

# **Report on Hunger Express**

**An online food delivery system**

**Group: A2\_Group4**

**Submitted By**

1.1705031

2.1705034

3.1705041

4.1705049

5.1705054

6.1705059

## **Introduction:**

Our project ‘Hunger Express’ is a food delivery app. Here lots of restaurants will be signed up and they can add their menu items on the app. The registered customers will be able to see those menu items in the app and can order any of the food. Customers can search food either by food type or name, or they can go through the menu of a specific restaurant. Here we tried to keep all the things as simple as possible. After placing an order, the restaurant will start processing the order and then a rider will be assigned who will deliver the food to the customer.

Unlike the currently available food delivery services, we have added some special features in our app. One of them is pre order. A customer can pre order when he thinks he'd need a vast amount of food for a special occasion. Or a service holder who needs a daily meal at 1pm every day, can make pre order for a month so that he doesn't have to worry about his lunch.

Another special feature is rush order. Suppose some guests just came over to our house without any notice. How can we treat them within this very short time? Hunger Express is the ultimate solution here. A customer can rush order any food that is available for rush order and the food will reach his home within the fastest period of time.

Along with these and normal order, we also added pickup order which will be beneficial to both the customer and the restaurant. We have added another feature named promo/voucher with which a customer can get discount if he orders food from this app. Moreover, we also have a coin system. The more a customer orders from this app, the more his coin will grow. And thus, he'll be able to redeem these coins to a special voucher of a large amount of discount.

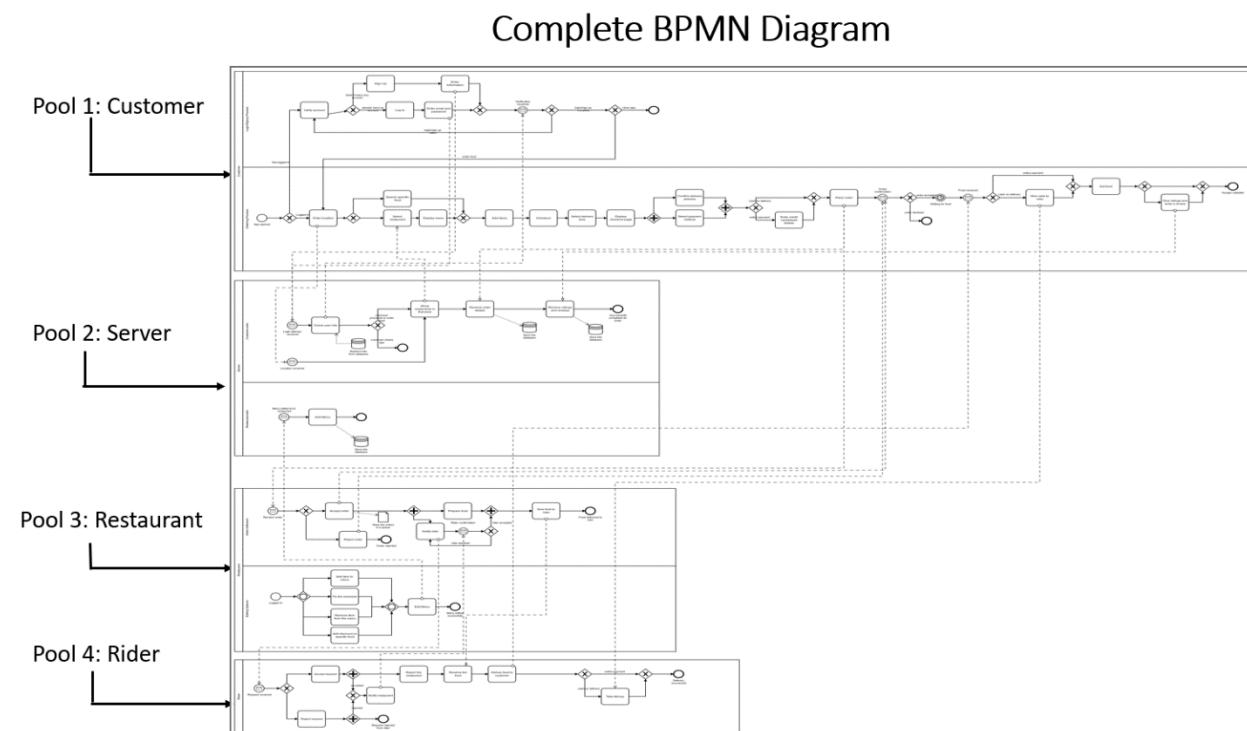
## Motivation:

We wanted to create an app where a user can easily order for a food and the delivery man will deliver the favorite food from different restaurants to the doorstep of the consumers as soon as possible. It will eventually both saves and money for a customer as they don't have to go outside to purchase item. Also it creates employment for so many unemployed people. It helps restaurants to grow their business and reach a lot of customers through internet. Cost effective promotion and a lot more service than dine in.

# Requirement Analysis using BPMN diagram:

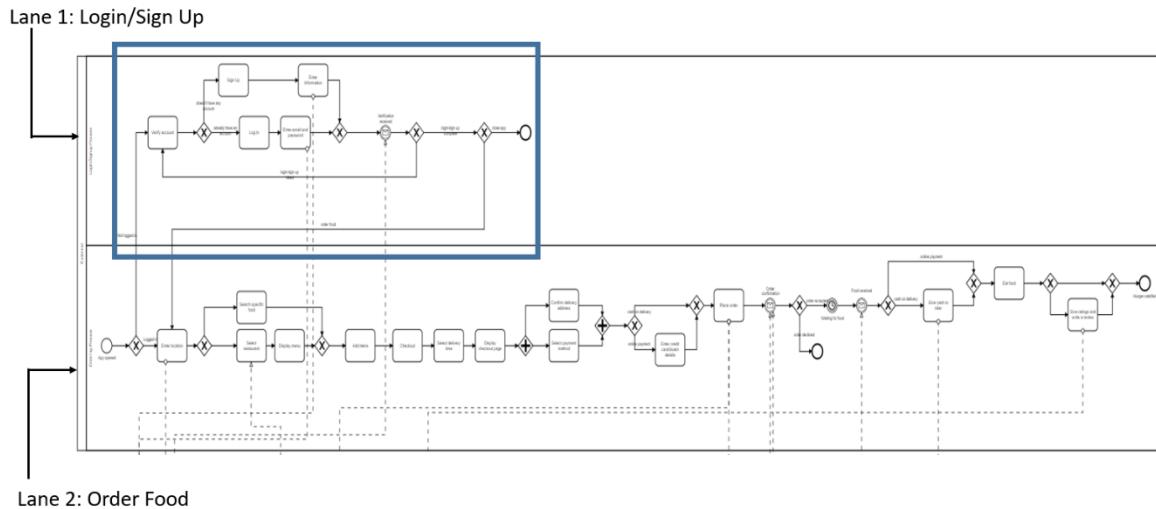
BPMN (Business Process Modeling & Notation) is a graphical representation of business process using simple objects, which helps the organization to communicate in a standard manner.

In our BPMN diagram, we have 4 pools in total each of which has one or two lanes. The four major pools are – customer, server, restaurant and rider.

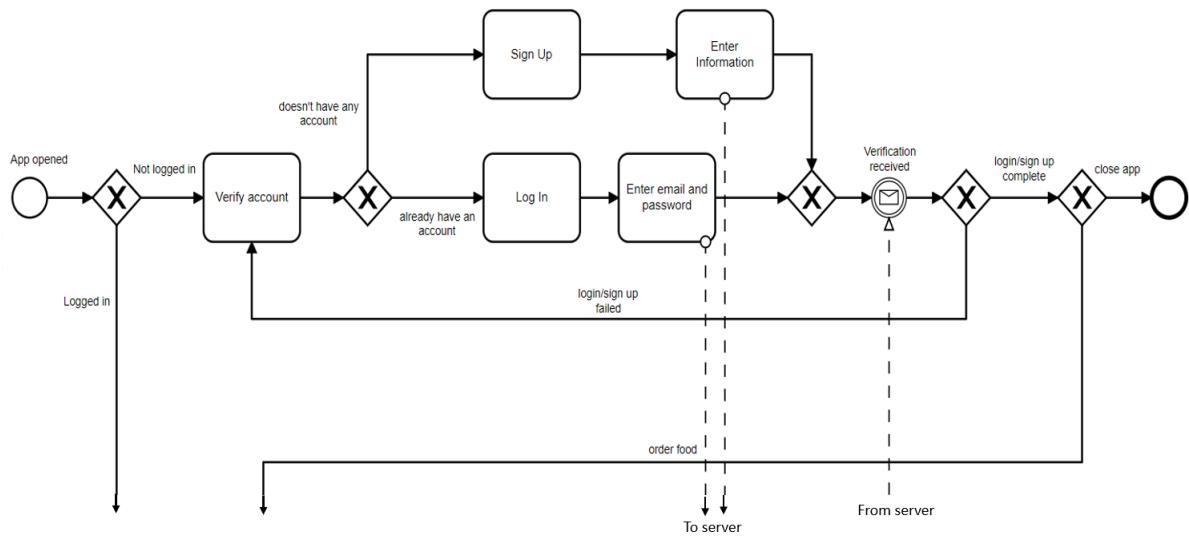


The customer pool has two lanes named ‘login/signup’ and ‘order food’.

### Pool 1: Customer

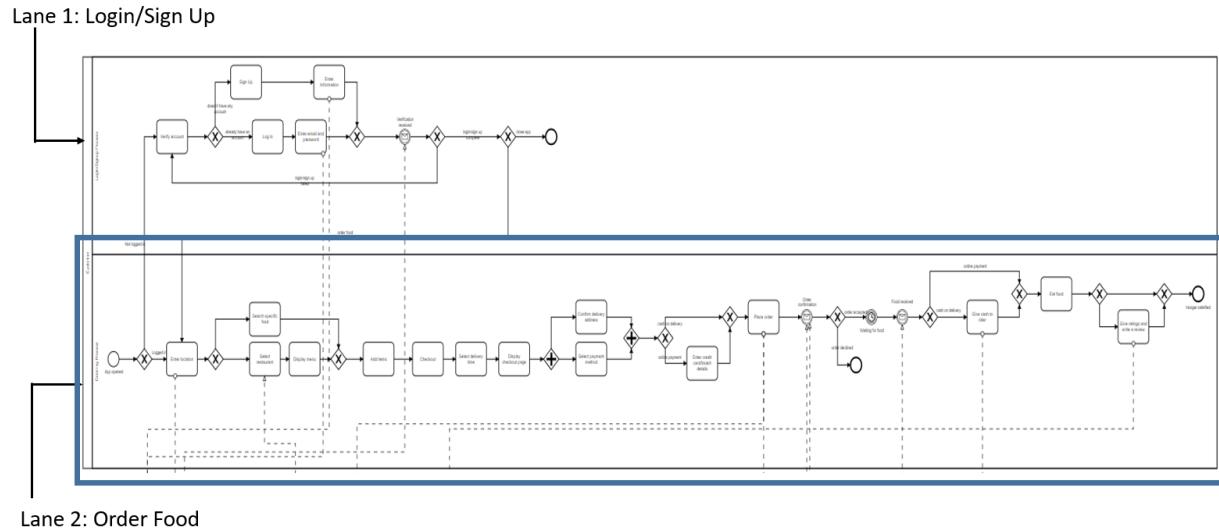


### Pool 1: Customer - Lane 1: Login / Sign Up

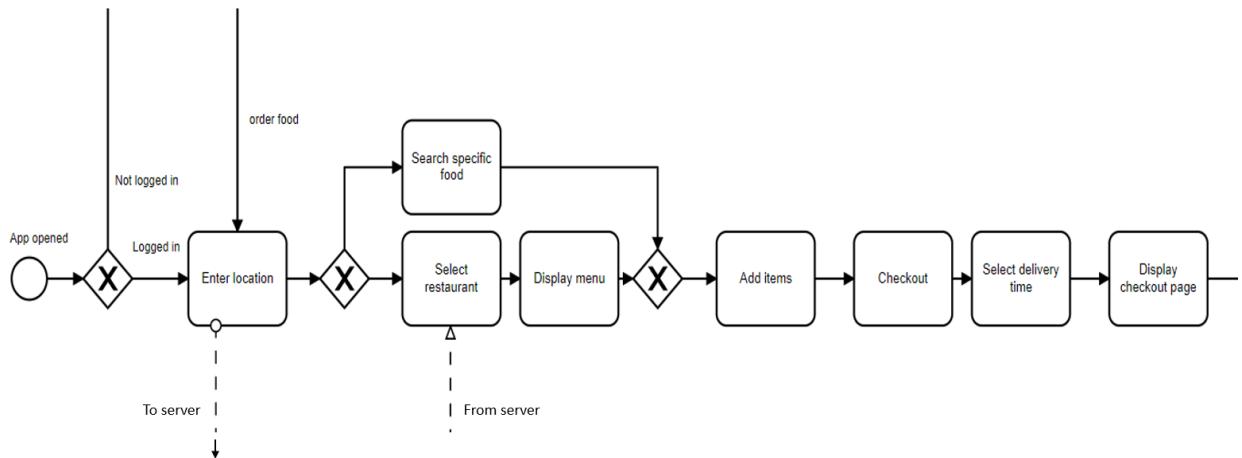


This login/signup lane shows the steps of how a customer can login or signup to his account.

## Pool 1: Customer

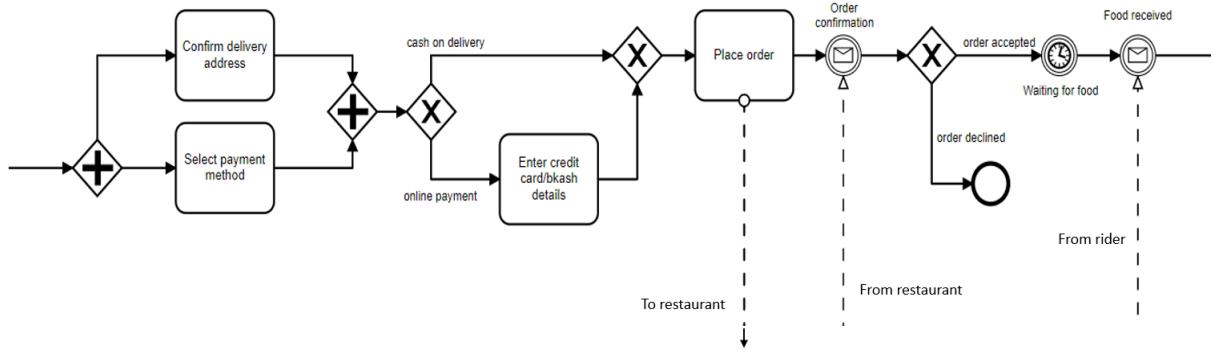


## Pool 1: Customer - Lane 2: Order Food

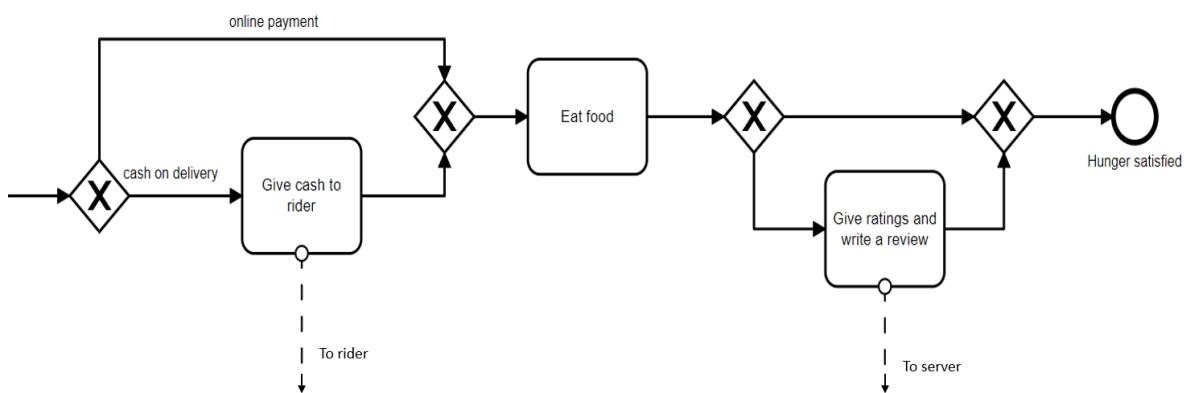


This ‘order food’ lane describes the steps of how a customer can order a specific food either by searching by food name or by restaurant. As this lane is too large, we broke it into 3 pieces.

## Pool 1: Customer - Lane 2: Order Food(Continued..)



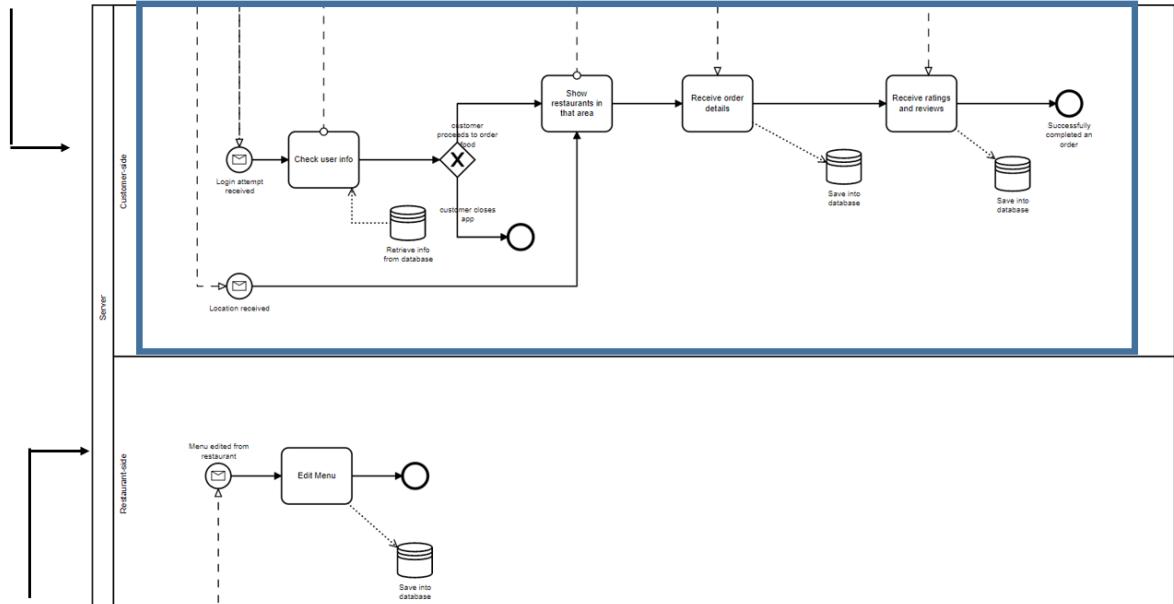
## Pool 1: Customer - Lane 2: Order Food(Continued..)



The server pool has also two lanes. One is ‘customer-side’ and the other is ‘restaurant-side’. As we discussed login/signup in the customer section, we didn’t show it here in the restaurant section.

