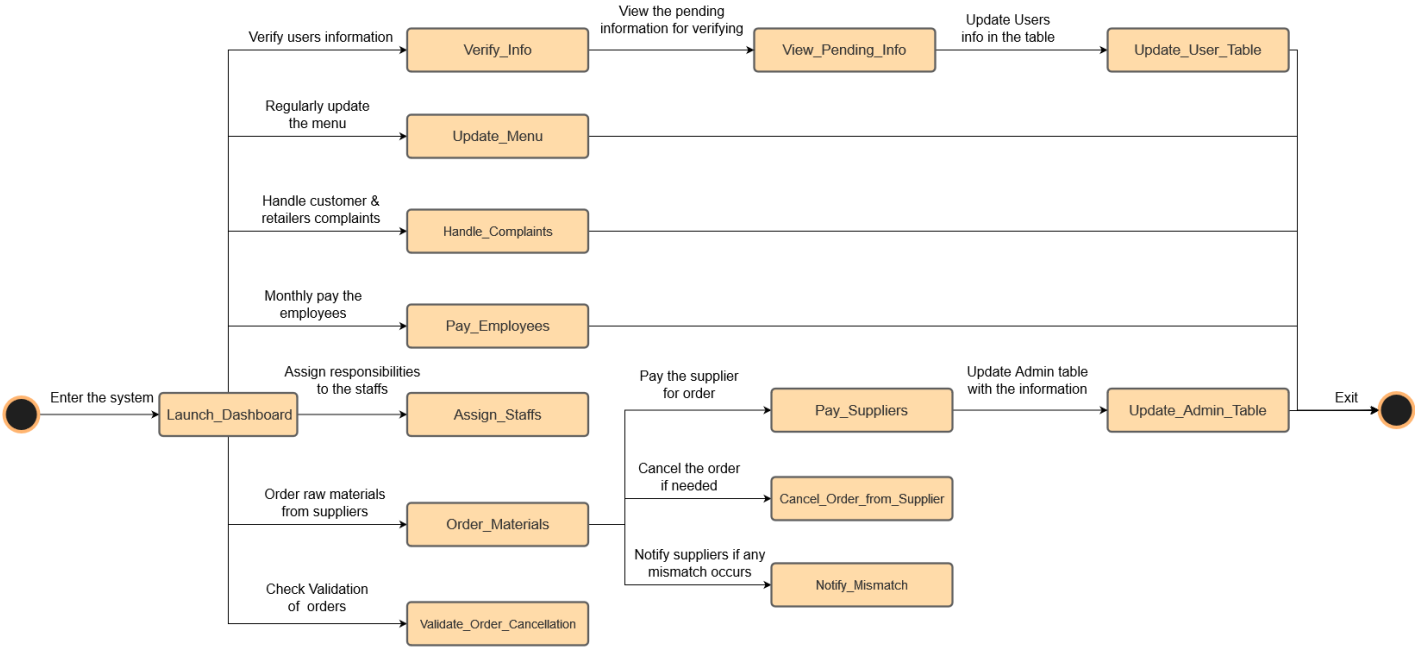


Figure 62 : State Transition Diagram for Admin Class

ID: 4

Name: Supplier

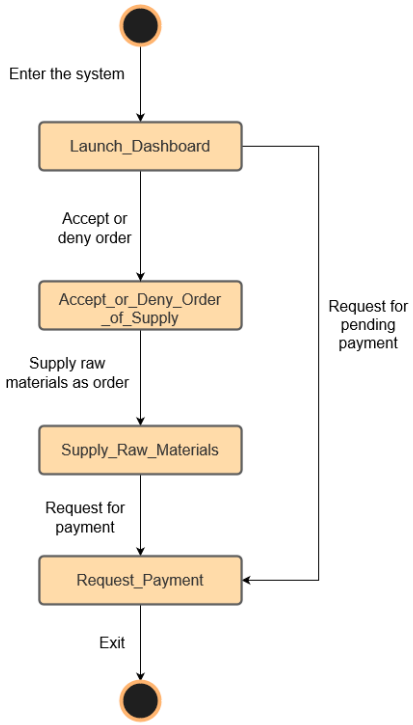


Figure 63 : State Transition Diagram for Supplier Class

ID: 5

Name: Retailer

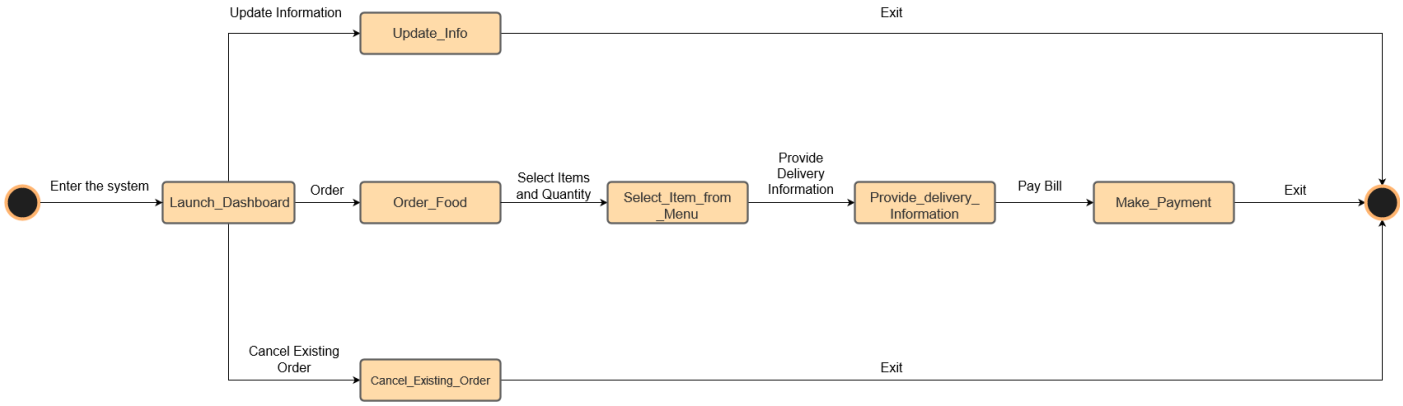


Figure 64 : State Transition Diagram for Retailer Class

ID: 6

Name: Customer

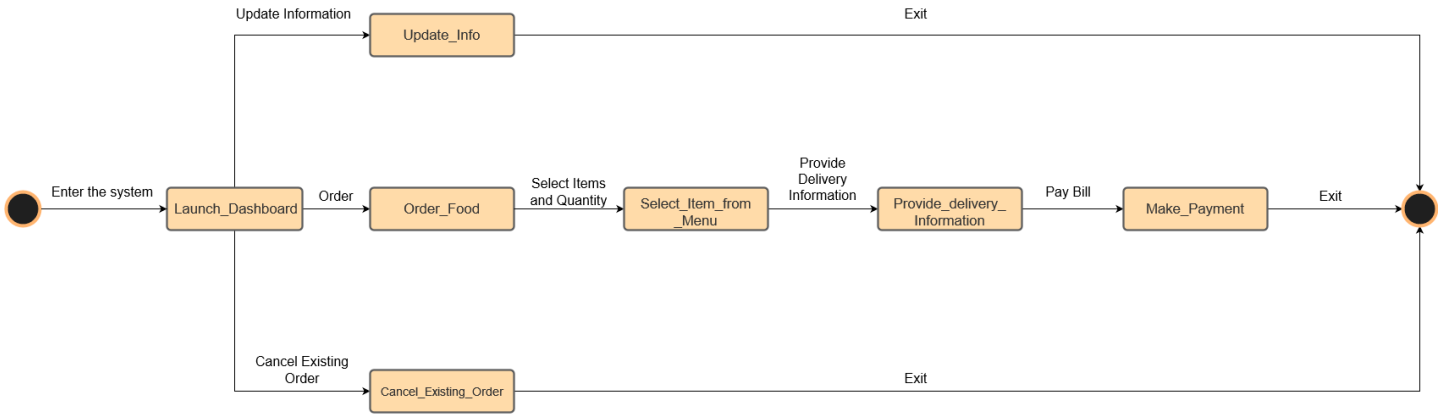


Figure 65 : State Transition Diagram for Customer Class

ID: 7

Name: Employee