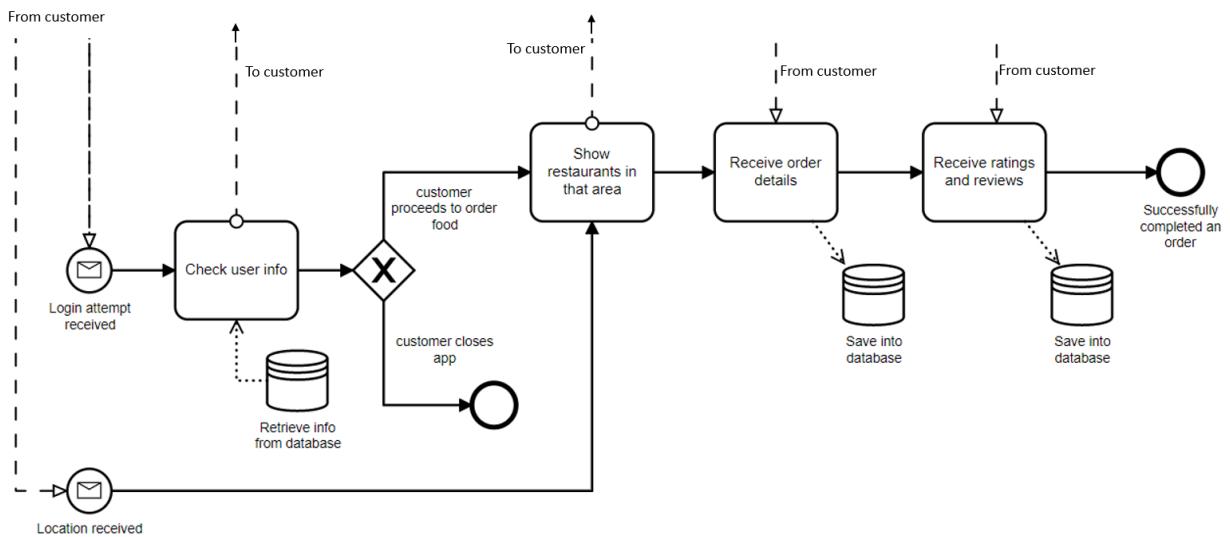
## Pool 2: Server

Lane 1: Customer side



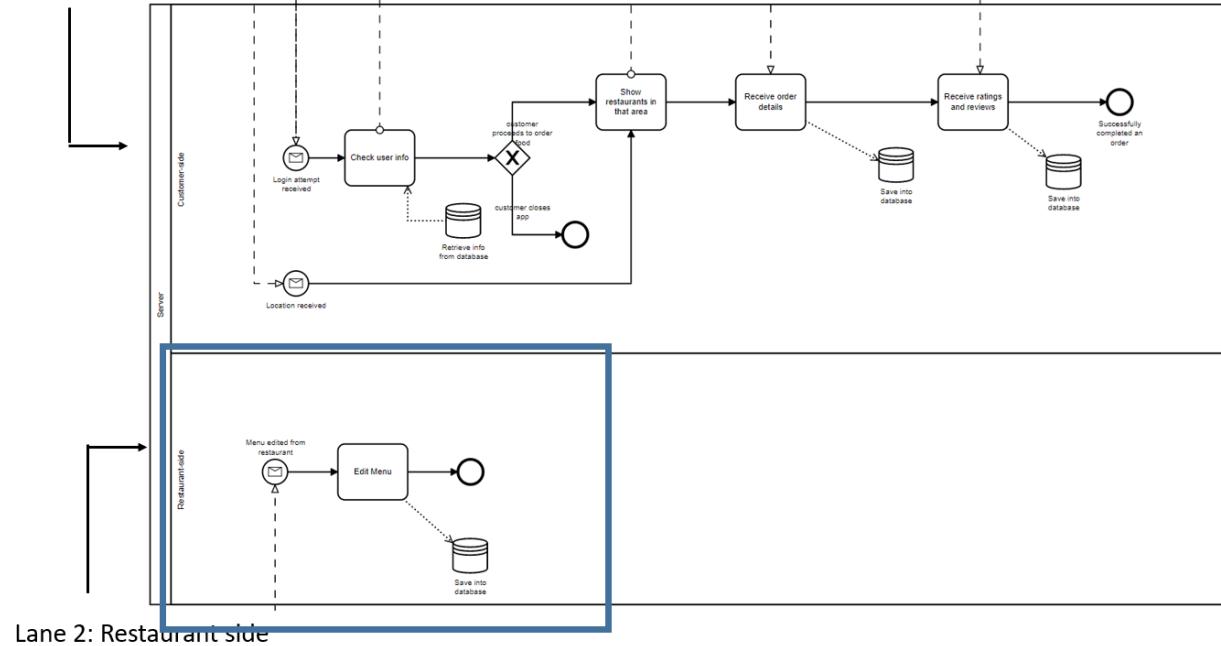
Lane 2: Restaurant side

## Pool 2: Server - Lane 1: Customer side



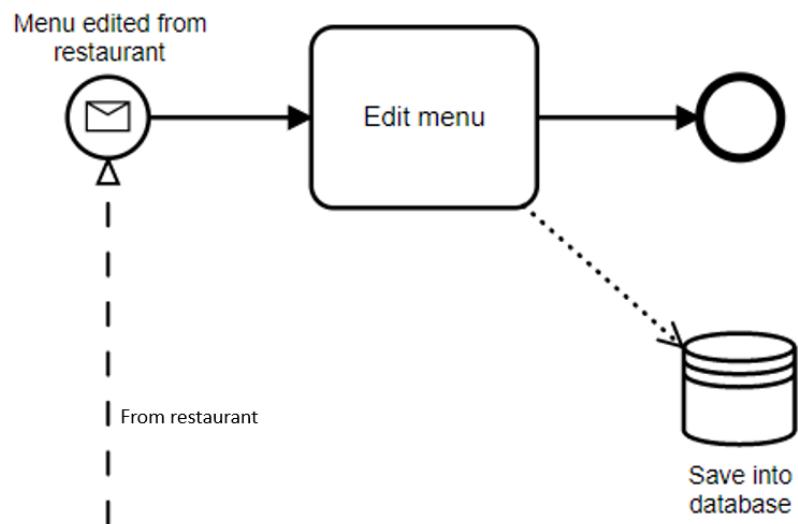
## Pool 2: Server

Lane 1: Customer side



Lane 2: Restaurant side

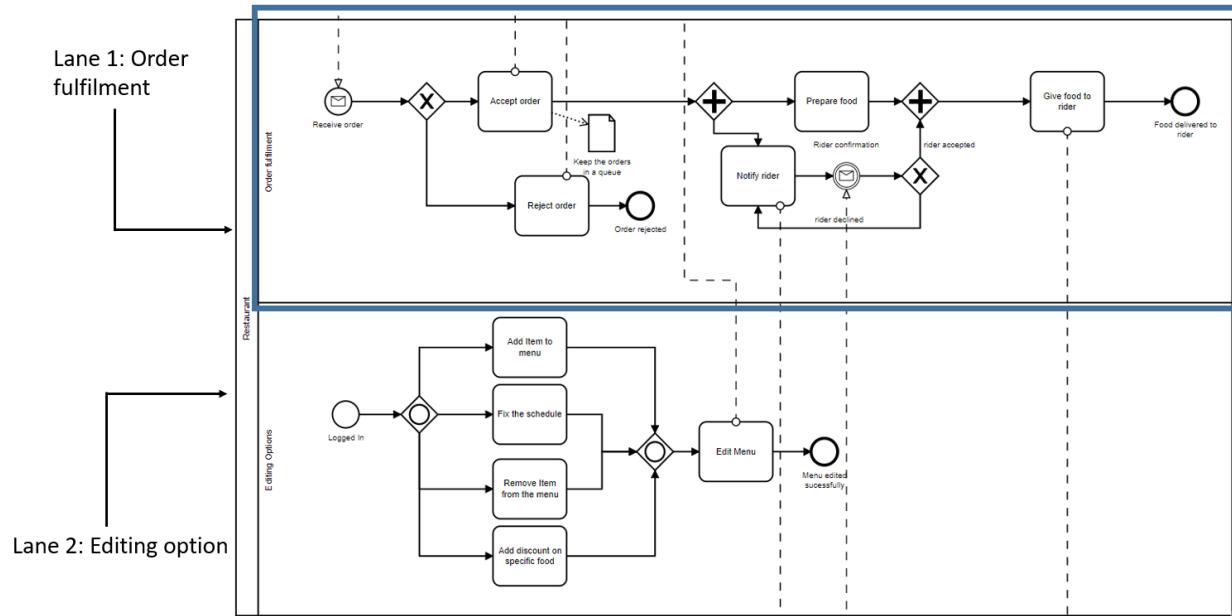
## Pool 2: Server - Lane 2: Restaurant side



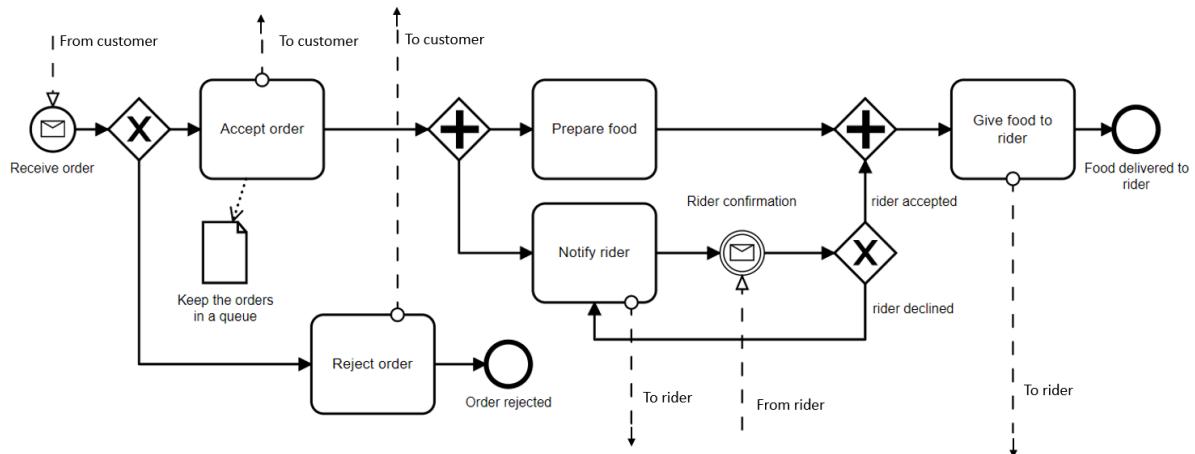
These two lanes of server pool show the interaction of the server with customer and restaurant accordingly.

We have two lanes in restaurant pool – ‘order fulfilment’ and ‘editing option’.

### Pool 3: Restaurant

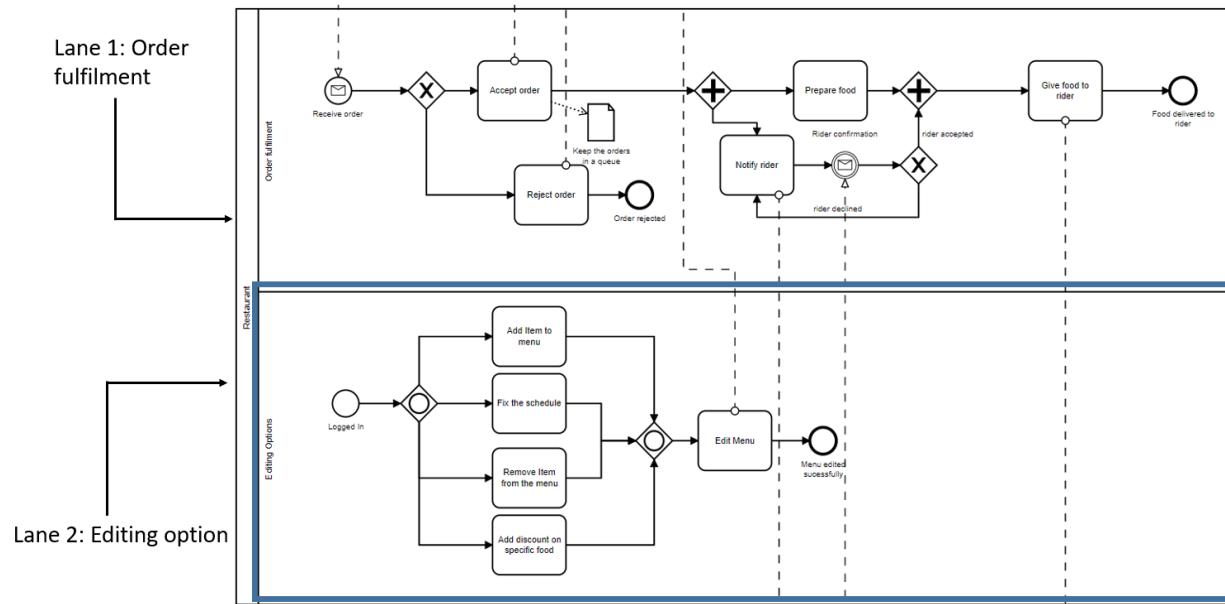


### Pool 3: Restaurant - Lane 1: Order fulfilment

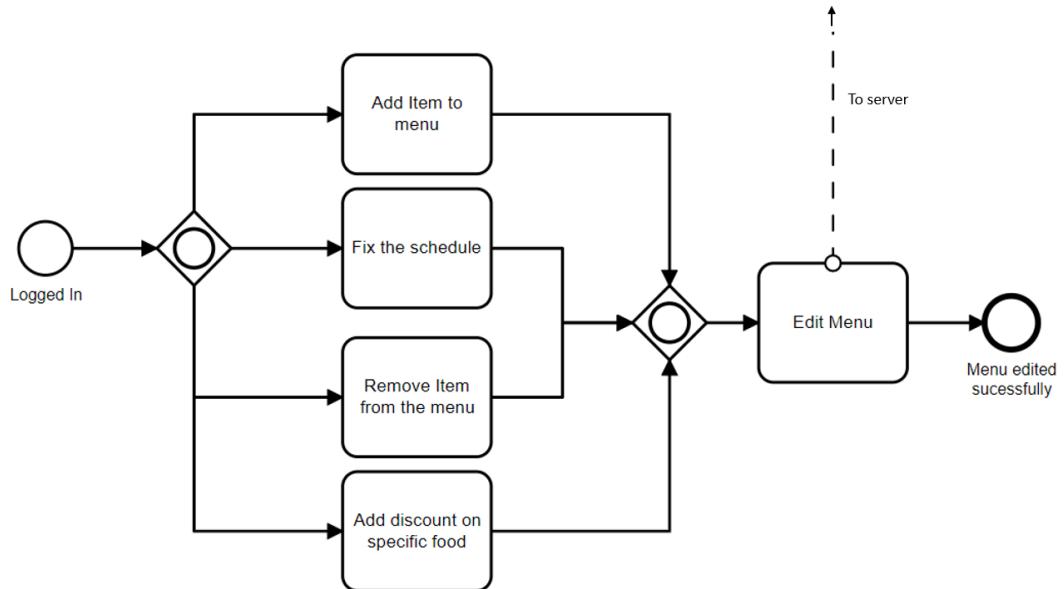


This ‘order fulfilment’ lane shows the steps of how a restaurant manager acts when an order is placed to his restaurant.

### Pool 3: Restaurant



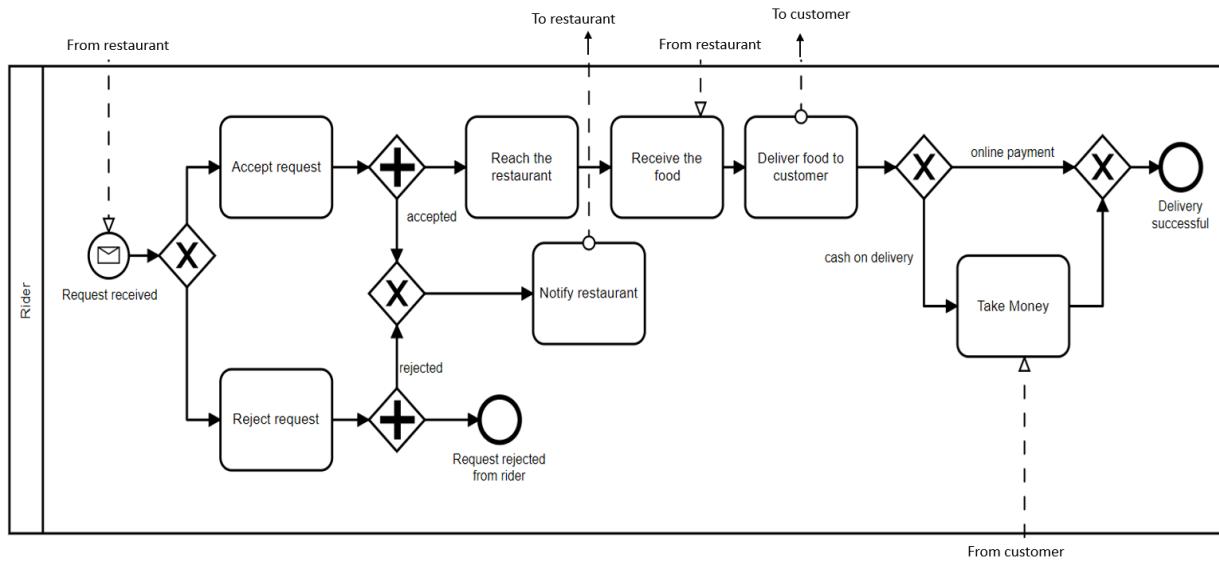
### Pool 3: Restaurant - Lane 2: Editing options



And this ‘editing options’ lane describes the way of how the restaurant manager manages the food items and the menu that is shared online.

And finally, in the rider pool, unlike the other pools, we have only one lane.

### Pool 4: Rider



### The feedback provided in the class for BPMN diagram:

In the BPMN diagram, we tried to show all the activities performed in the proper order and the interactions among the pools. But there were some flaws which were pointed by the respected course teachers. One of their feedbacks was that in the rider pool, we could add an end event after ‘notify restaurant’ activity in case of a rejected request. Another feedback was to check if a rider is available before taking an order in the restaurant pool. Moreover we could maintain a list to pass more than one order at a time to a rider if the orders are from a nearby area. And lastly, we should have added the order types for handling rush orders and preorders.

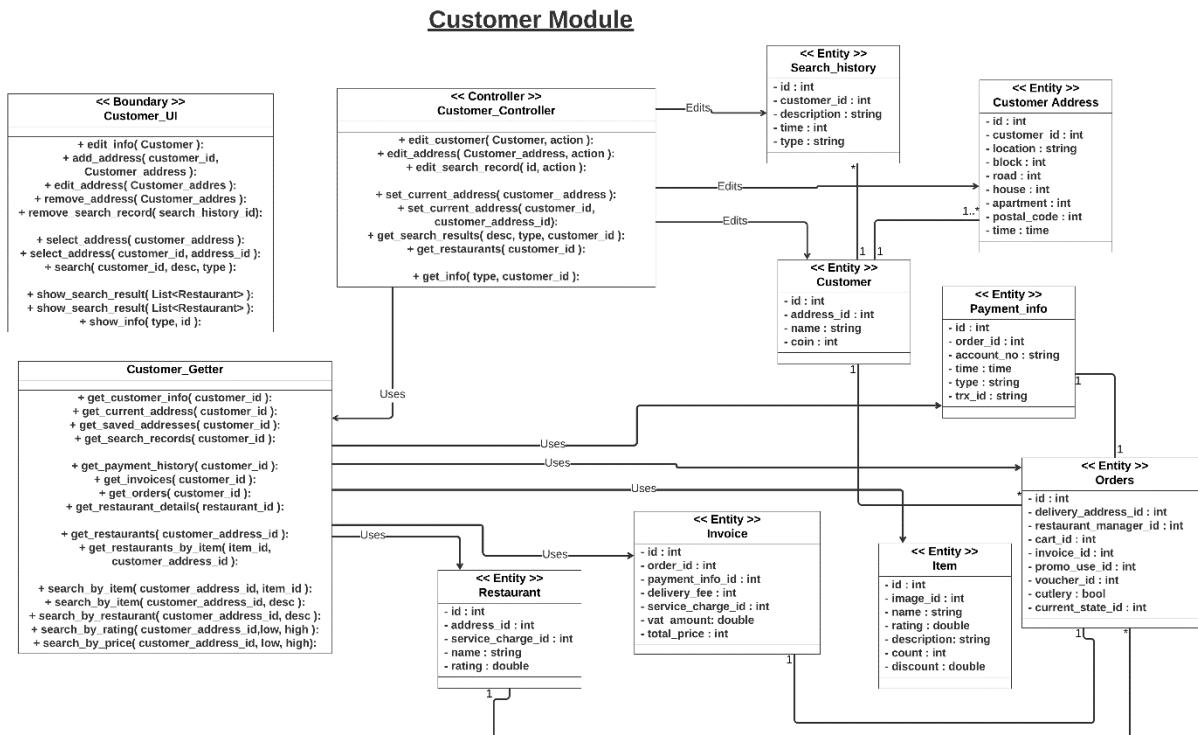
## Class Diagram:

There are total 8 diagram in our class diagram-

- Customer module
  - Menu/Cart module
  - Order module
  - Review module
  - Coin module
  - Restaurant module
  - Promo module
  - Vat/Address module

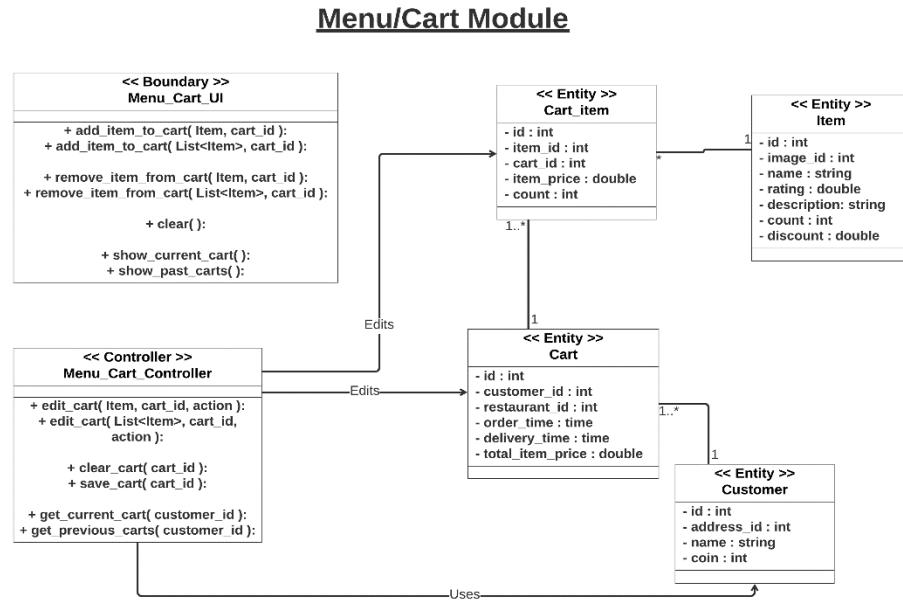
## Customer Module

This diagram shows class needed for customer operations and search operations. This diagram excludes order related operations of a customer.



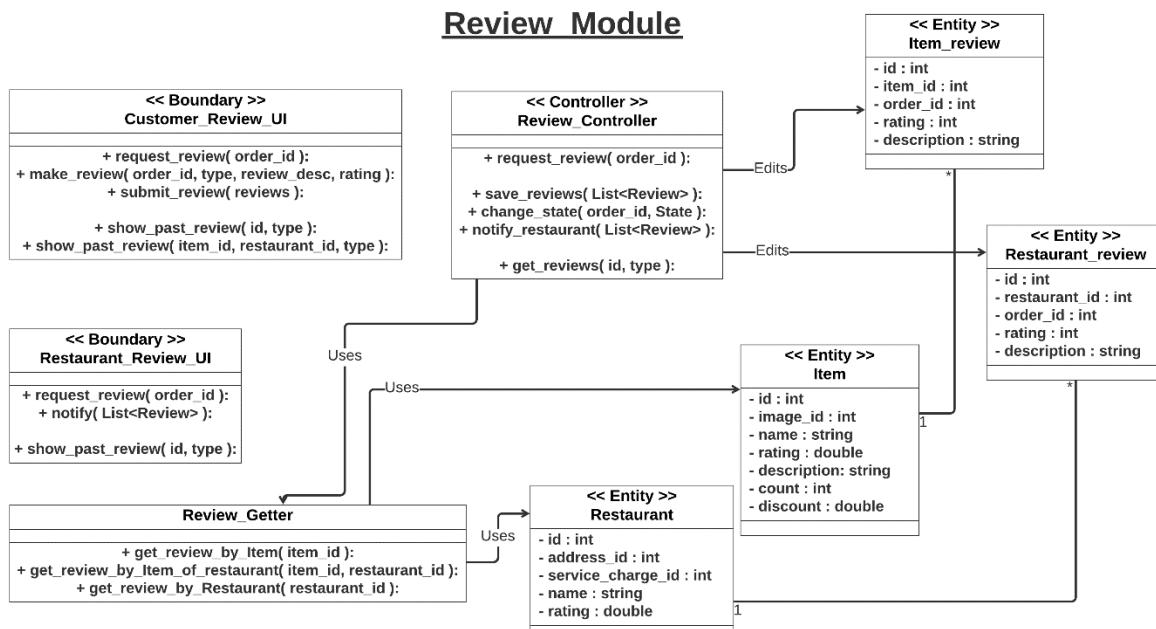
## Menu/Cart Module

This diagram shows classes related to building up carts for orders in Customer end.



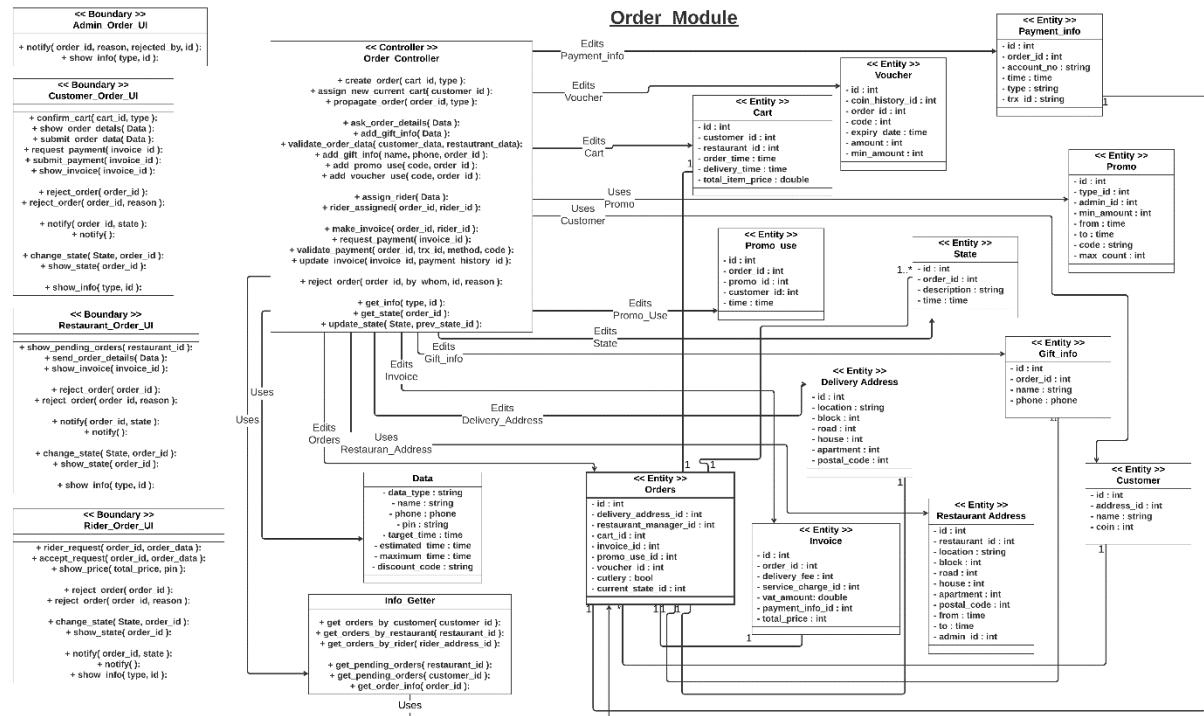
## Review Module

This diagram shows the classes need to make a review an order from customer end, viewing order history of a restaurant from both customer and restaurant end.