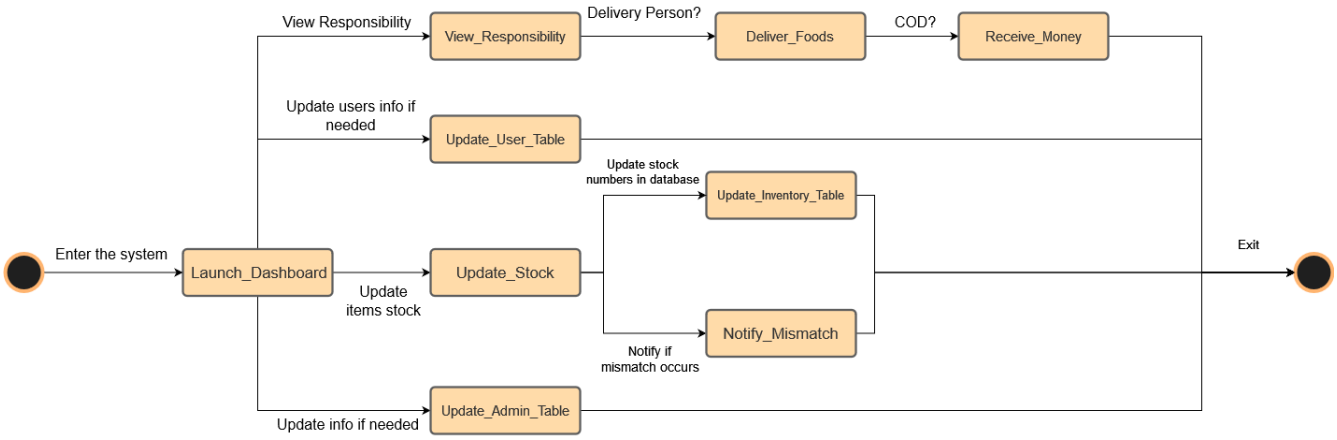


Figure 66 : State Transition Diagram for Employee Class

ID: 8

Name: Order

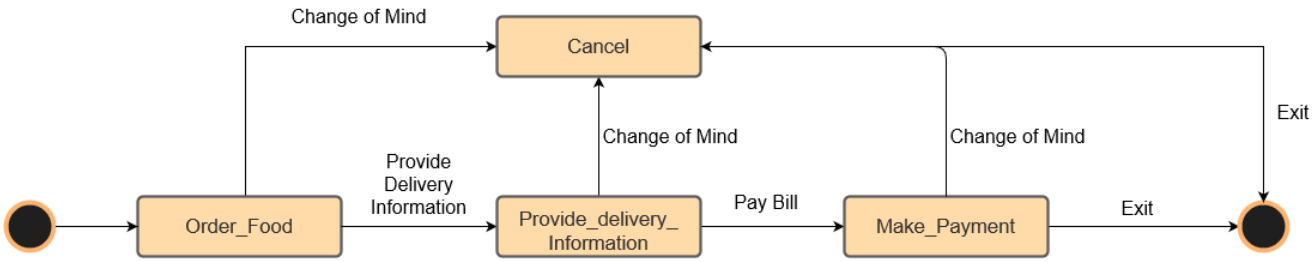


Figure 71 : State Transition Diagram for Order Class

ID: 9

Name: SSLCommerz

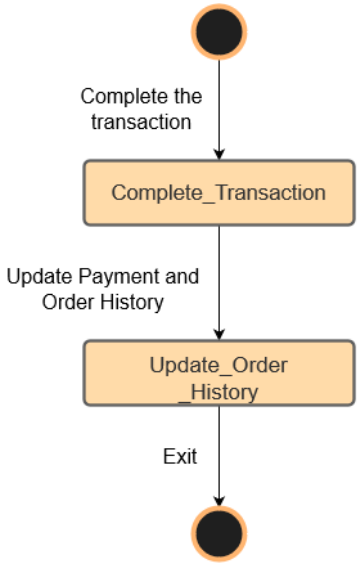


Figure 67 : State Transition Diagram for SSLCommerz Class

ID: 10

Name: Transaction

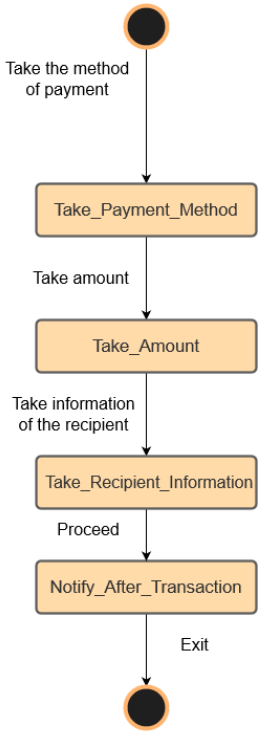


Figure 68 : State Transition Diagram for Transaction Class

ID: 11

Name: SMS