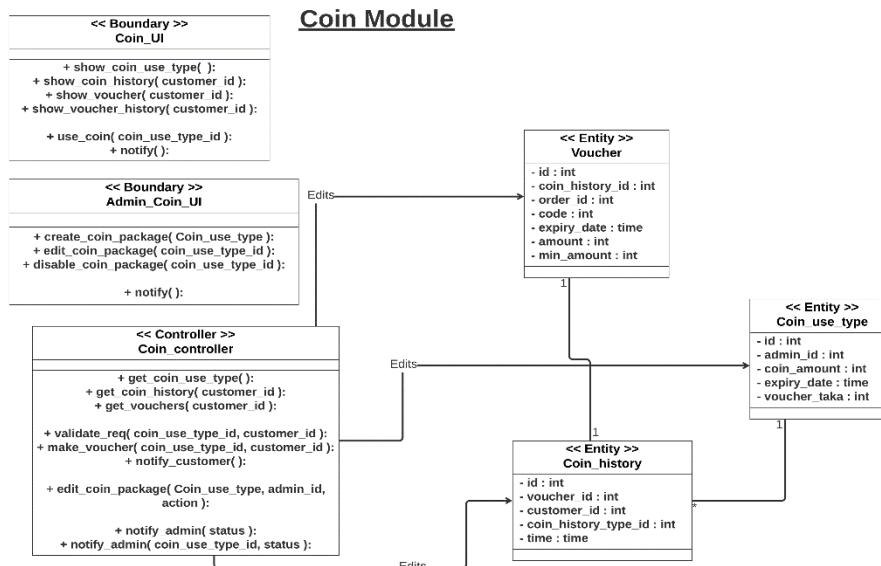
## Order Module

This diagram shows classes related from making an order to completing the order. This considers three ends of UI: restaurant end, customer end and rider end.



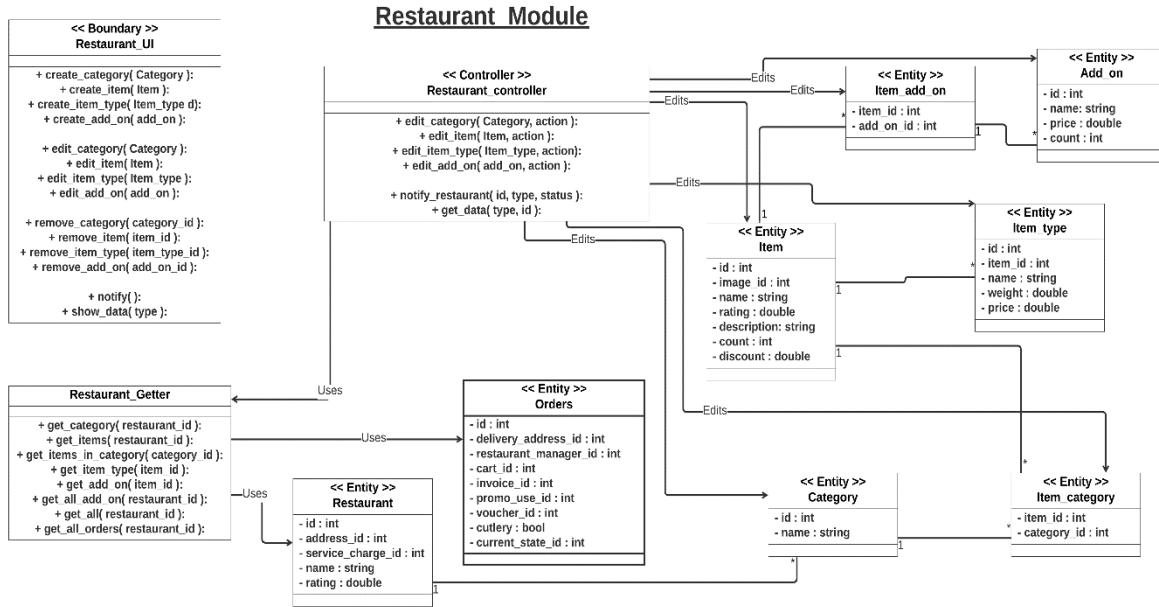
## Coin Module

This diagram shows classes for coin module.



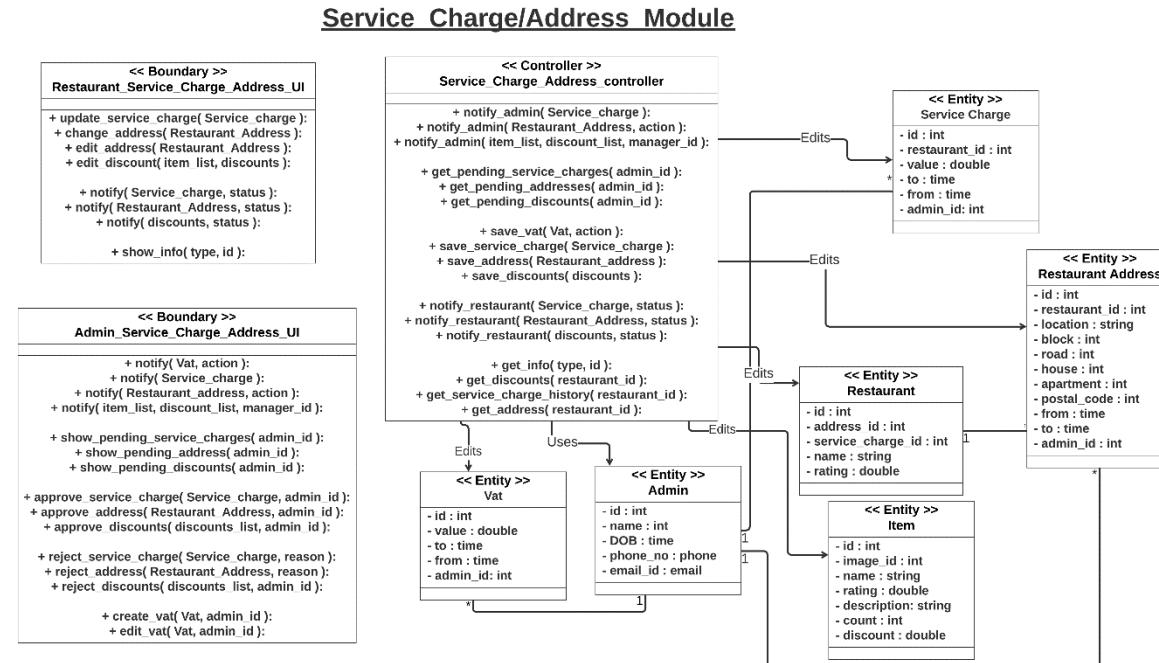
## Restaurant Module

This diagram shows classes of all operations for the restaurant end except operations related to orders.



## Service Charge/Address module

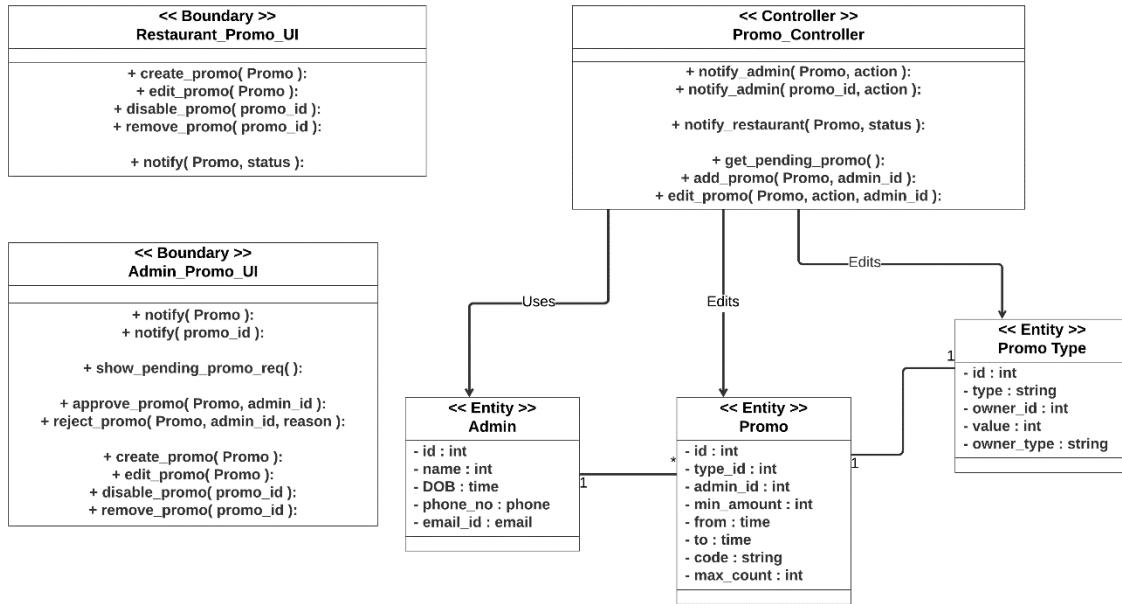
This diagram shows classes needed for restaurant end for operations related to service charge assignments or address edit operations.



## Promo Module

This diagram shows classes of promo module.

### Promo Module

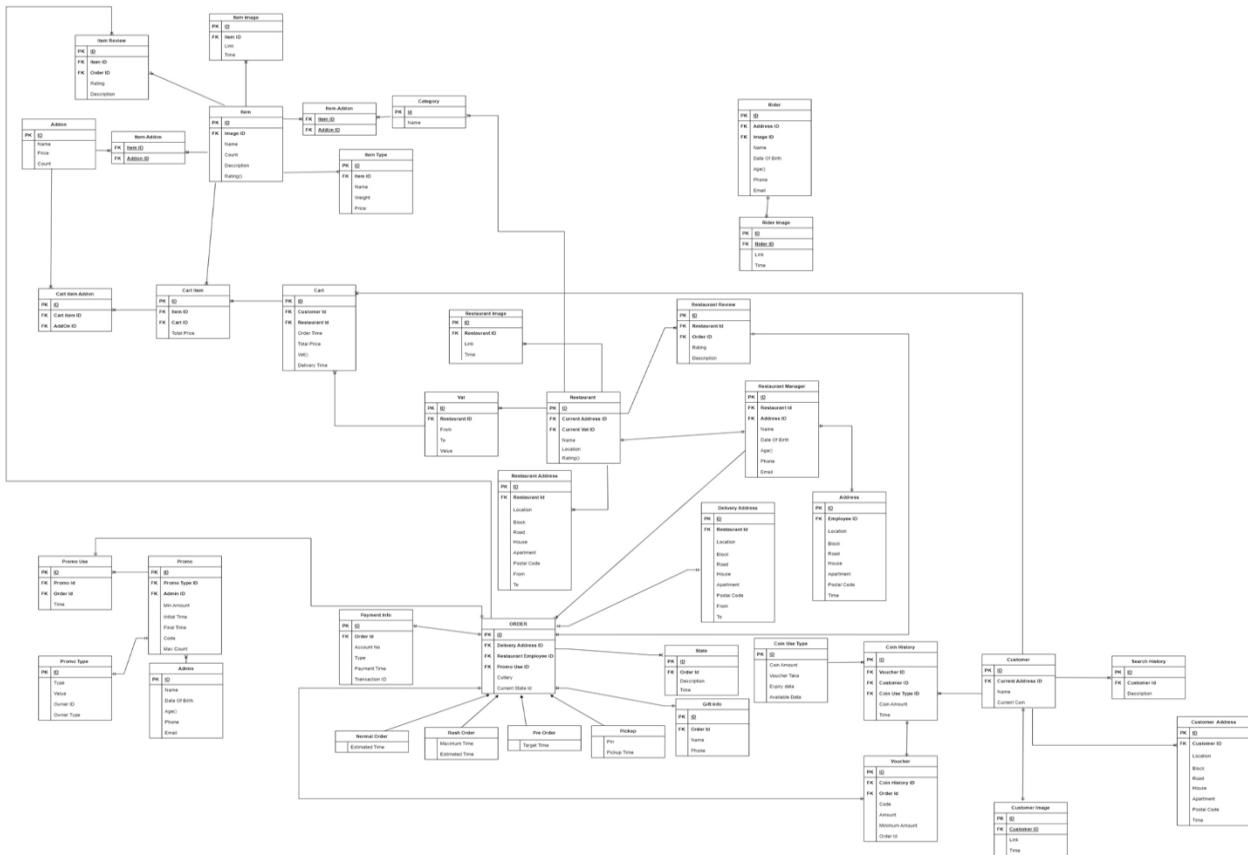


### The feedback provided in the class:

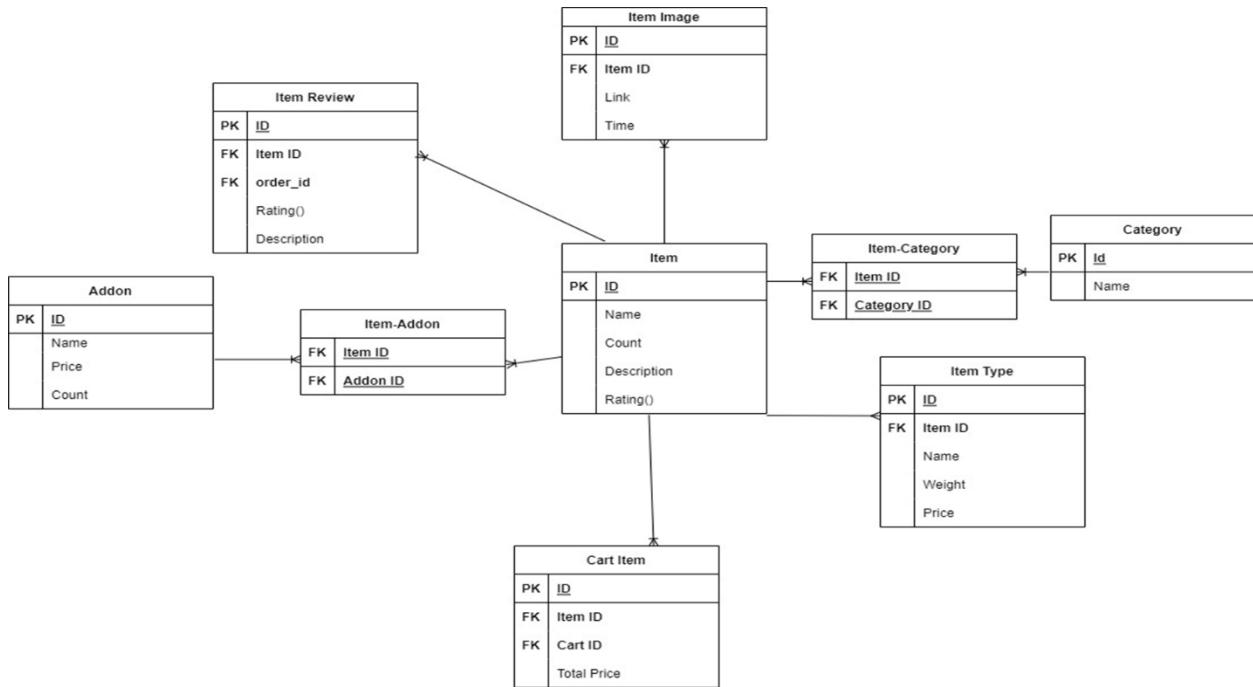
1. We were advised to add a variable related to total price in “Cart” entity class in “Menu/Cart” module. It has been added later and named as “total\_item\_price”.
2. We were advised to keep a same vat for all the restaurant. It has been edited later and we added service\_charge class in place of vat. The new “Vat” entity shows the change.

# ER Diagram:

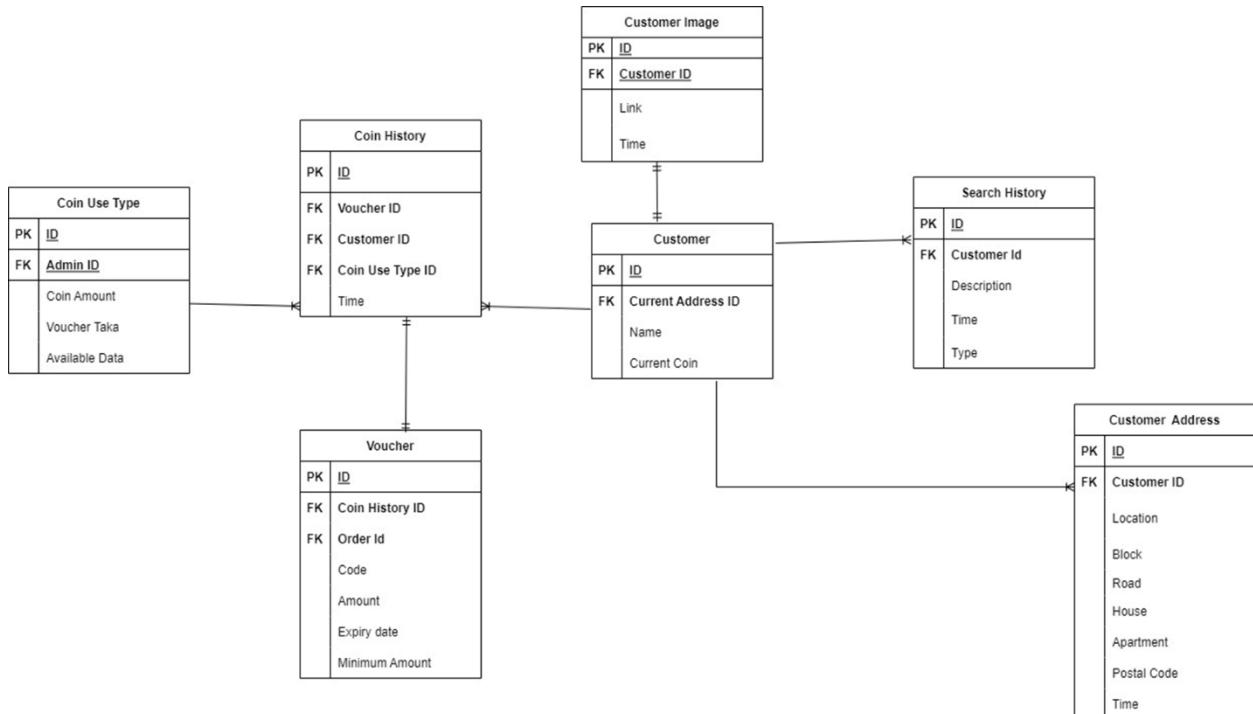
This is the complete Entity Relationship Diagram Of our project .There is a total of 40 entity.We have 7 modules in total.



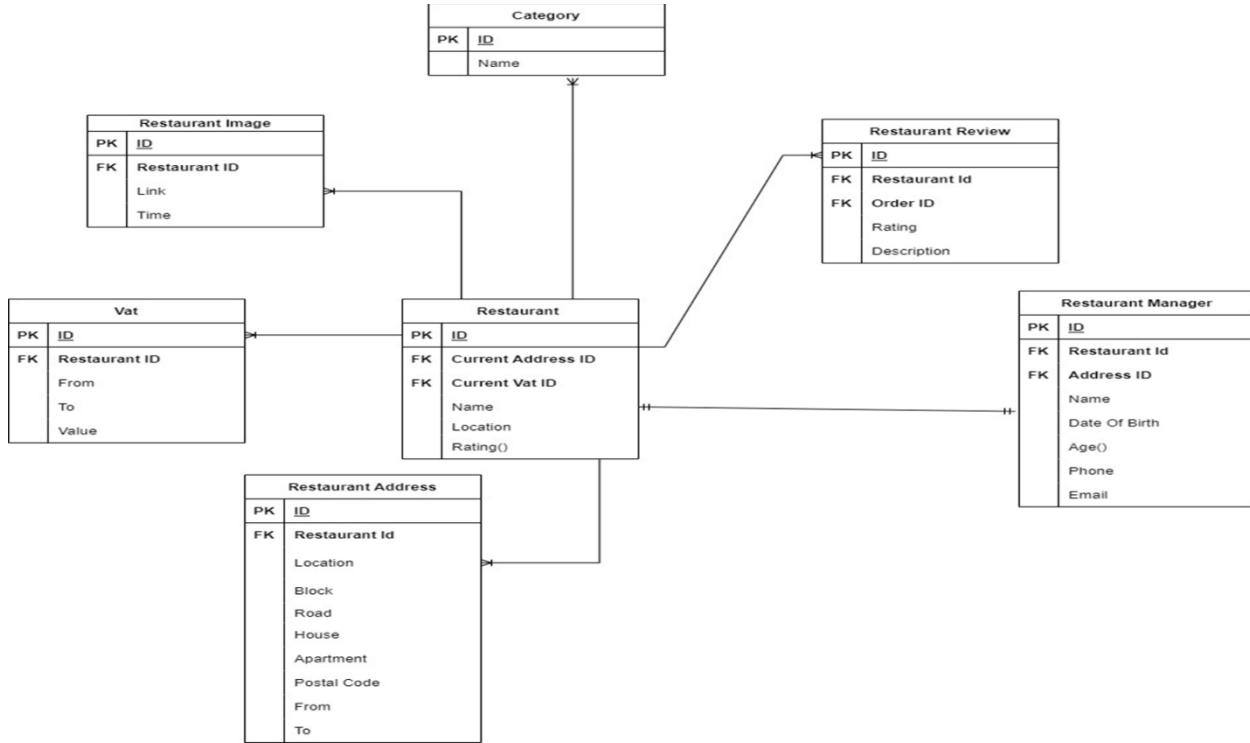
**Item module :** This module shows relation of item with category, addon ,cart,item image and item review.