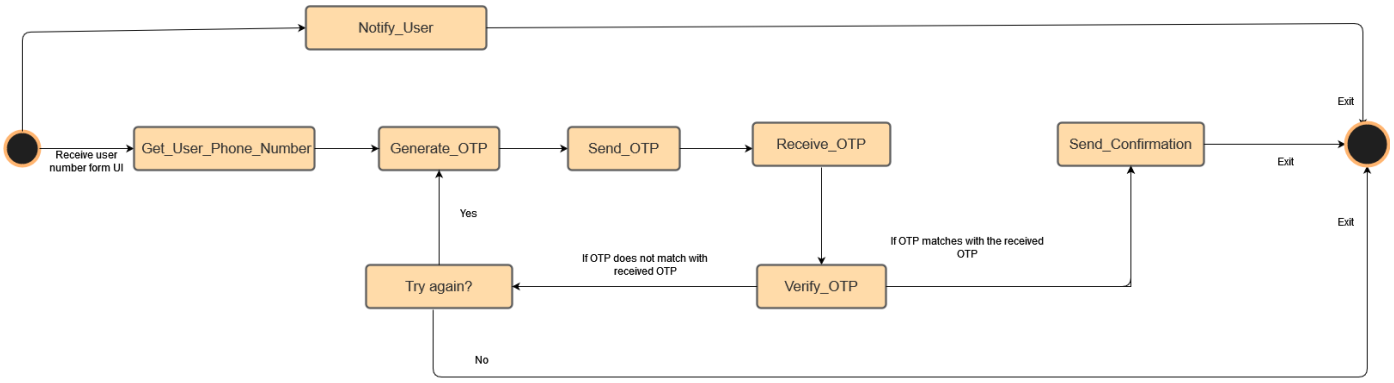


Figure 69 : State Transition Diagram for SMS Class

ID: 12

Name: GPS

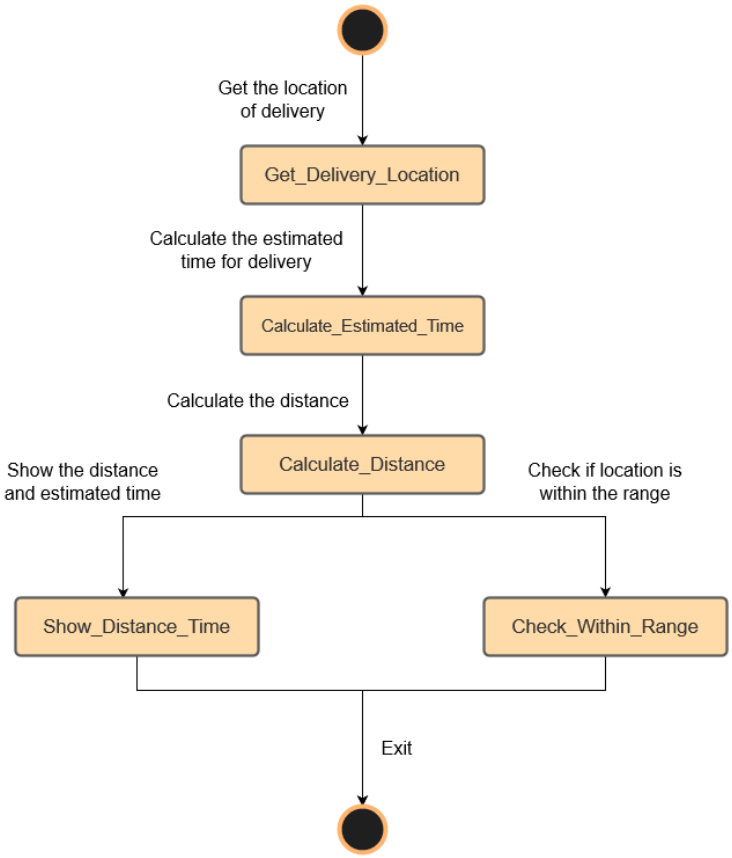
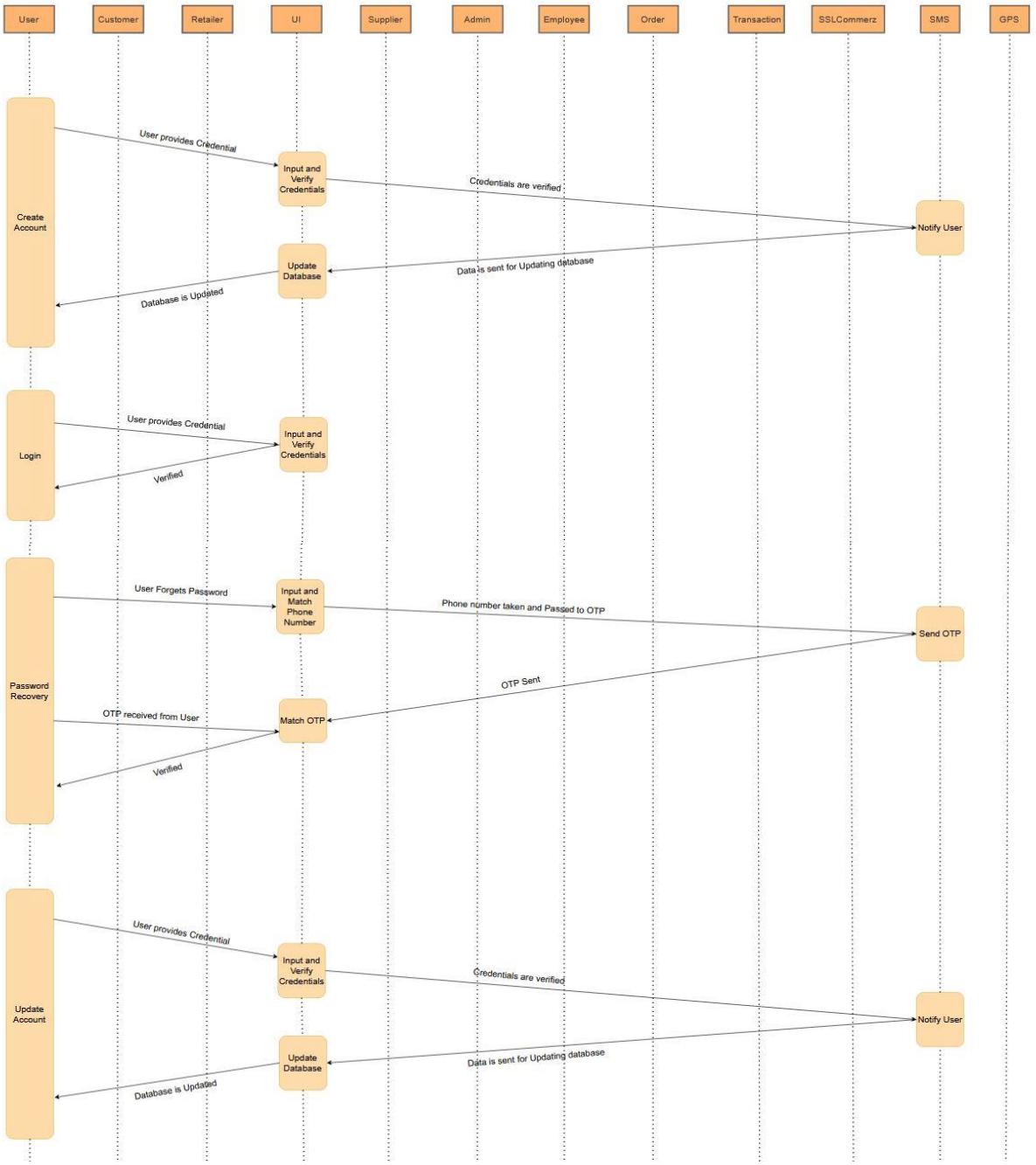
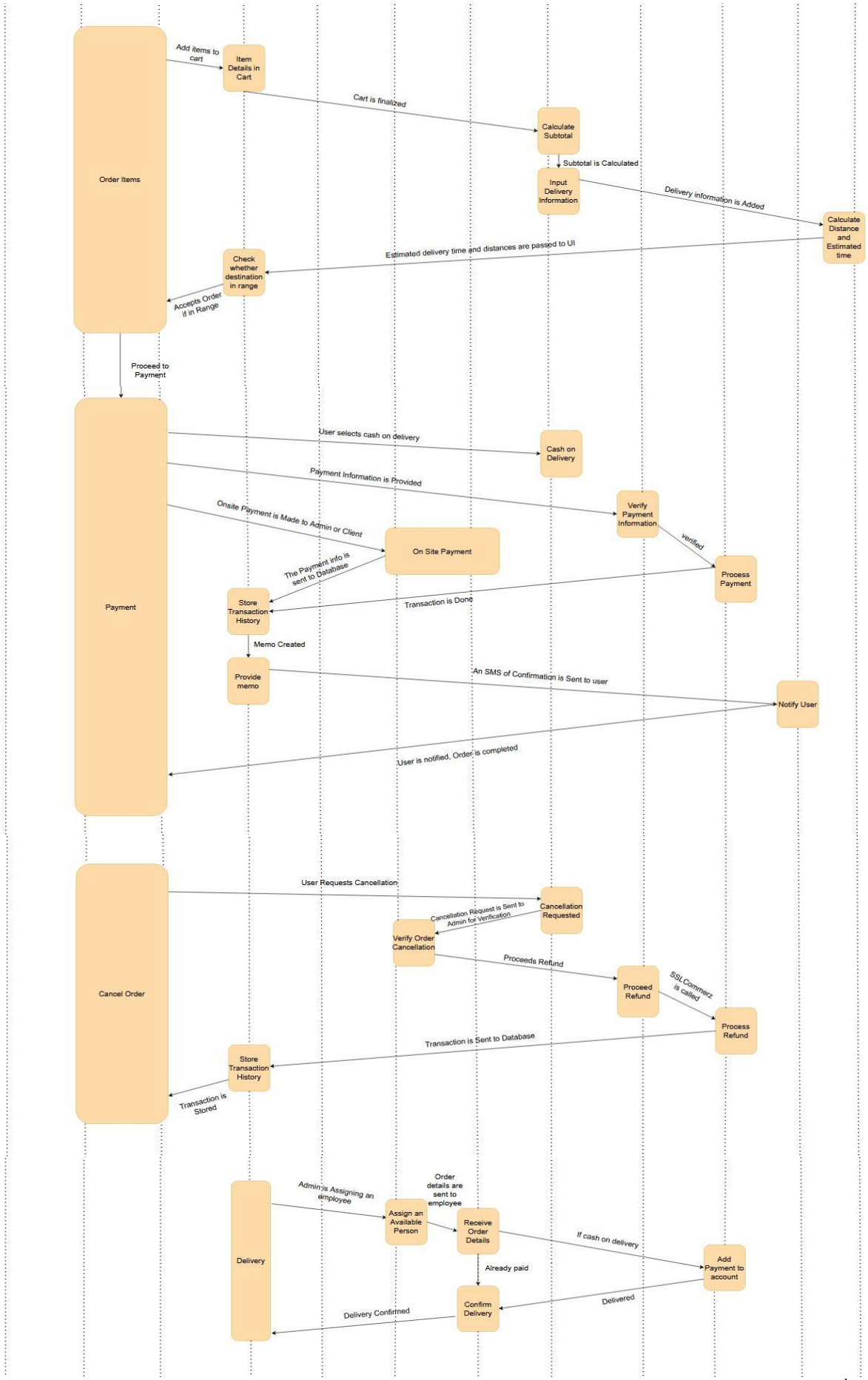