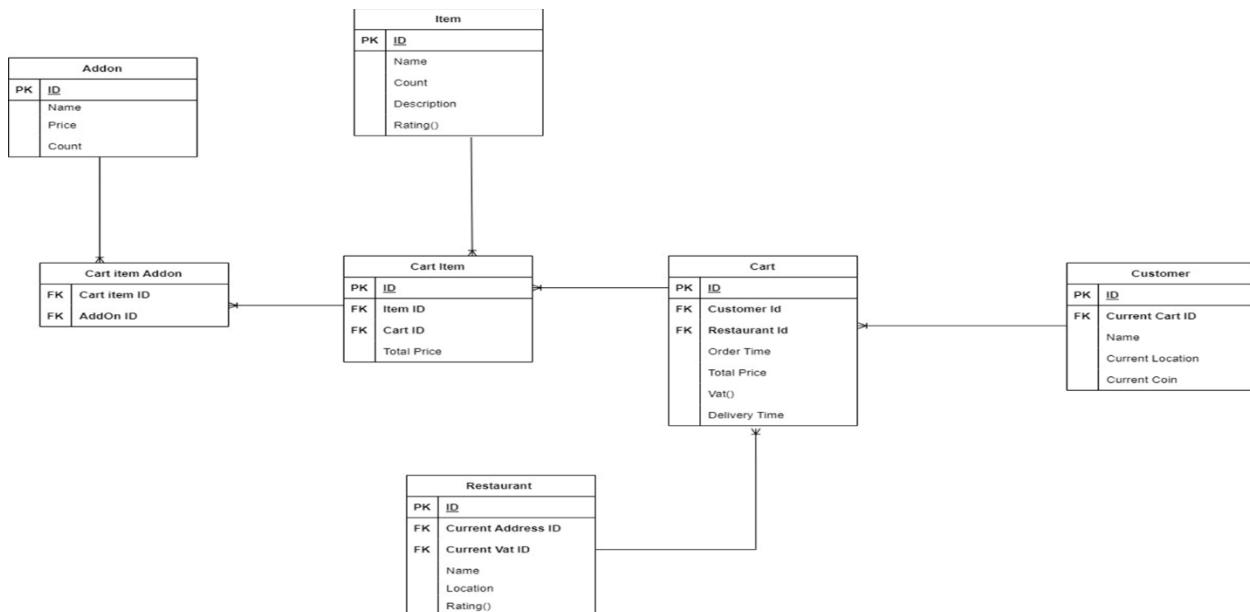
**Customer Module:** This module shows relation of customer with coin history,search history,customer address and customer image.



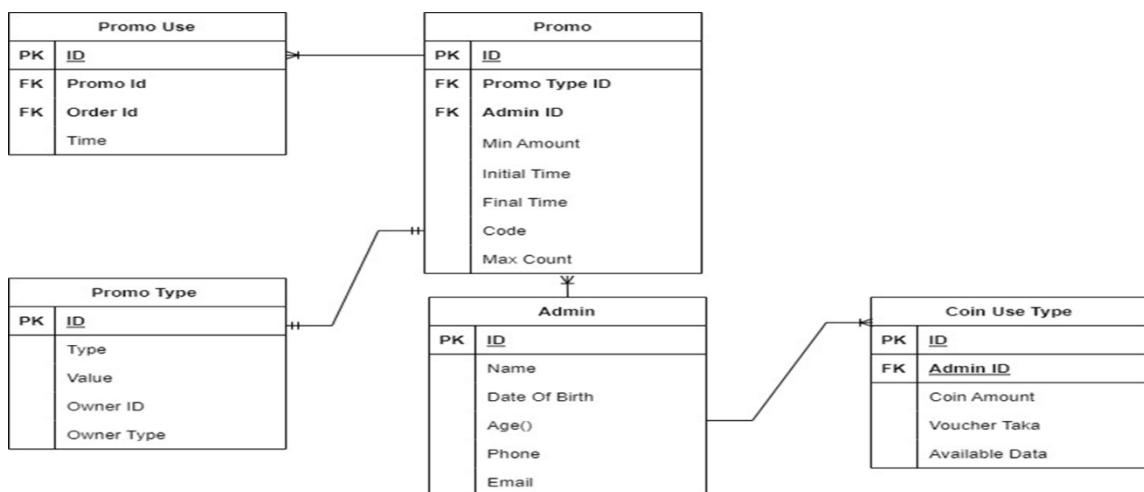
**Restaurant Module:** This module shows relation of Restaurant with category, vat, restaurant manager ,restaurant address, restaurant review and restaurant image.



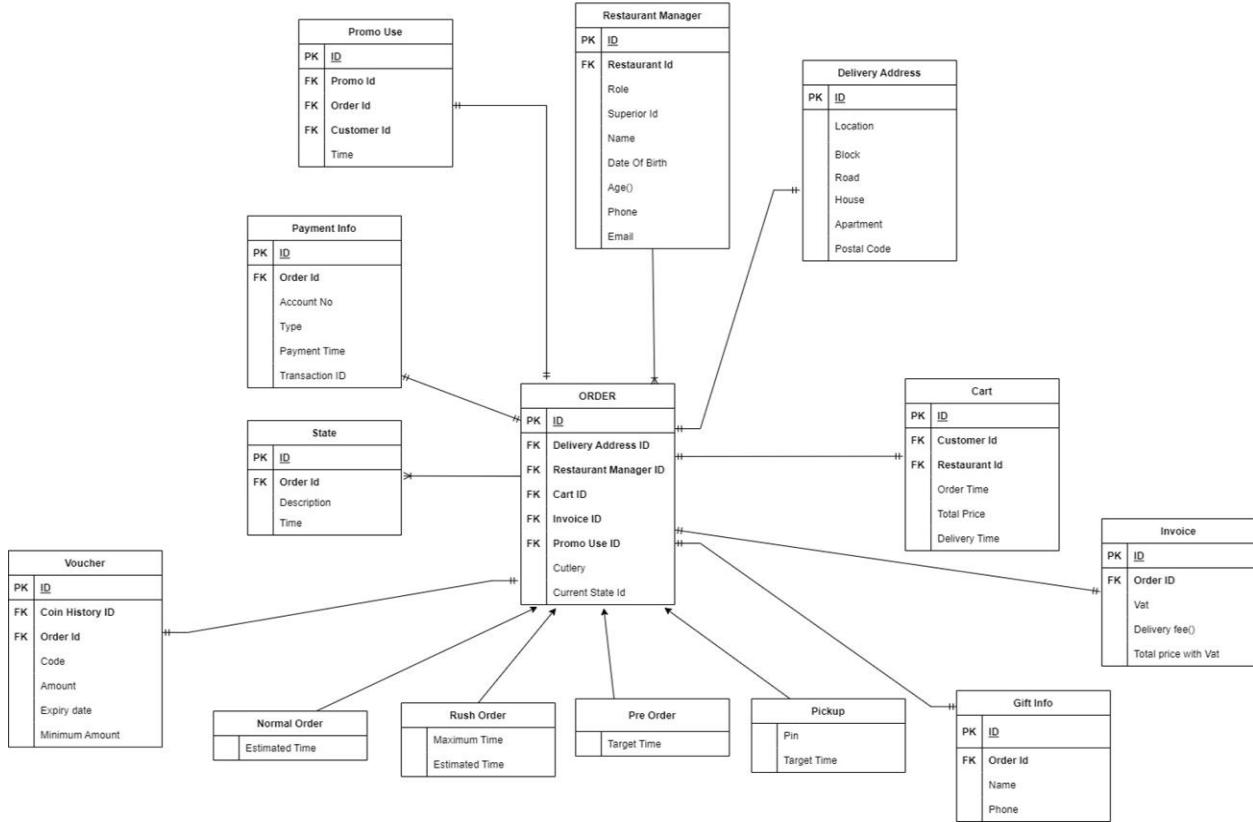
**Cart Module:** This module shows relation of cart with restaurant cart item and customer .



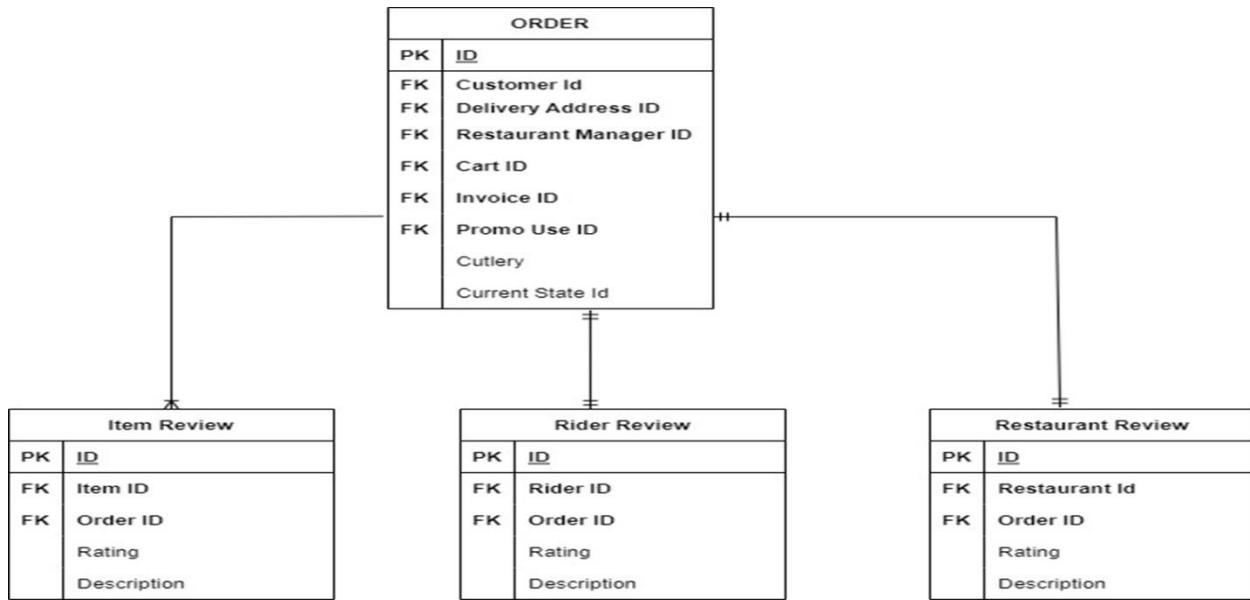
**Promo Module:** This module shows relation of promo with promo use ,promo type ,admin and coin use type.



**Order Module:** This is our biggest module . It shows the full order process. There are four types of order normal order,rush order,pre order and pickup.



**Review Module:** This module shows relation between order and all reviews.



### The feedback provided in the class for ER diagram :

1. We considered the vat restaurant specific .But sir advised us to make two entity one service charge and another is vat .Service charge is restaurant specific and vat is same for all the restaurant.
2. We added total price in cart item.But,sir advised us to remove that add item count there and make total price in cart only.

## Sequence Diagram

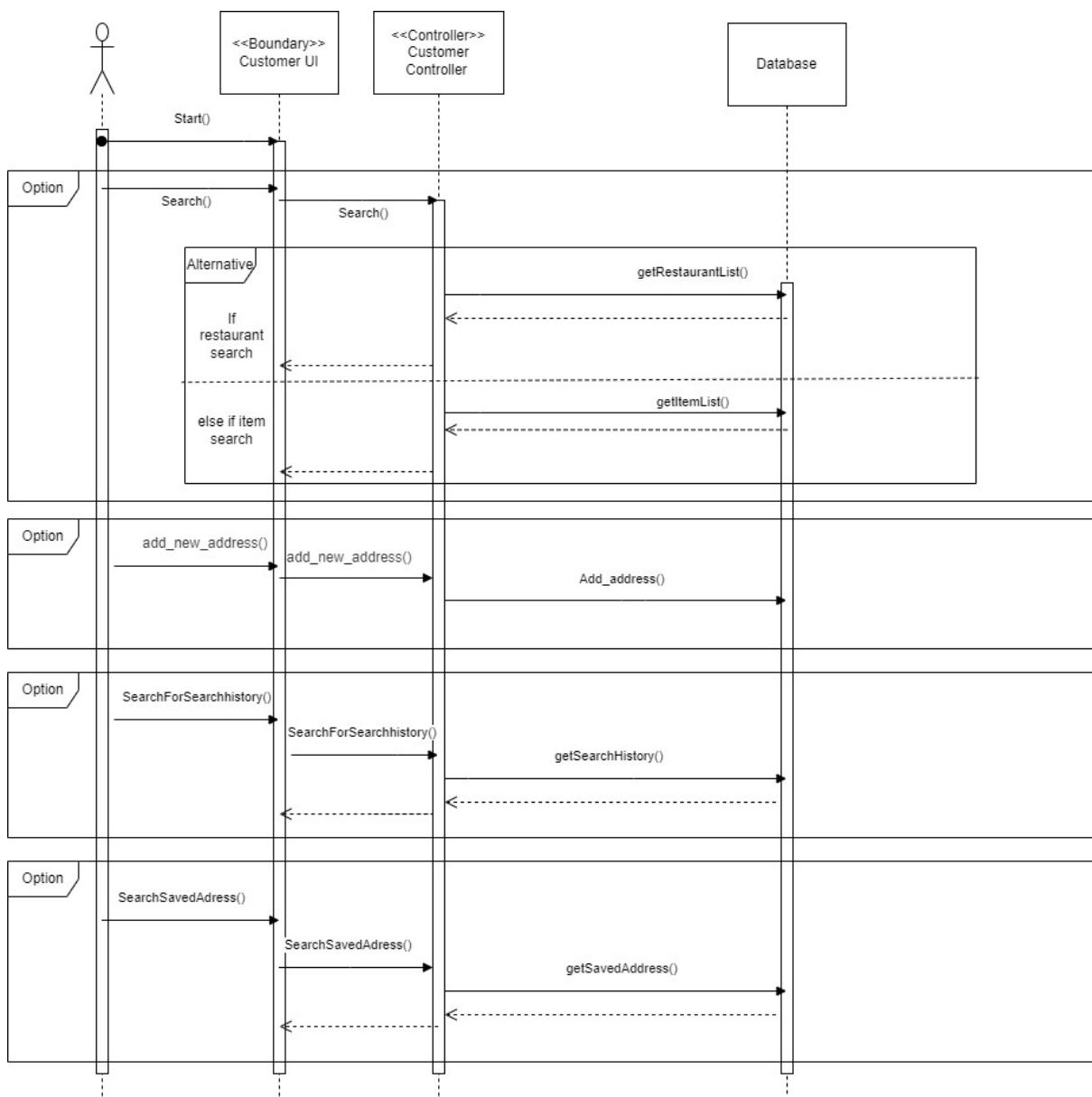
There are total 12 diagram in our sequence diagram-

- Customer search and address
- Create new category
- Create new item
- Adding item to cart
- Editing cart item to cart
- Customer order
- Restaurant order
- Rider order
- Admin order
- Customer feedback
- Promo

Some module were parted into multiple sequence diagram for the convenience of drawing the diagrams.

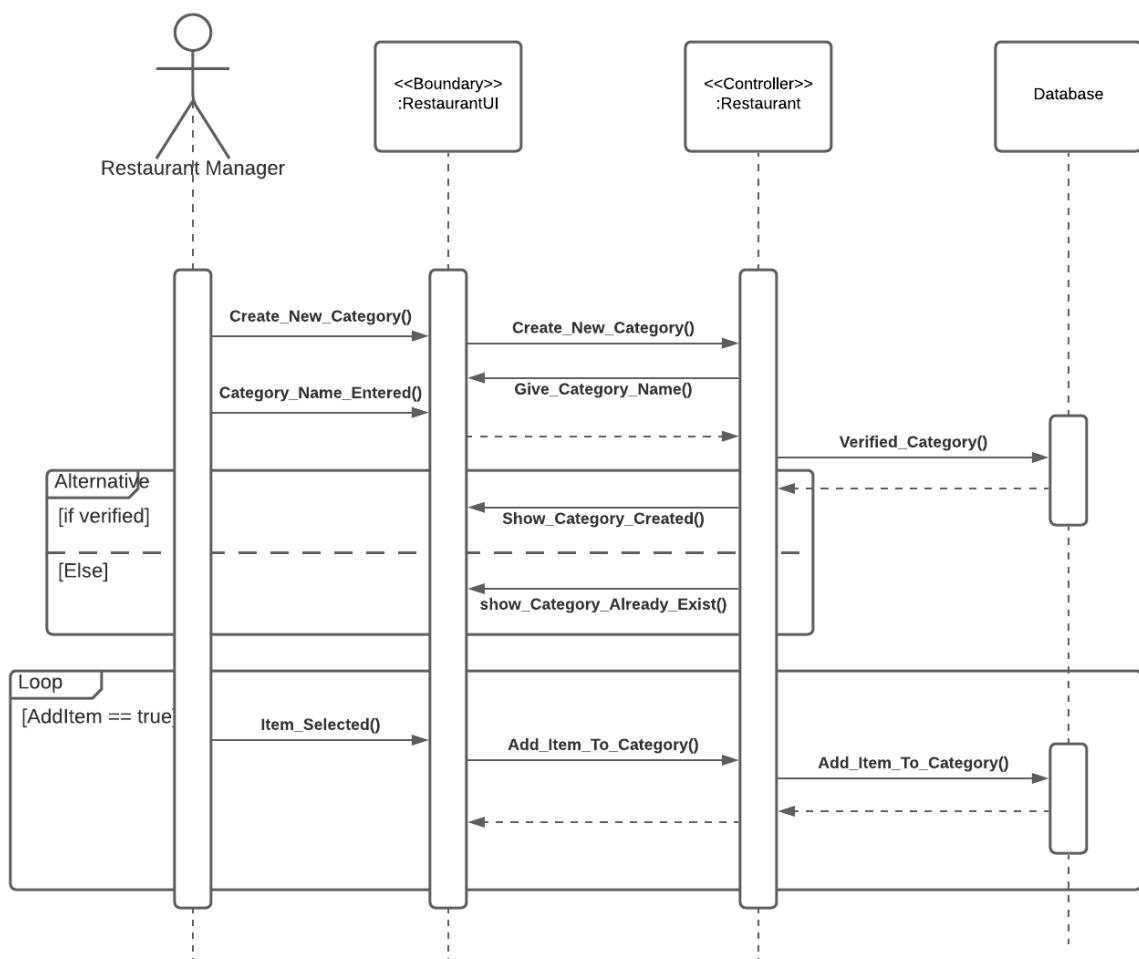
## Customer Search and Address

This diagram shows sequence diagram needed for customer operations and search operations.



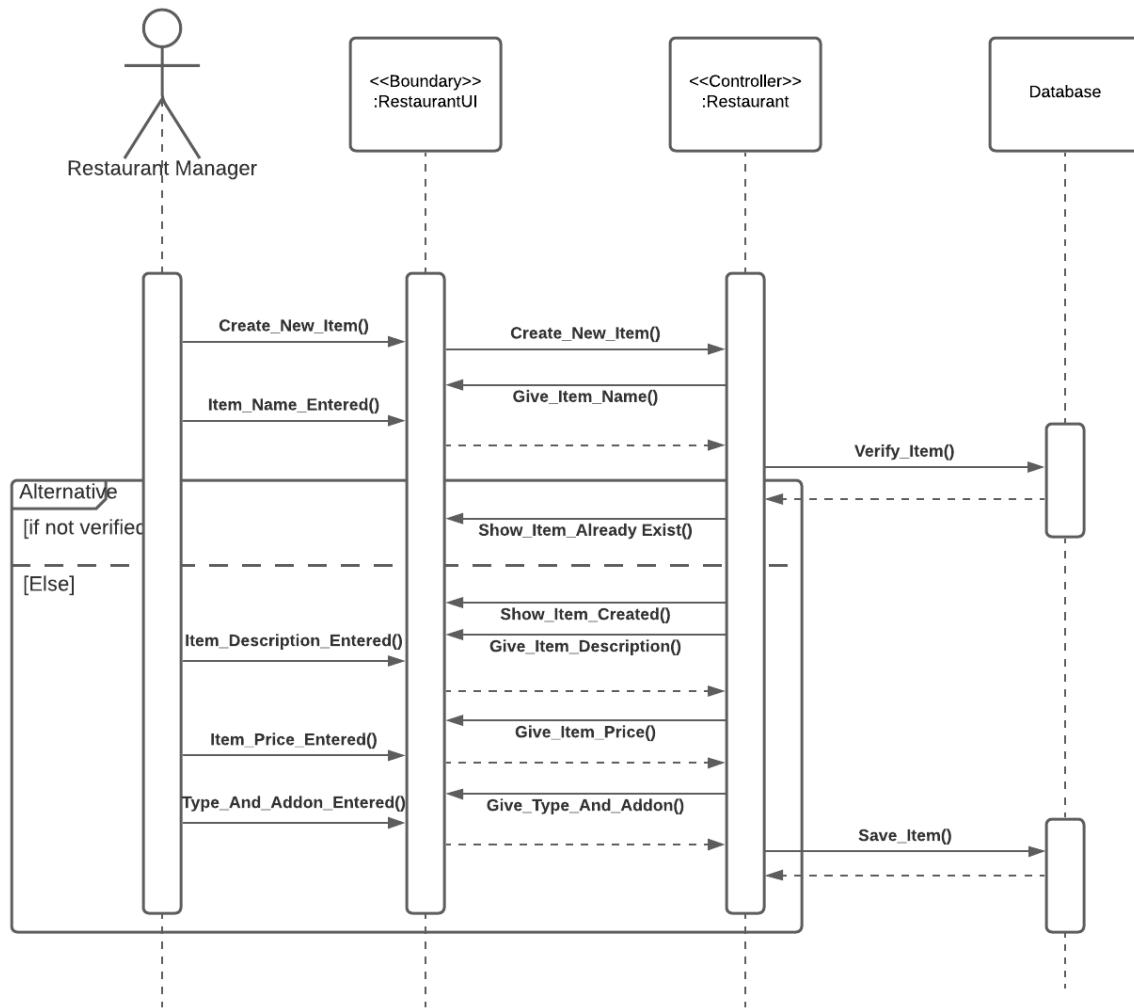
## Create New Category

This diagram shows sequence diagram related to creating new category in Customer end.



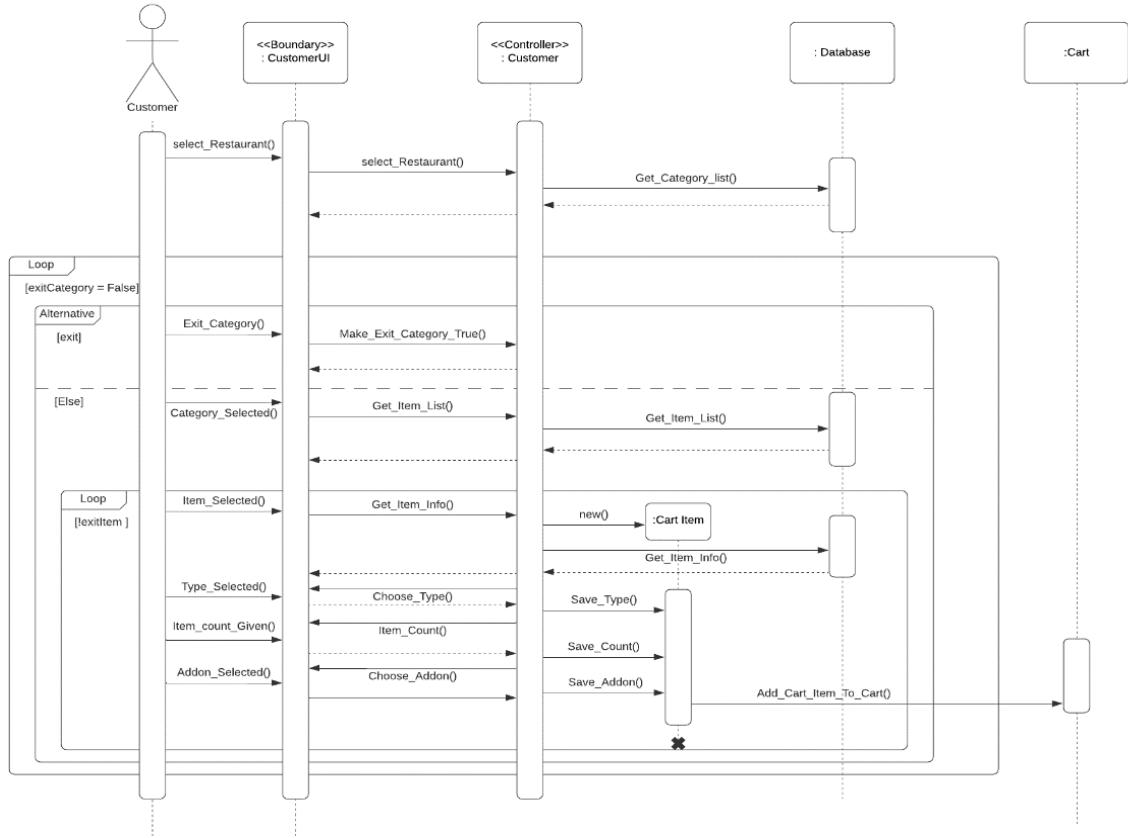
## Create New Item

This diagram shows the sequence diagram need to create a new item in restaurant end.



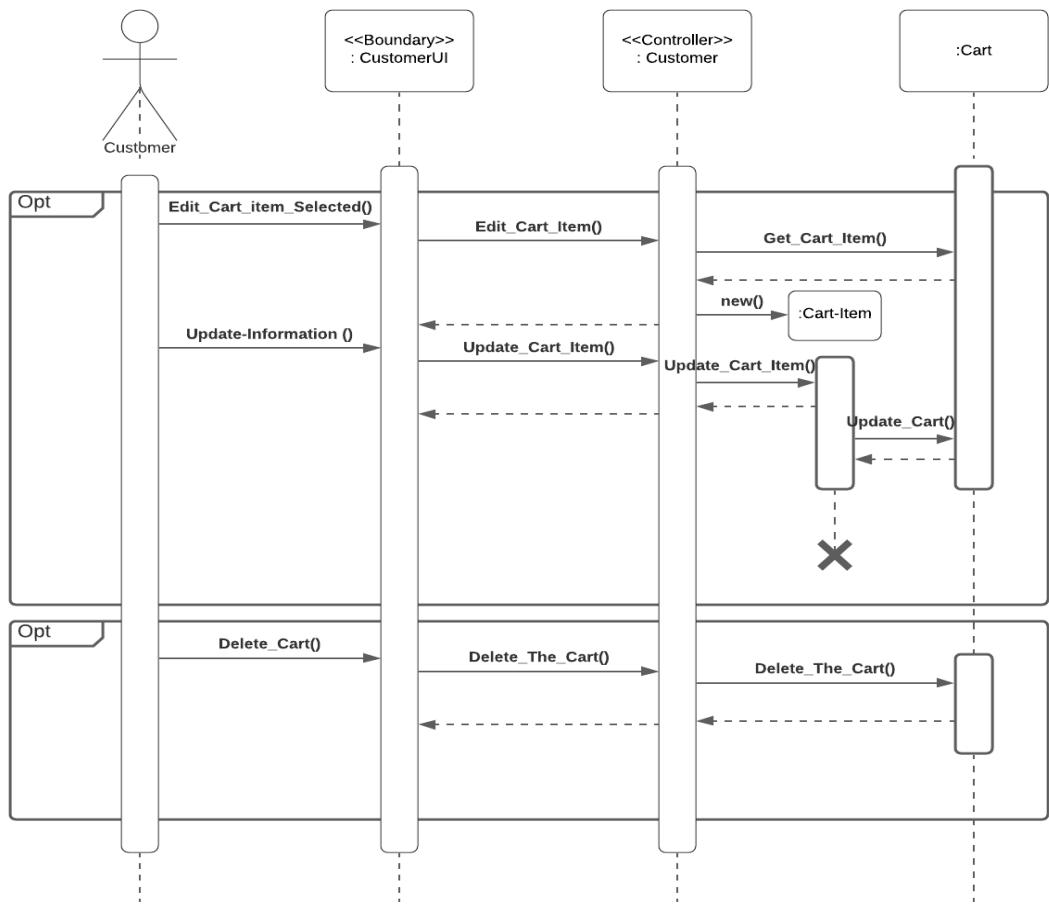
## Adding item to Cart

This diagram shows sequence diagram related to building up order Cart in customer end.



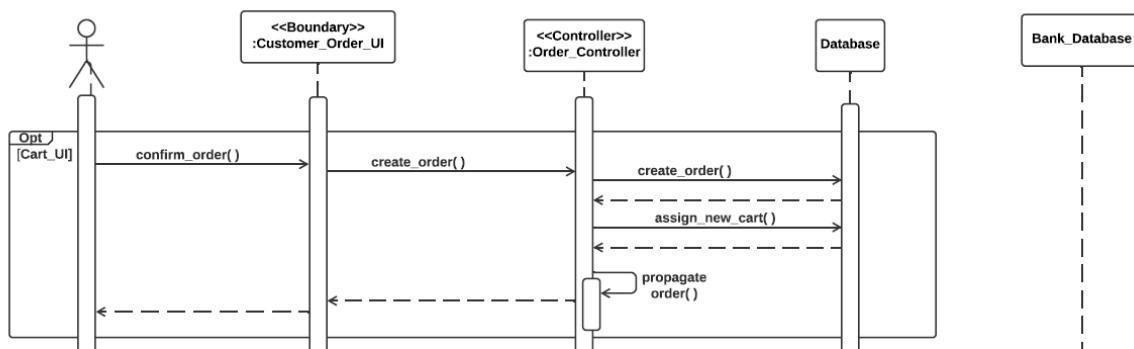
## Editing Cart-item to Cart

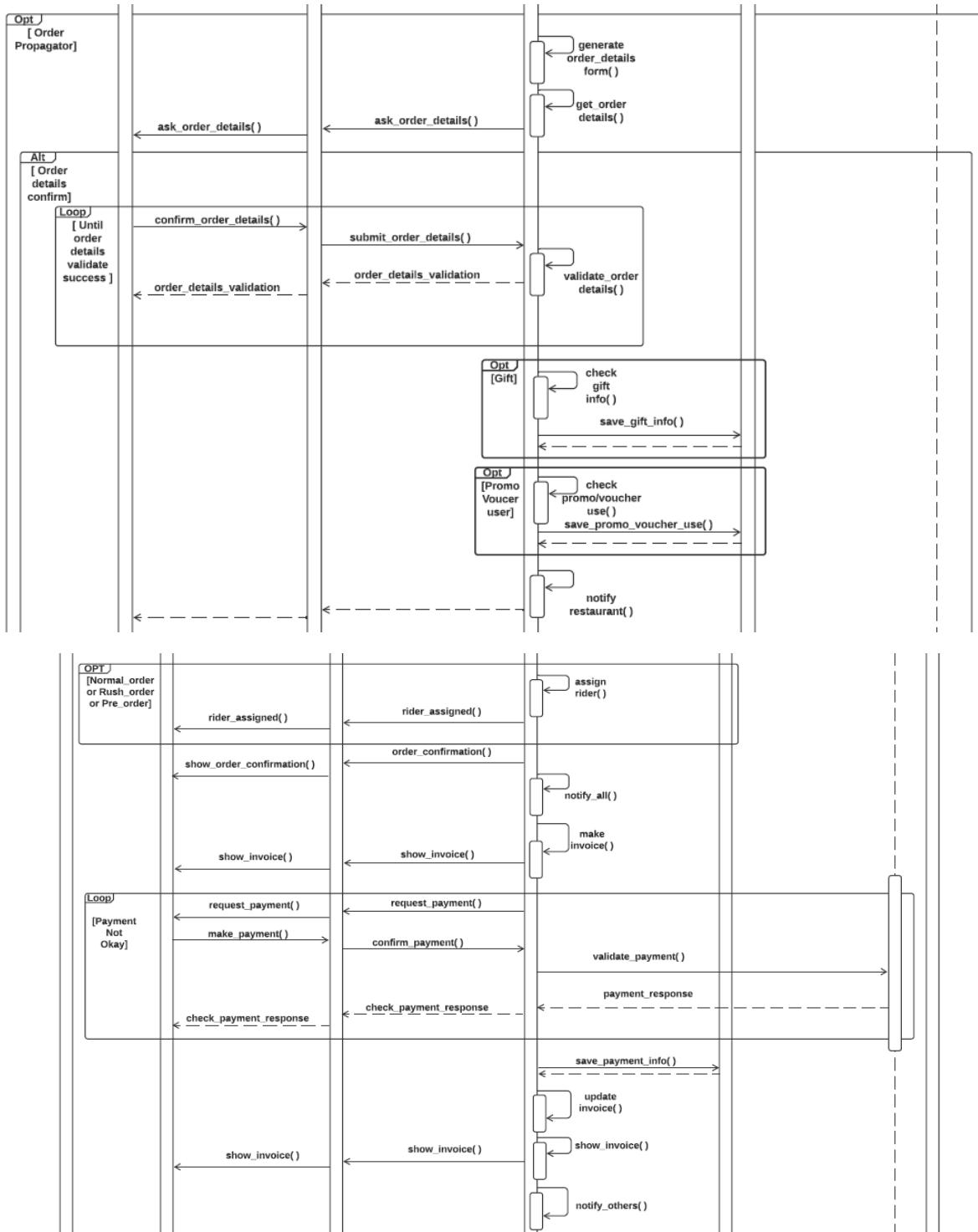
This diagram shows sequence diagram related to editing cart for making an order.

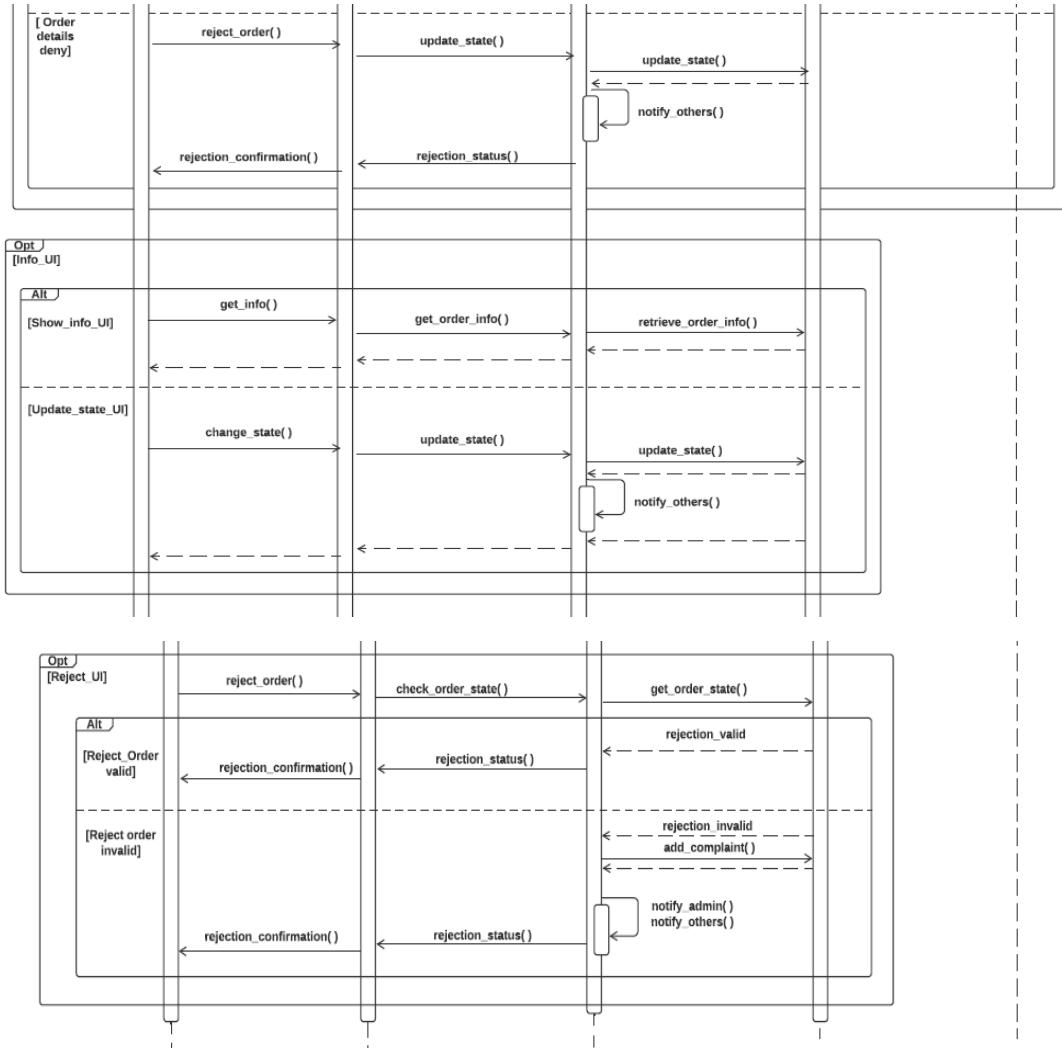


## Customer Order

This diagram shows sequence diagram of making an order from customer end.

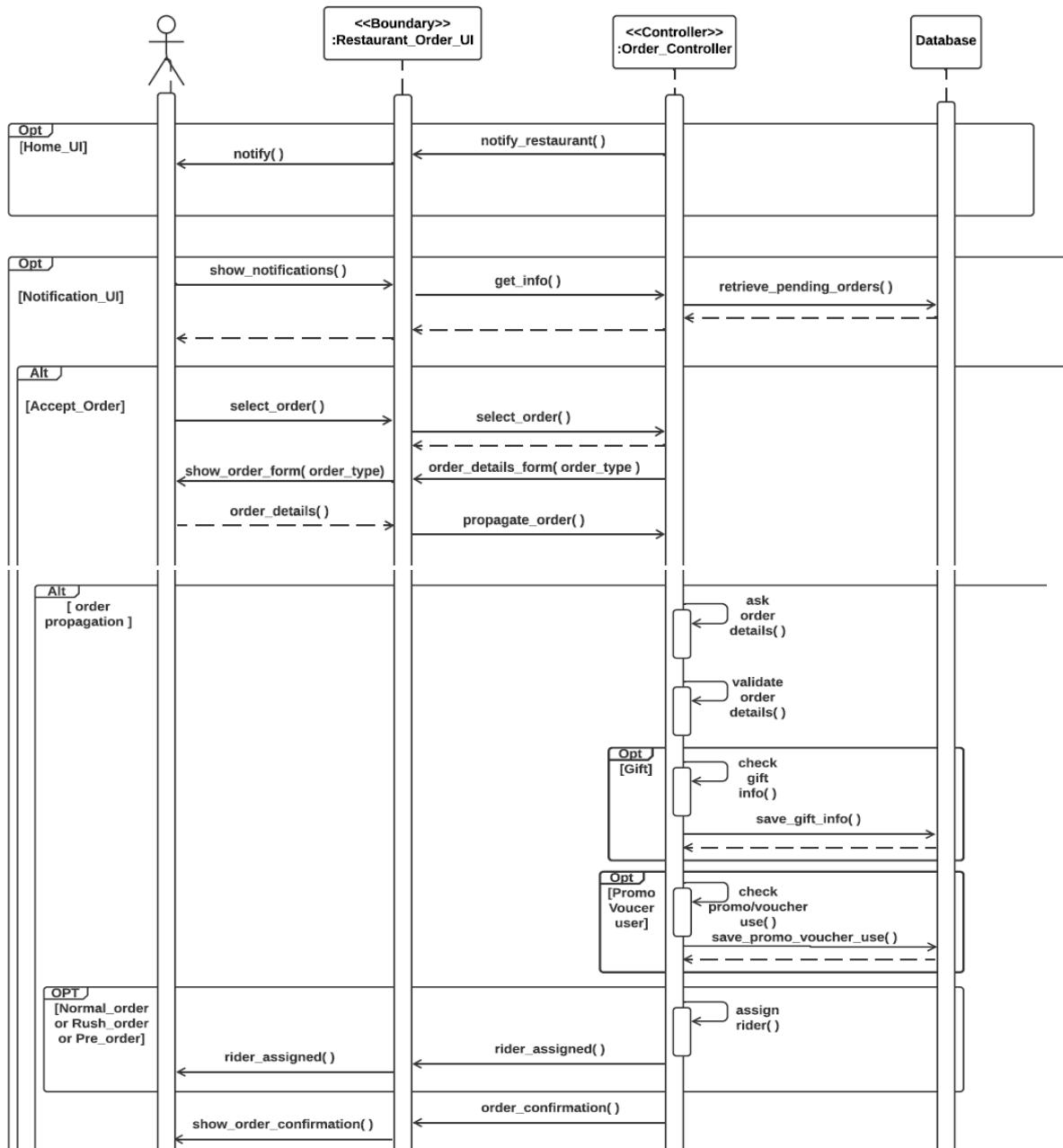


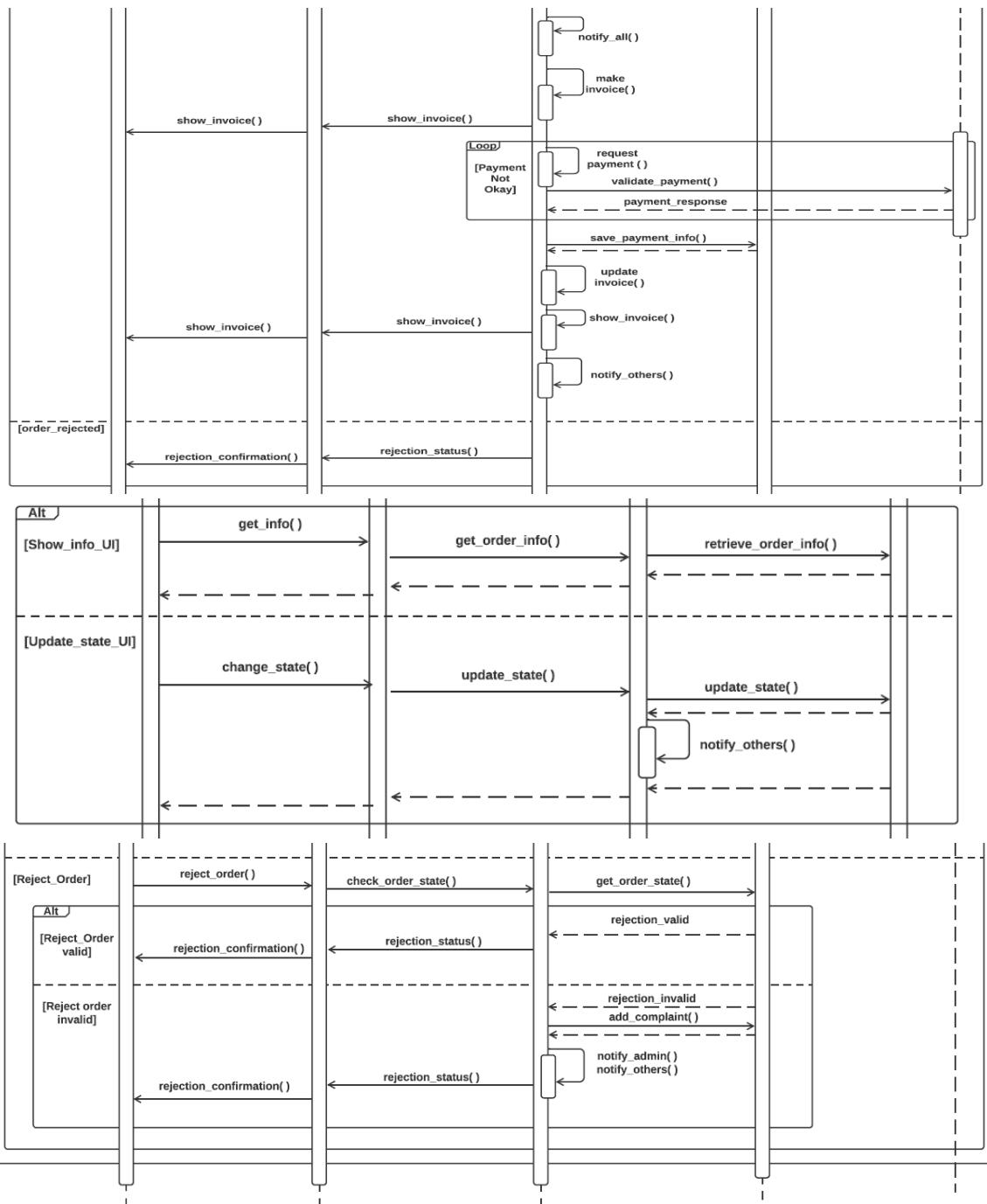




## Restaurant Order

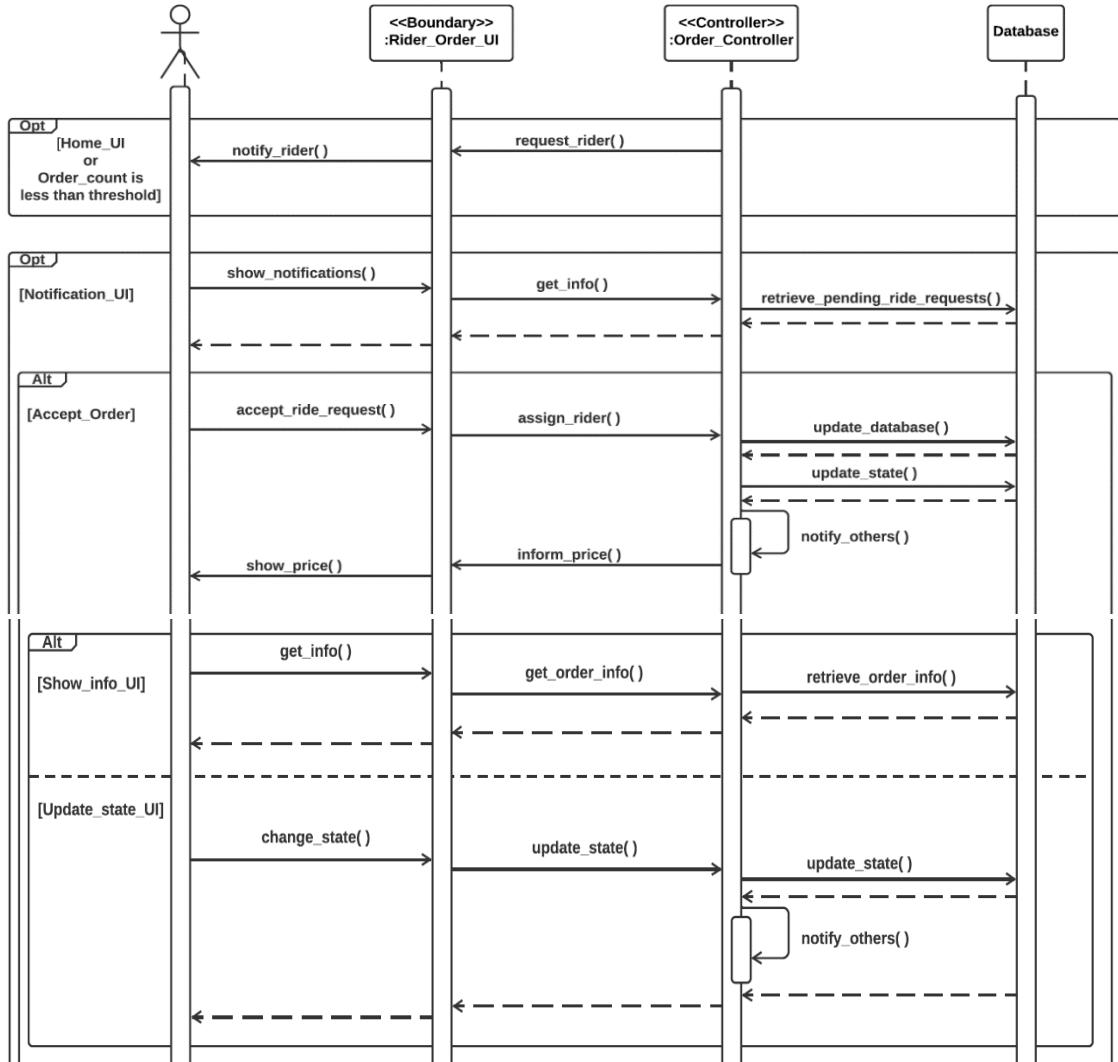
This diagram shows sequence diagram related to operations of an order in restaurant end. This is part of Order module.