Figure 70 : State Transition Diagram for GPS Class

Sequence Diagram

The second type of behavioral representation, called a sequence diagram in UML, is a representation of how events cause flow from one object to another as a function of time. In essence, the sequence diagram is a shorthand version of the use case. It represents key classes and the events that cause behavior to flow from class to class.




Order Items
Add items to cart
Item Details in Cart
Cart is finalized
Calculate Subtotal
Subtotal is Calculated
Input Delivery Information
Delivery information is Added
Calculate Distance and Estimated time
Estimated delivery time and distances are passed to UI
Check whether destination in range
Accepts Order if in Range
Proceed to Payment
Payment
User selects cash on delivery
Cash on Delivery
Payment Information is Provided
Verify Payment Information
verified
Process Payment
Onsite Payment is Made to Admin or Client
On Site Payment
The Payment info is sent to Database
Store Transaction History
Transaction is Done
Memo Created
Provide memo
An SMS of Confirmation is Sent to user
Notify User
User is notified, Order is completed
Cancel Order
User Requests Cancellation
Cancellation Requested
Cancellation Request is Sent to Admin for Verification
Verify Order Cancellation
Proceeds Refund
Proceed Refund
SSLCommerz is called
Process Refund
Transaction is Sent to Database
Store Transaction History
Transaction is Stored
Delivery
Admin is Assigning an employee
Order details are sent to employee
Assign an Available Person
Receive Order Details
If cash on delivery
Add Payment to account
Already paid
Confirm Delivery
Delivered
Delivery Confirmed