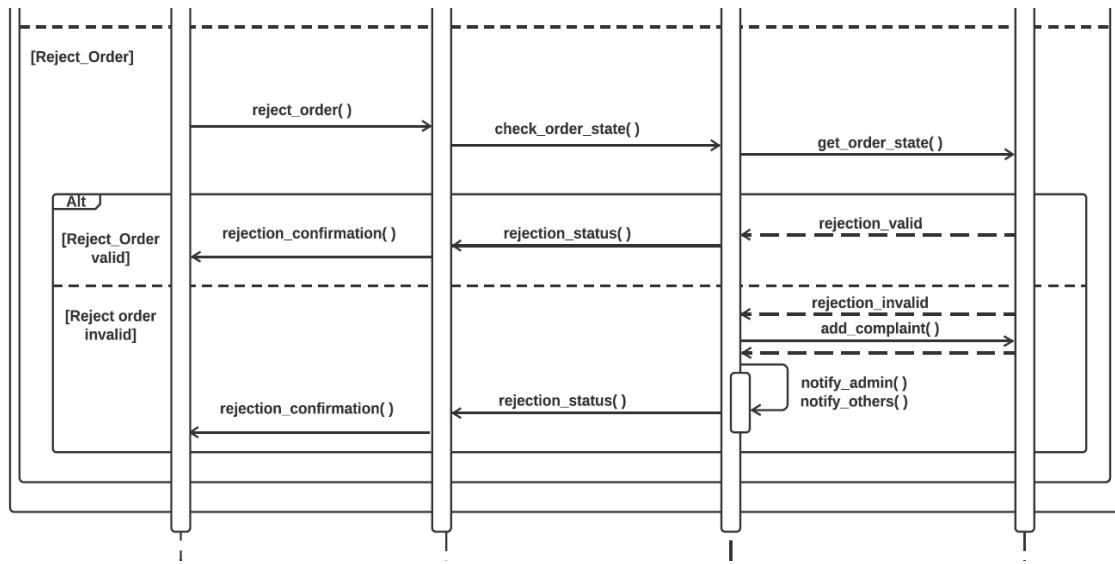




# Rider Order

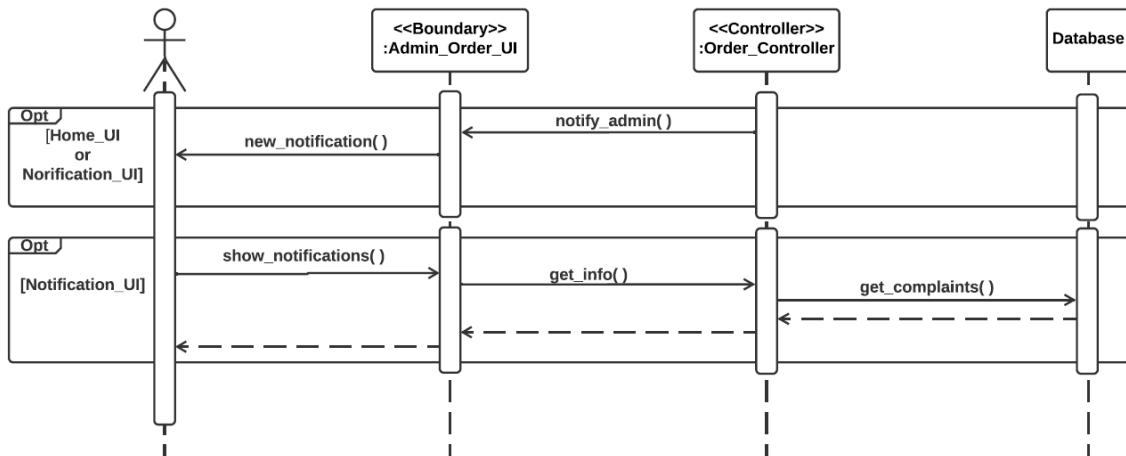
This diagram shows sequence diagram related to operations of an order in rider end. This is part of Order module.





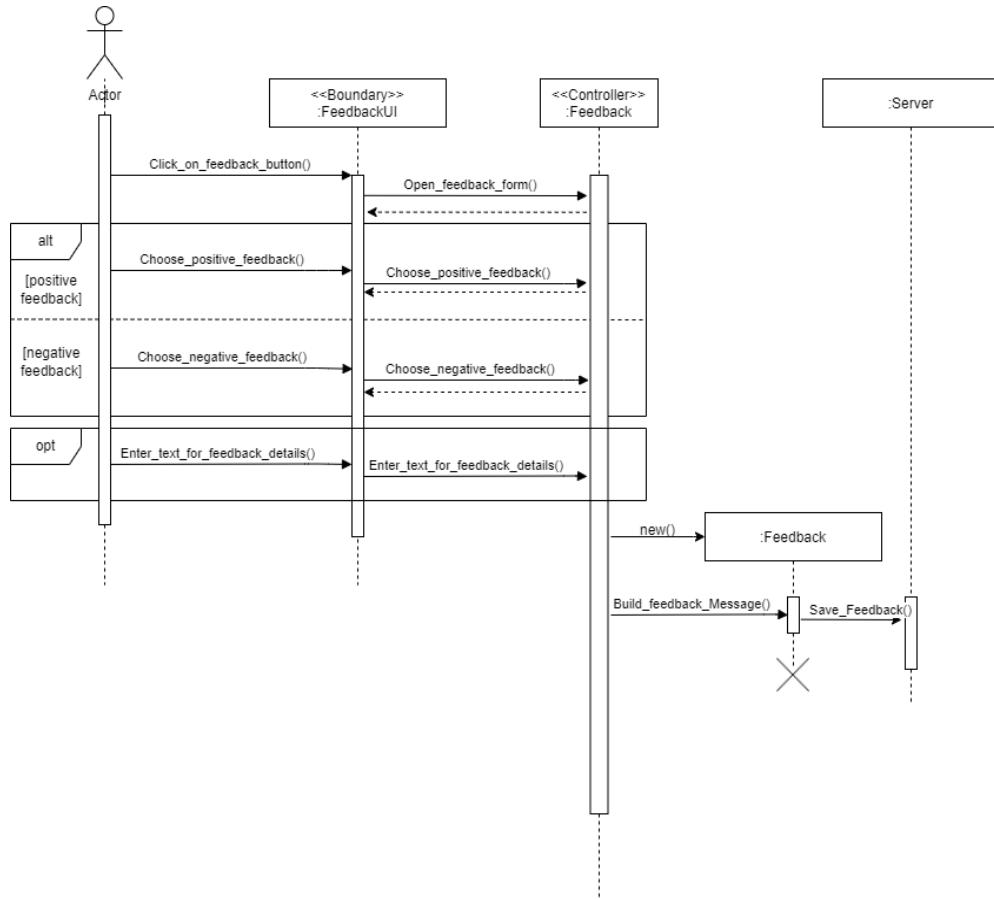
## Admin Order

This diagram shows sequence diagram related to operations of an order in admin end. This is part of Order module.



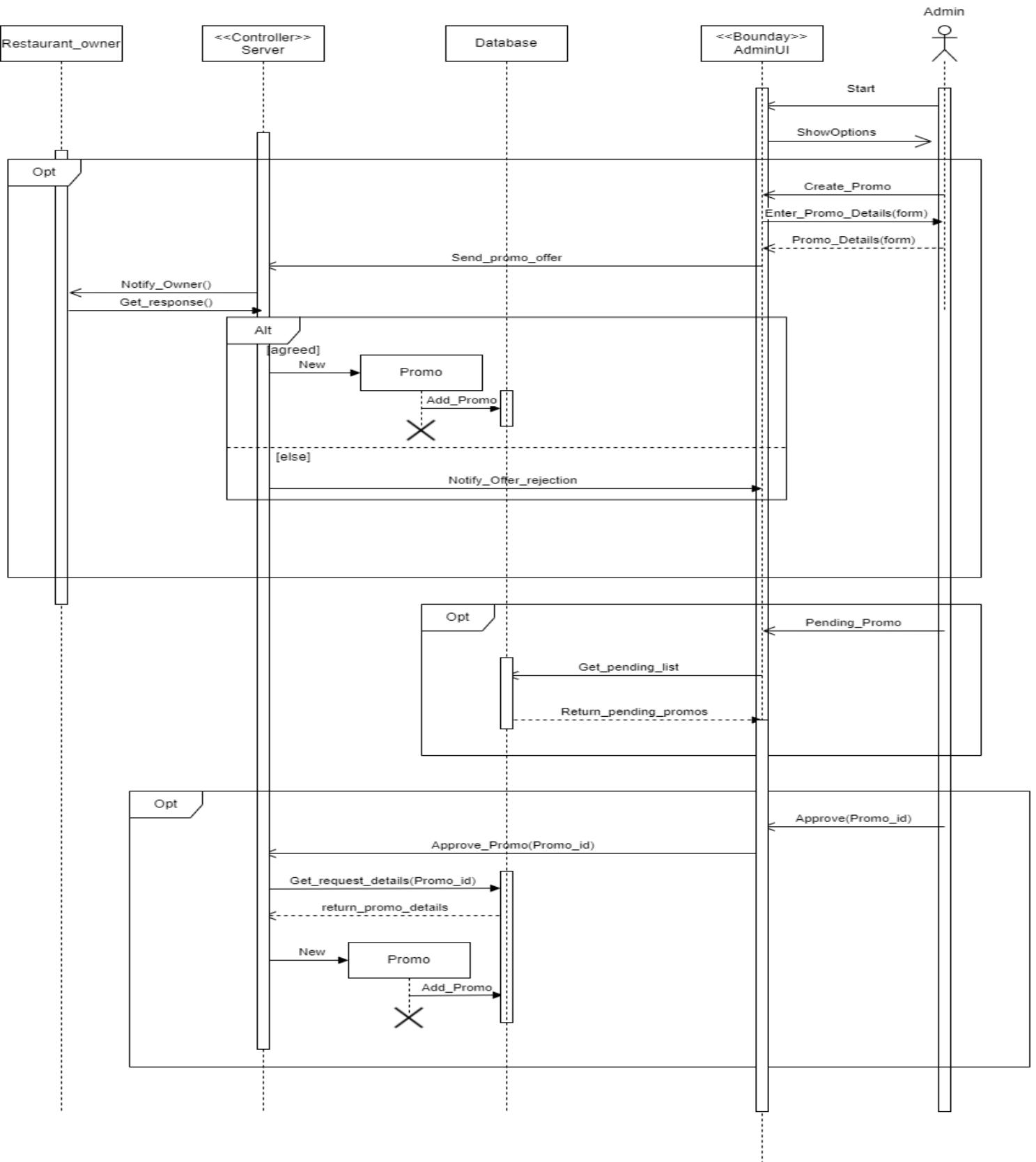
## Customer Feedback

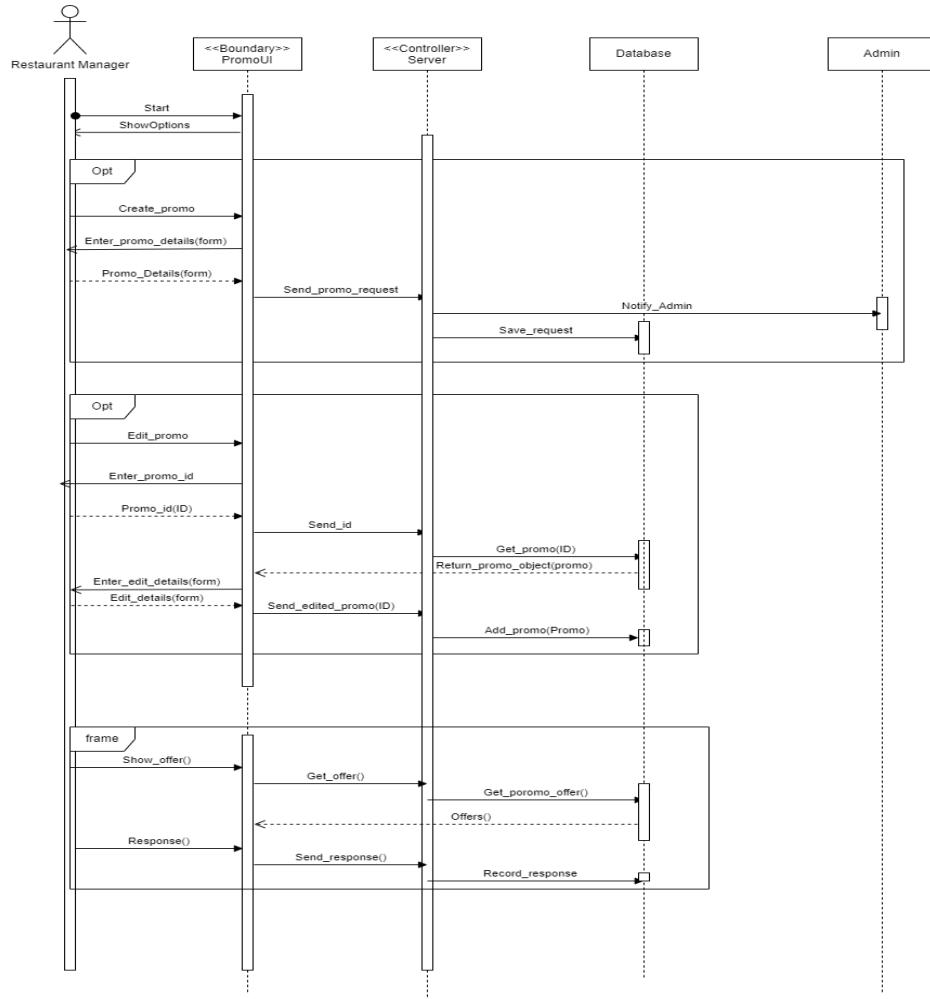
This diagram shows sequence diagram related to making review on an order in customer end.



## Promo

This diagram shows sequence diagram related to promos.





## The feedback provided in the class:

1. We were advised to prefer assignment of rider from restaurant end. This was later edited while implementing a significant module of our project.

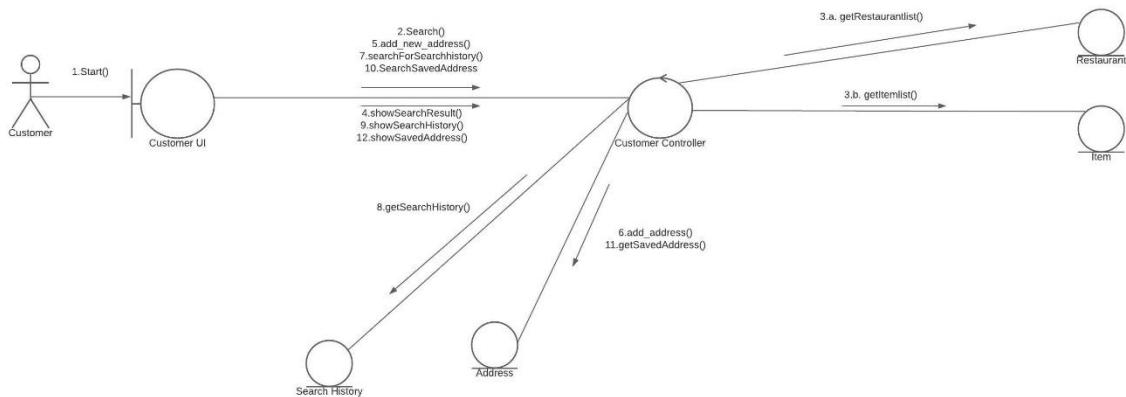
## Collaboration Diagram:

There are total 7 diagram in our project-

- Customer search and address
- Create new category
- Adding item to cart
- Ordering process
- Customer feedback
- Hunger express giving promo
- Restaurant giving promo

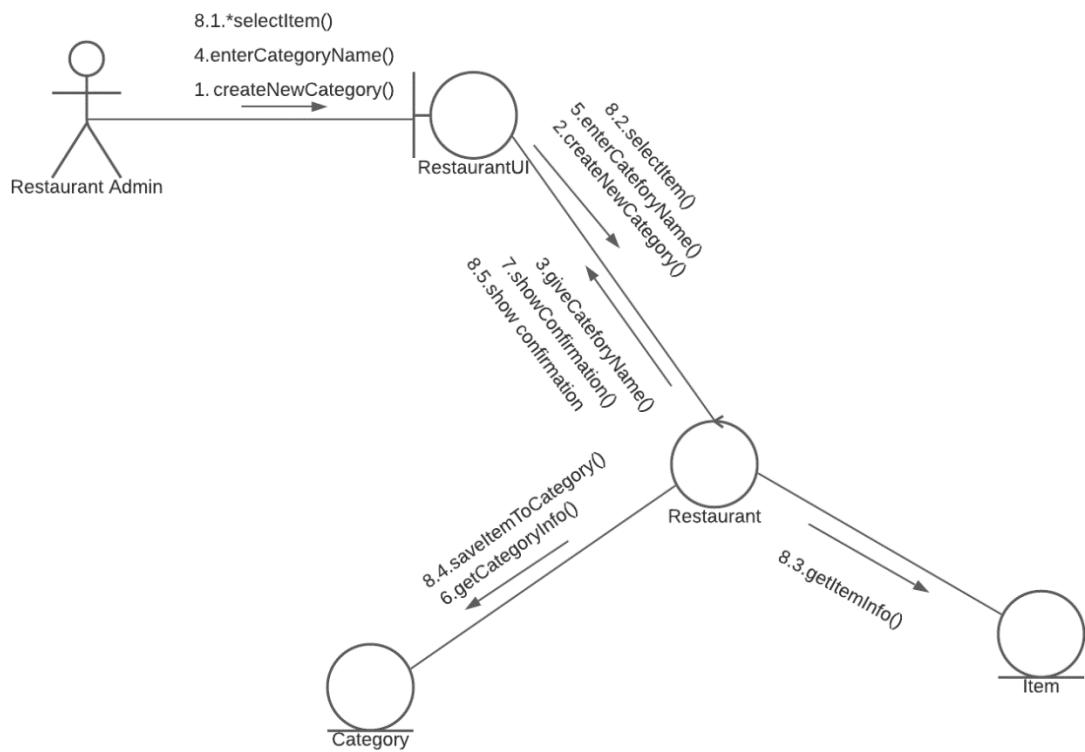
### Customer Search and Address

This diagram shows how a customer can search items and save his preferred address of delivery.



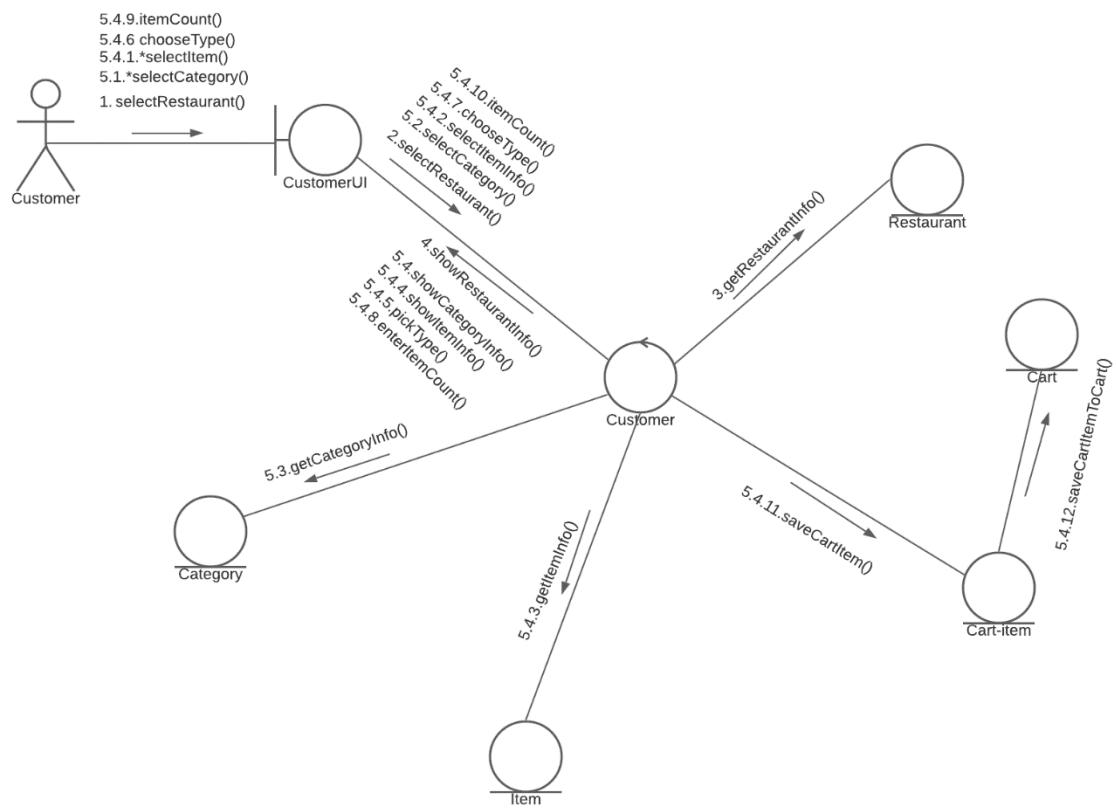
## Create New Category

This diagram shows how a restaurant owner can create a new category with necessary information.



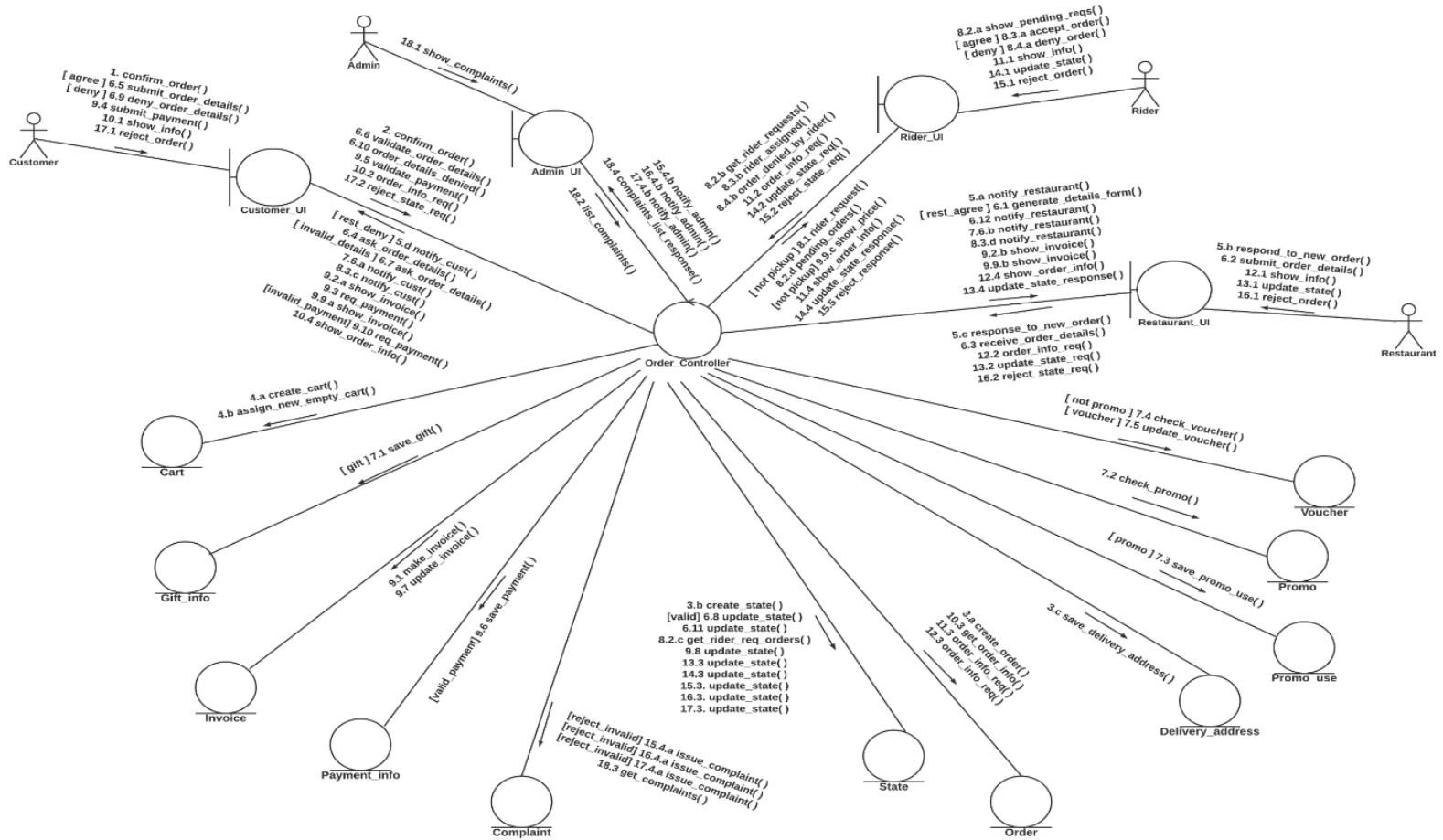
## Adding Item to Cart

This diagram shows how a customer can add items to his cart.



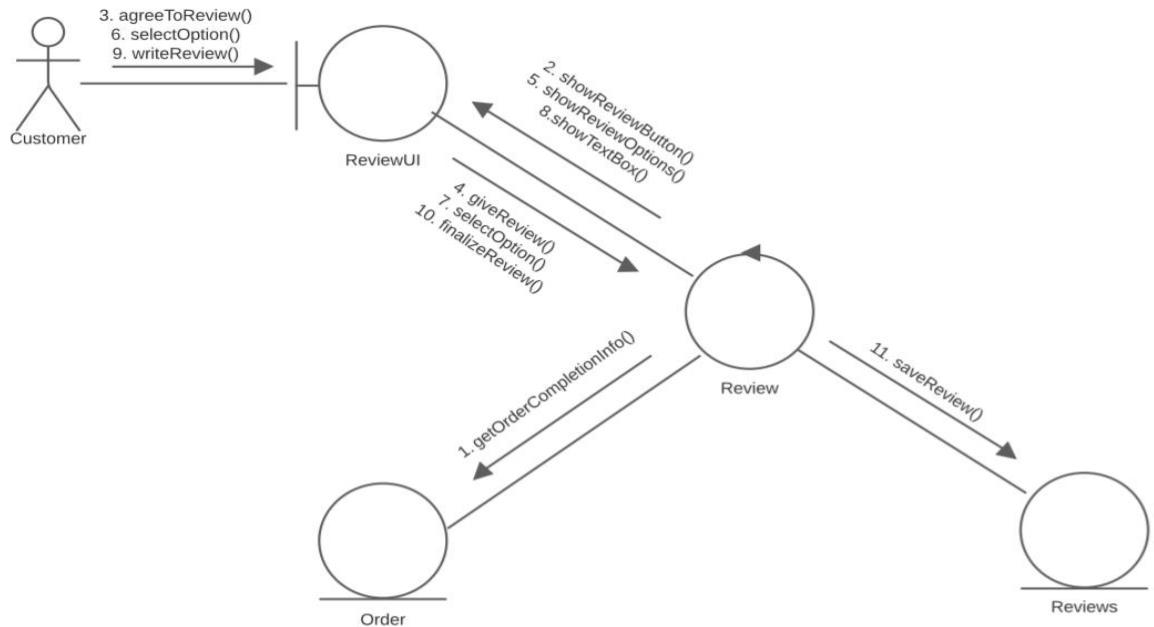
## Ordering Process

This diagram is the biggest diagram in our collaboration diagram and shows how a customer can order with necessary steps.



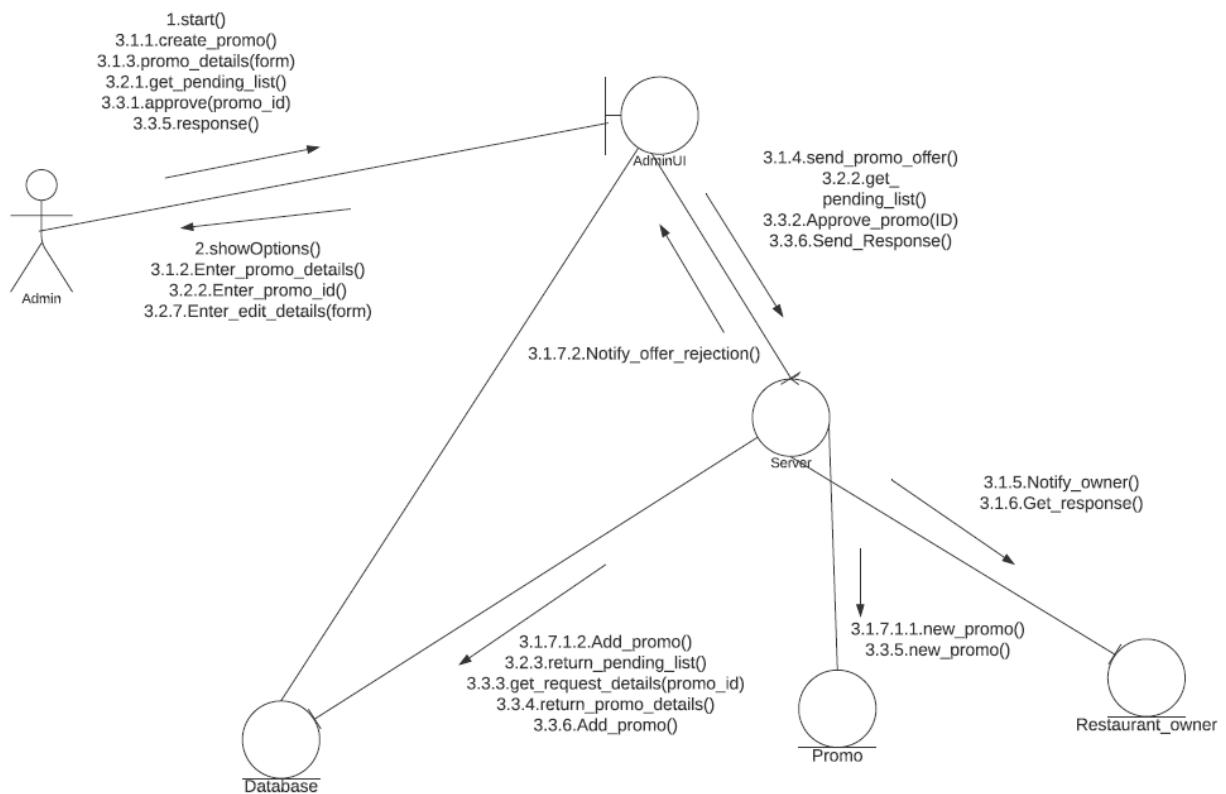
## Customer Feedback

This diagram shows how a customer can give feedback for betterment.



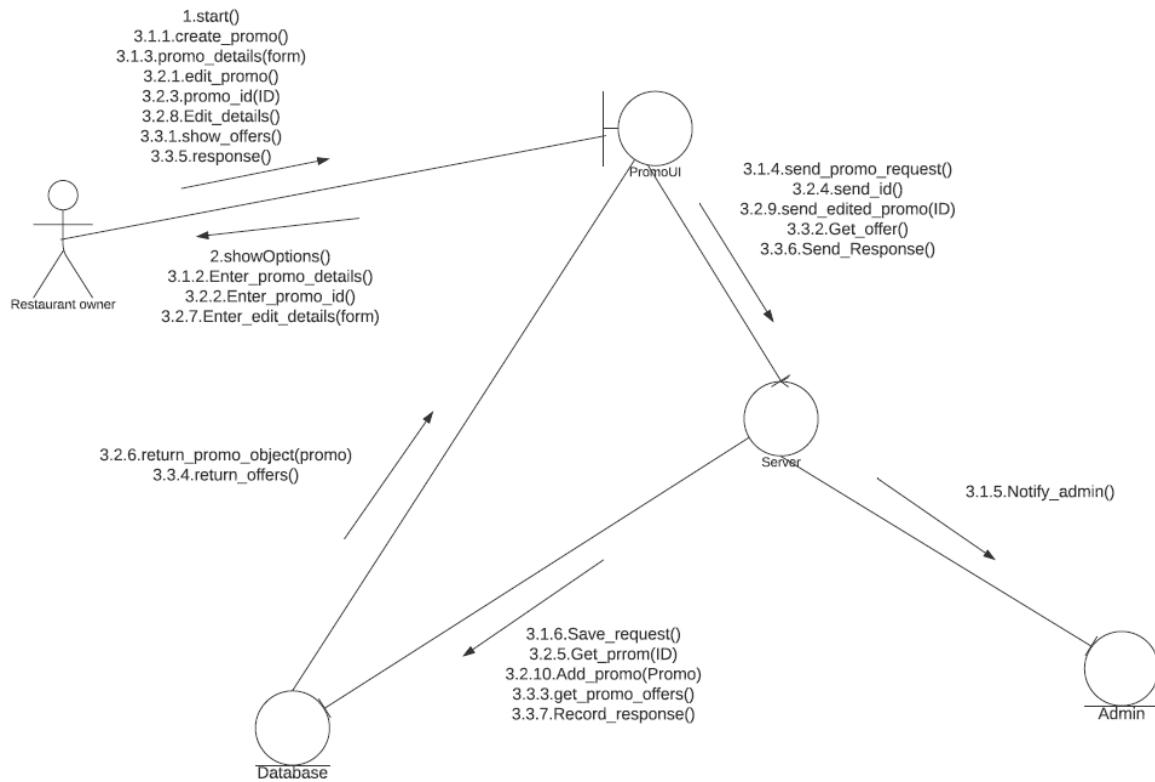
## Hunger Express giving Promo

This diagram shows how admin can give promo to customers.



## Restaurant giving Promo

This diagram shows how a restaurant owner can give promo to his customers.



## The feedback provided in the class for collaboration diagram:

1. We considered the delivery would be picked up by rider himself. But sir advised us to make this work should be moderated by admin or intelligent algorithm.
2. Order process module need to be slightly changed showing some other collaboration between others.
3. Also using loop in collaboration diagram is not a good idea.

## Mock UI:

A mockup is a static wireframe that includes more stylistic and visual UI details to present a realistic model of what the final page or application will look like.

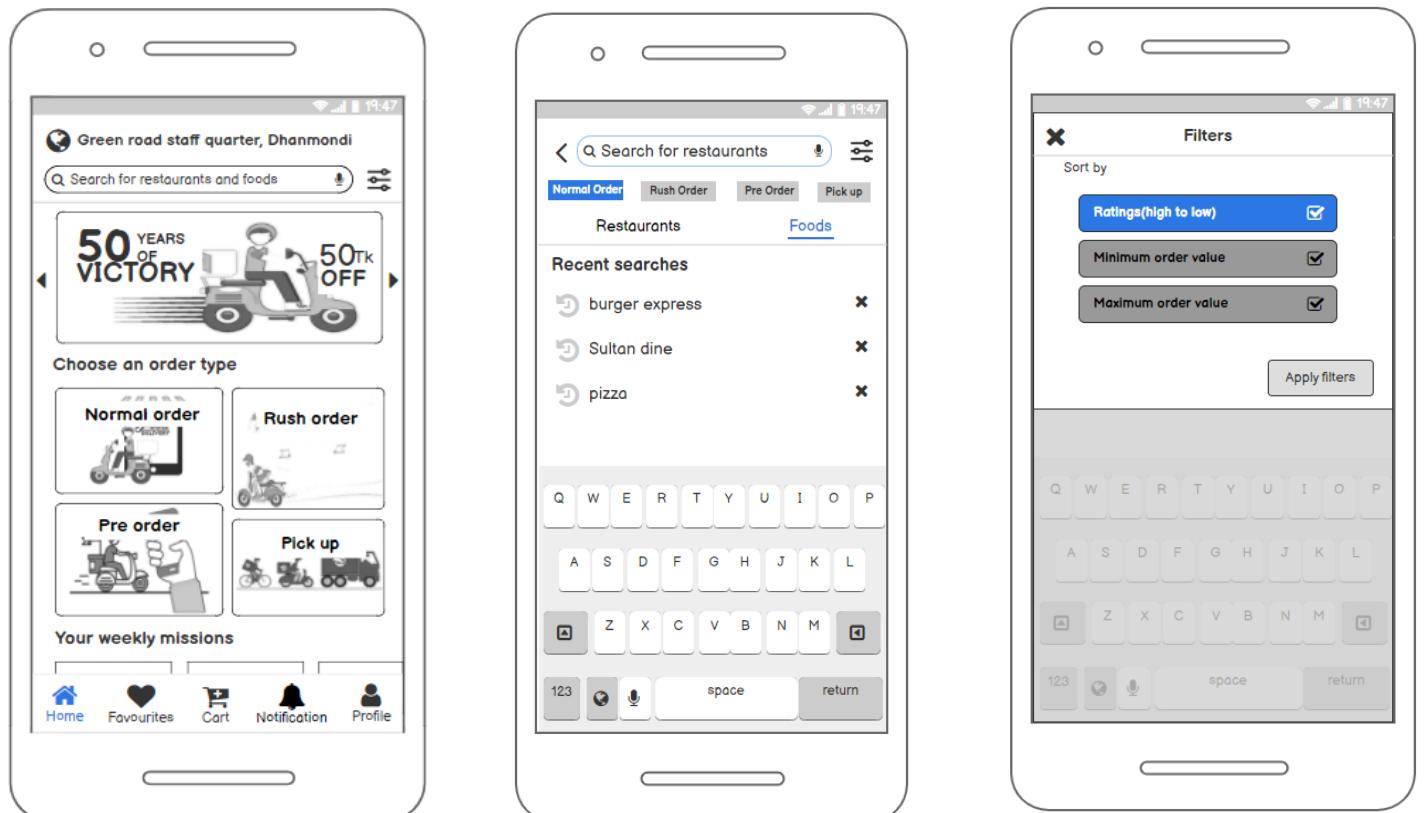
There are total 3 modules in our project

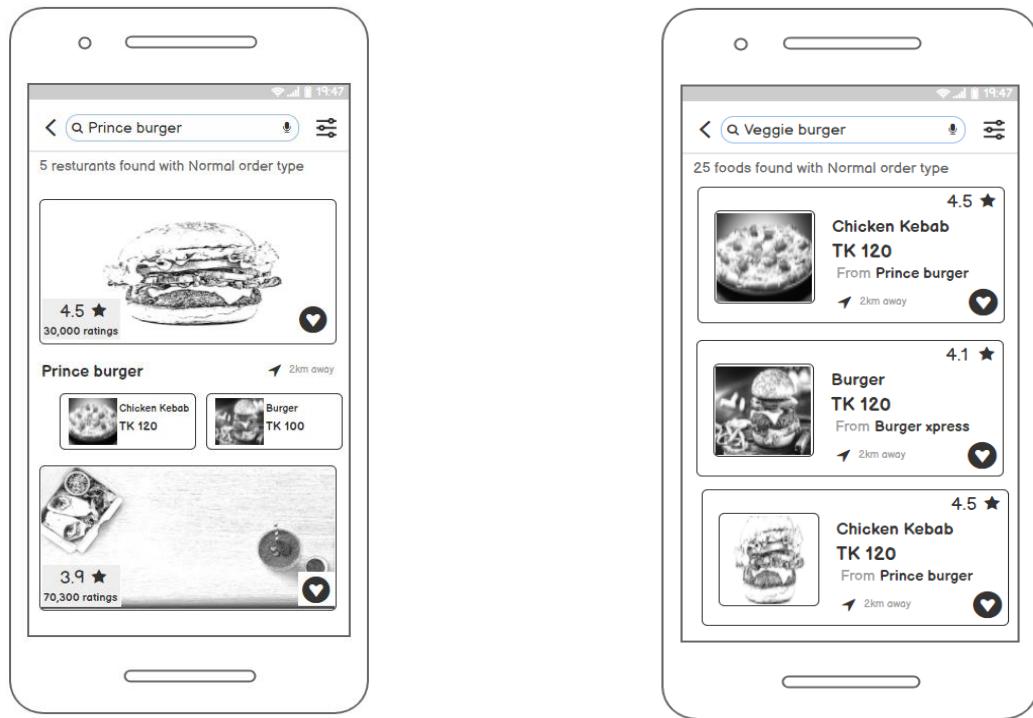
- i) Customer Module
- ii) Restaurant Module
- iii) Rider Module

We designed all the pages of every module.

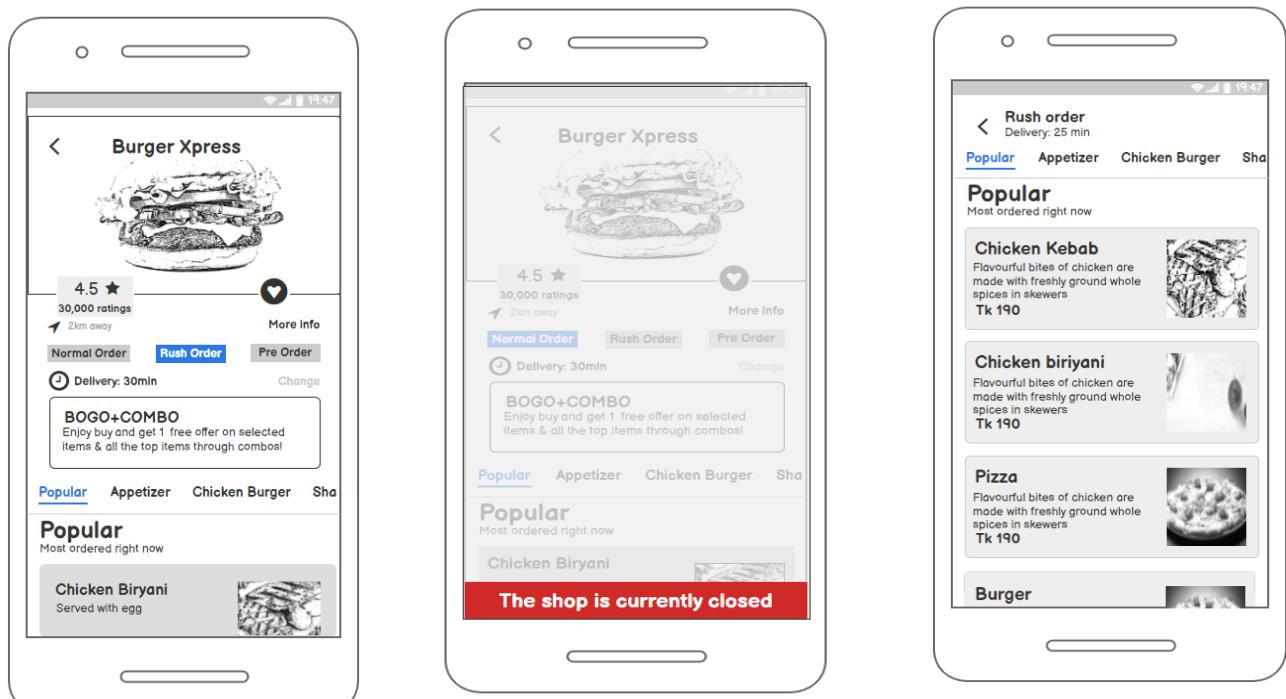
### 1. Customer Module:

So, in this customer page, a customer can order for food according to the order type like normal order, pre order , rush order and pick up order. Also they can search for food. We can also use the filter while searching for item or restaurant.