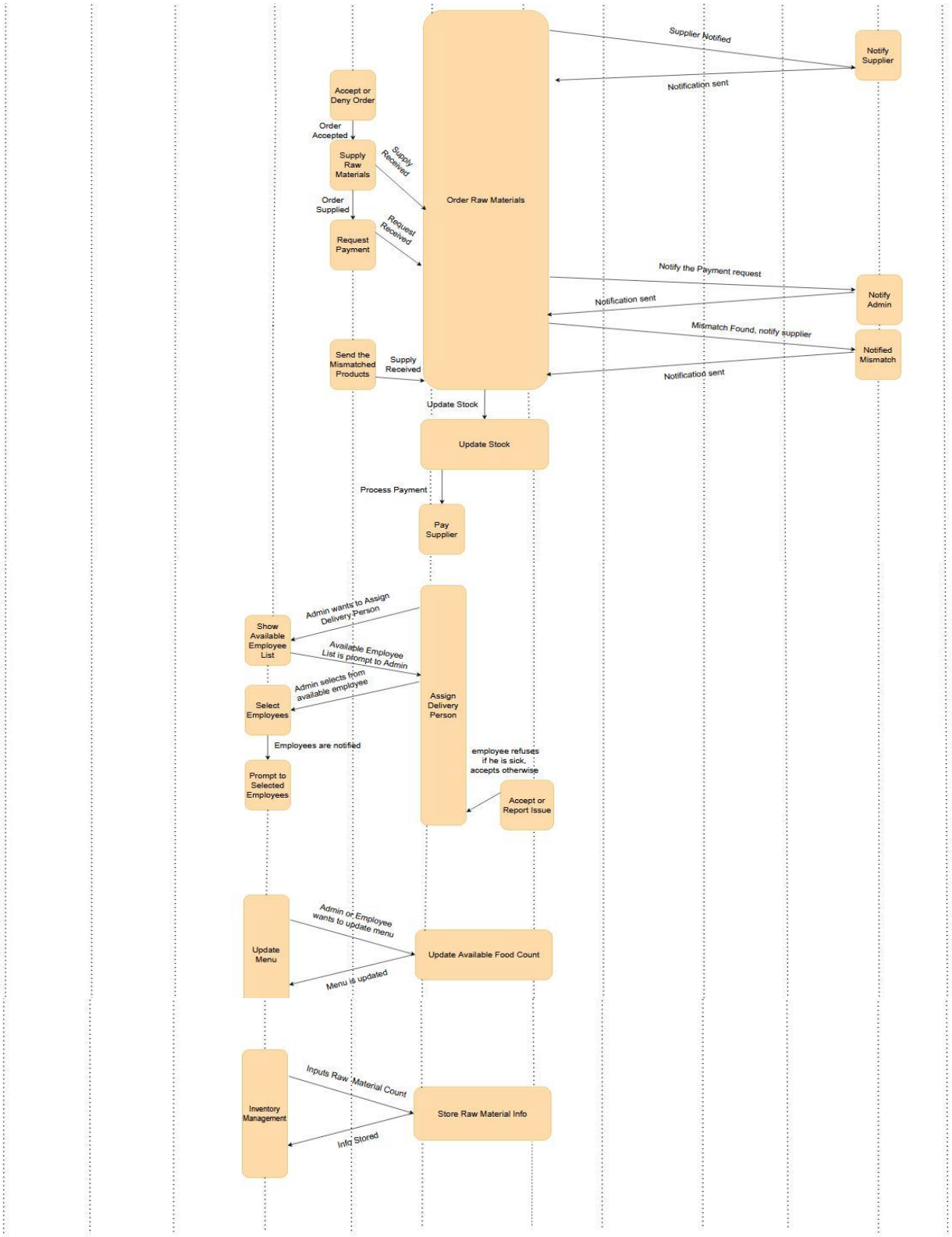


Figure 71 : Sequence Diagram for BMS