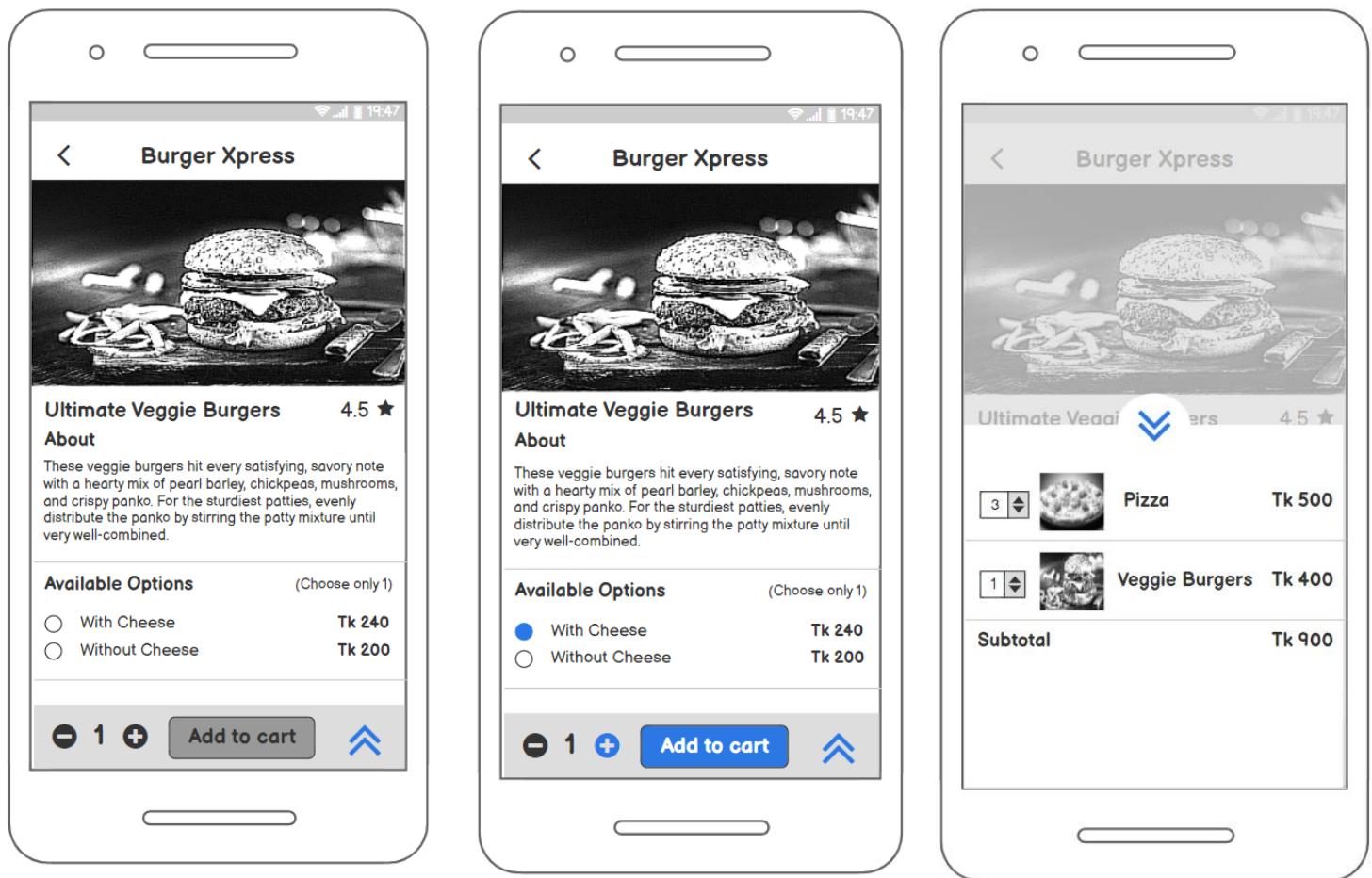




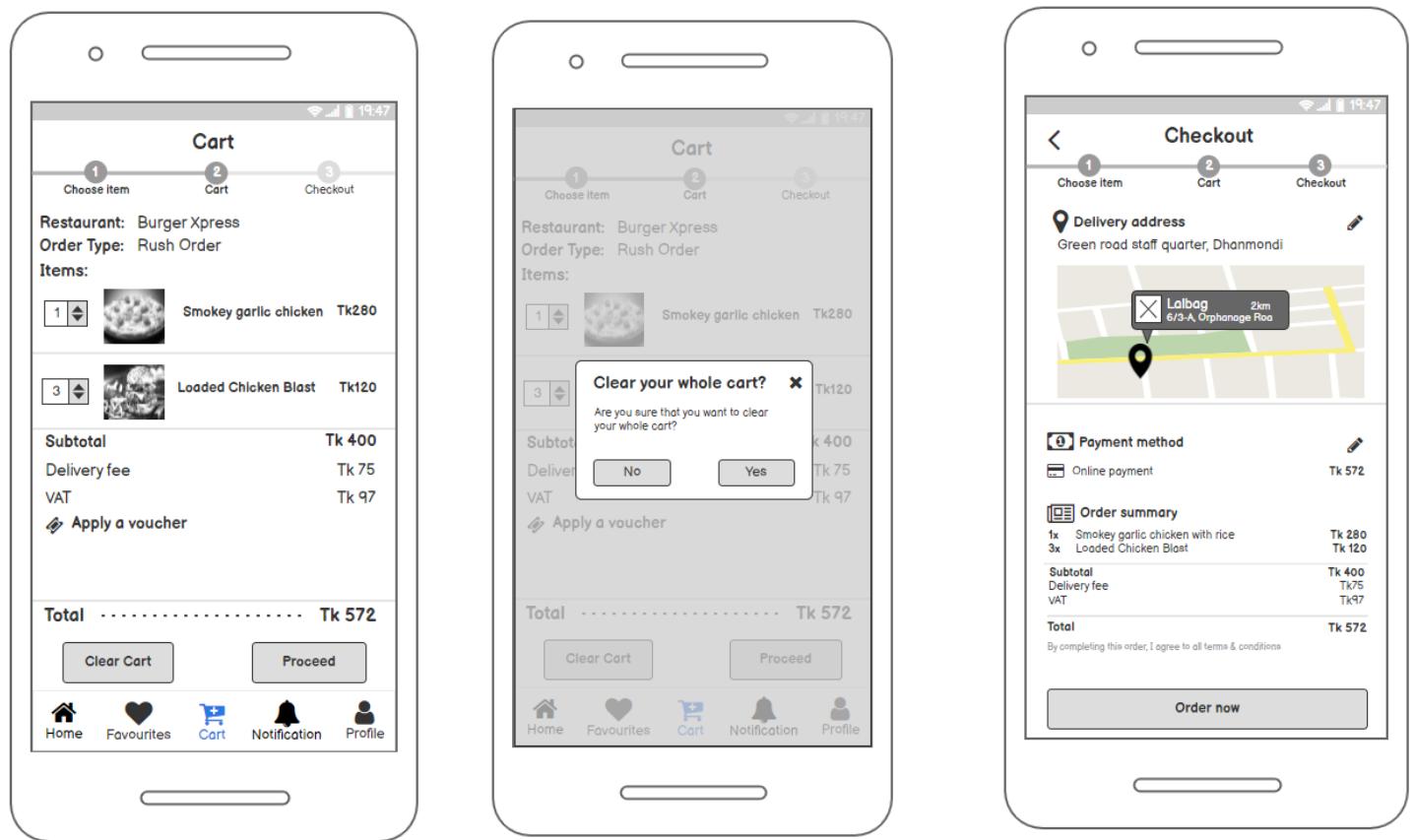
This is the inside view of the restaurant page form the customer perspective. They can choose category from the list and pick a food. Also the customer can change the order type.



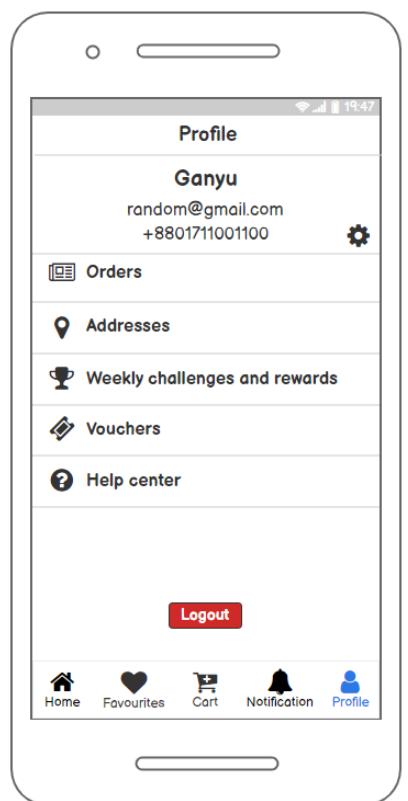
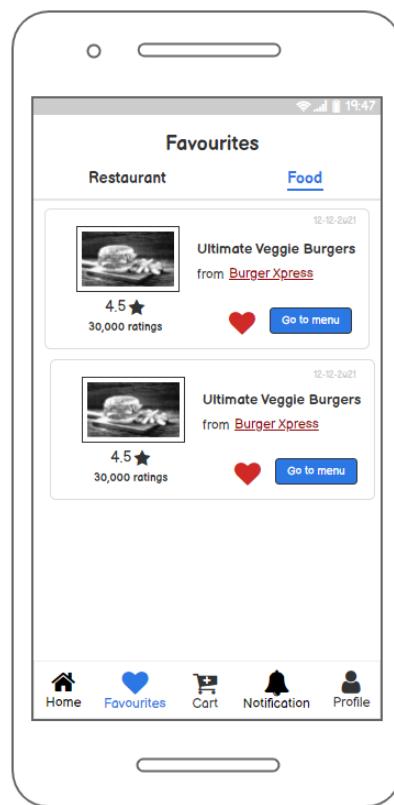
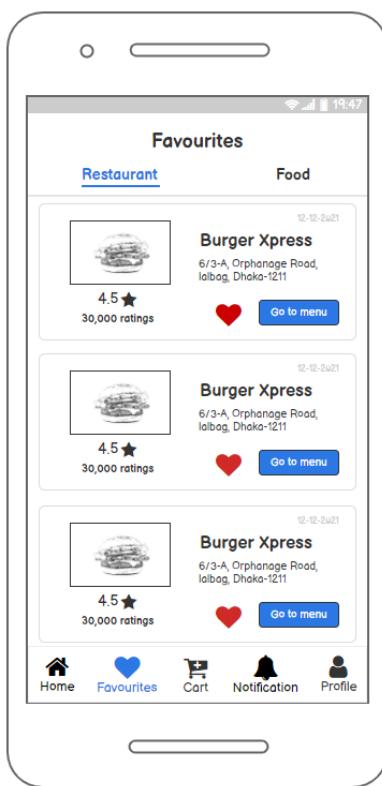
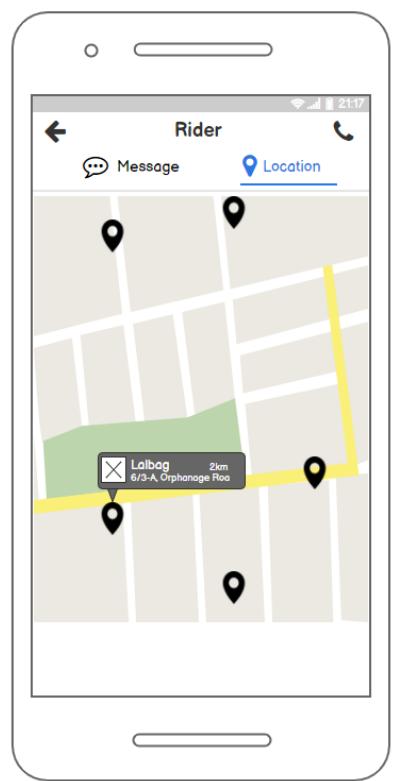
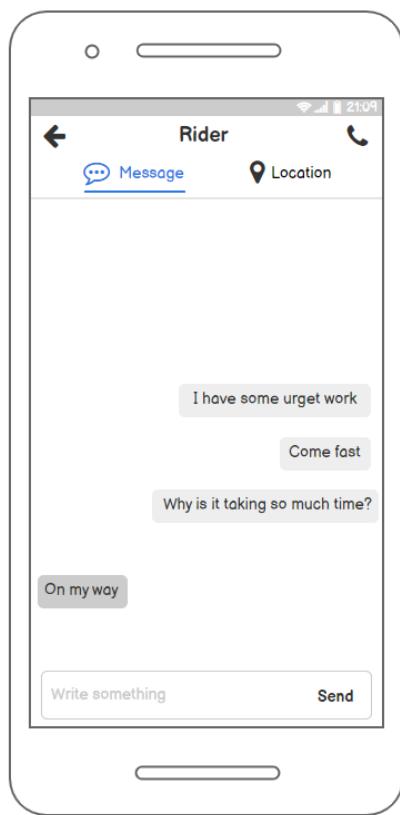
This is the inside view of food item page. Here a customer can see the rating, description of the item. Also, he/she can pick this item and also given the count number, he/she can add this item to cart. And also can see how much they put into the cart from the button.



Now, in the cart page a customer can view all of the item he/she add also can change the item amount and also can clear the whole cart. After that in the checkout page, the customer have to give the payment method and there also we show the customer the total payment of the order with details.

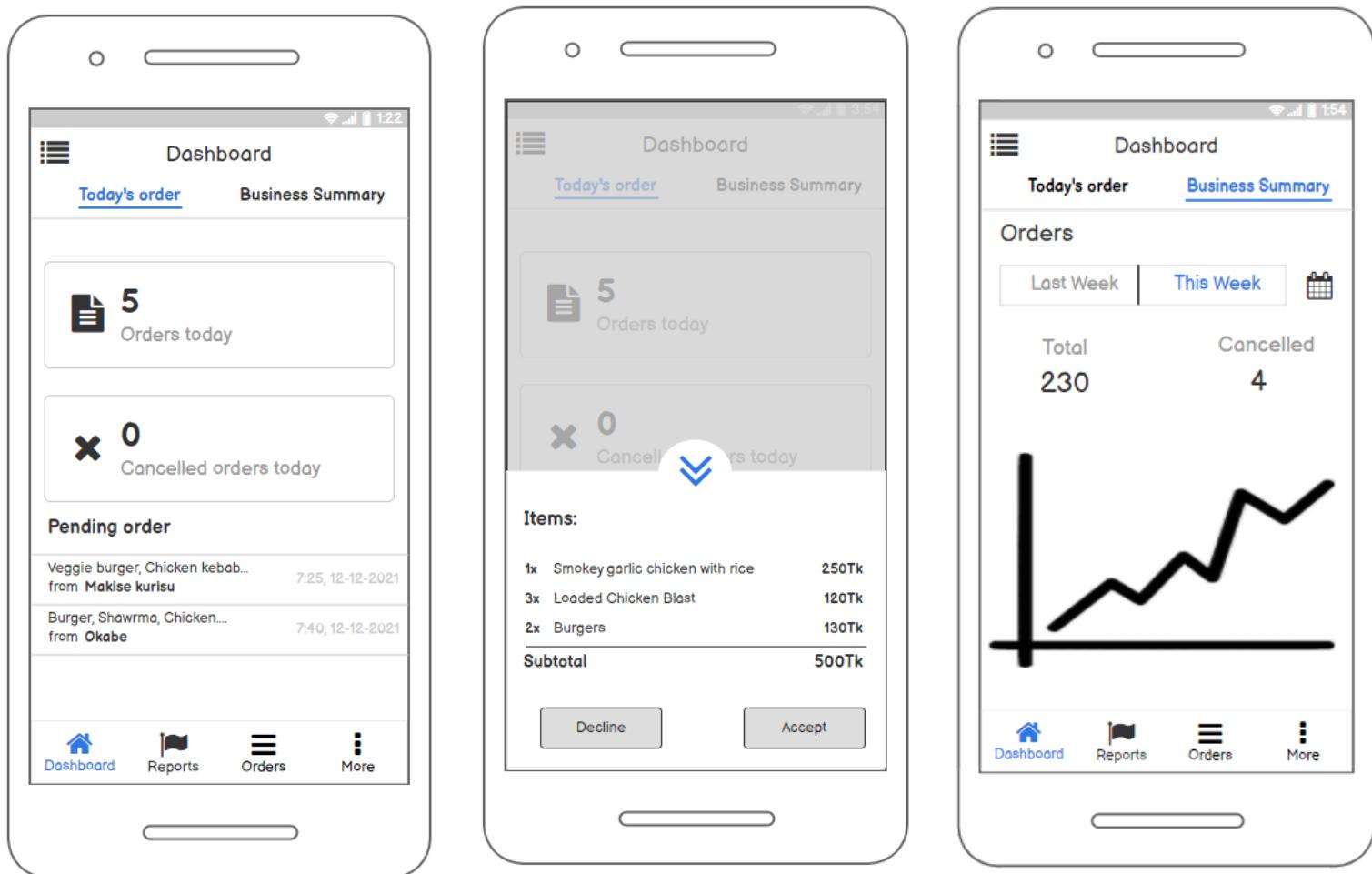


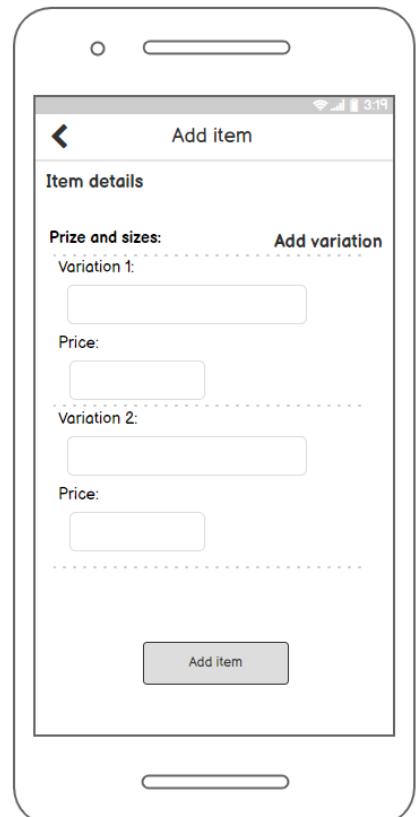
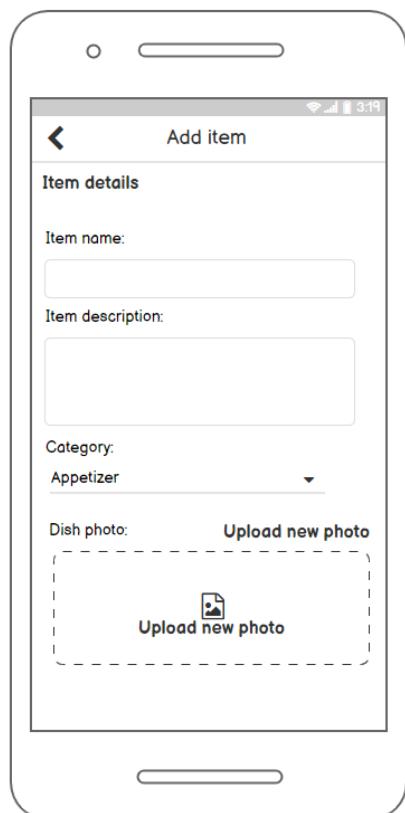
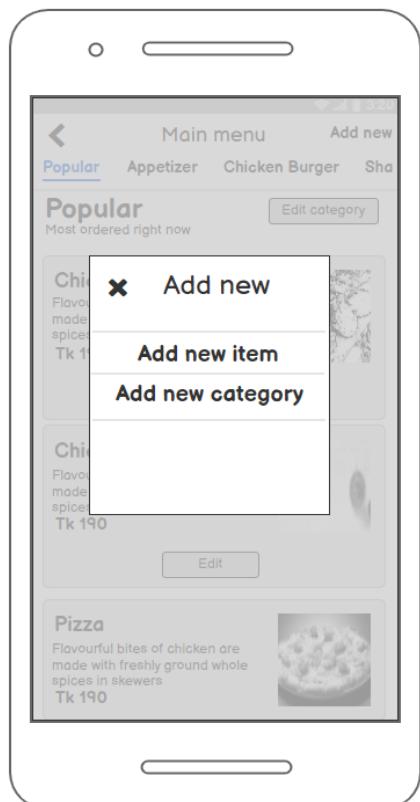
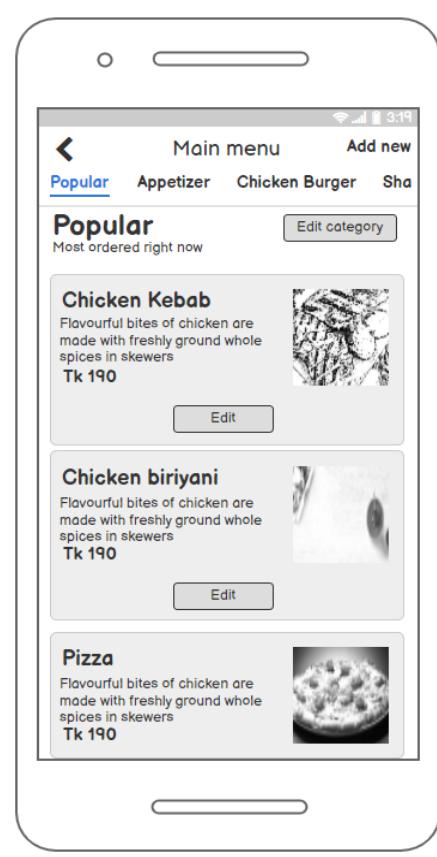
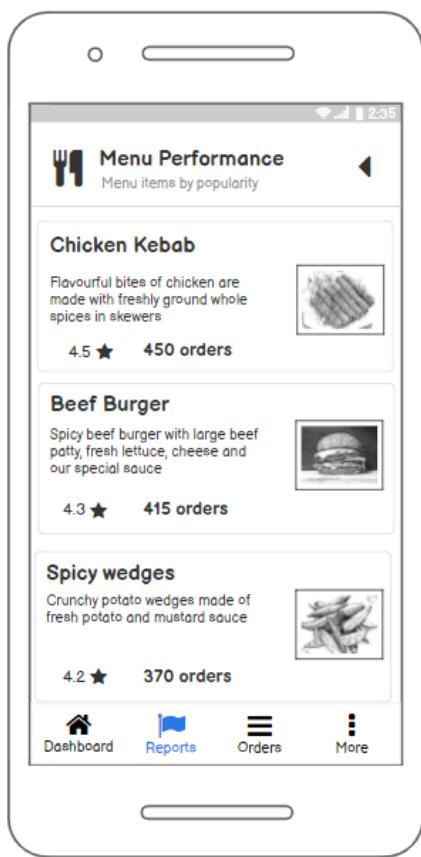
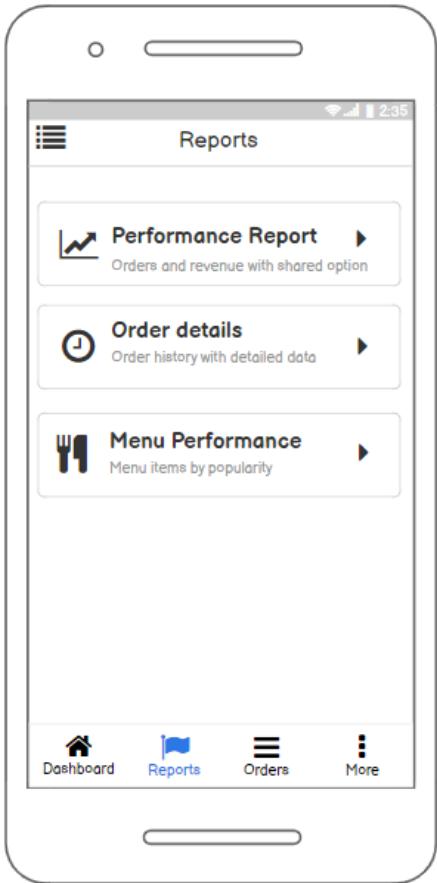
After a successful order, the customer can talk with the rider via phone call and thorough message. Also the customer can see the rider location via google map. Then there are couple of page for customer daily to daily details.



## 2. Restaurant Module:

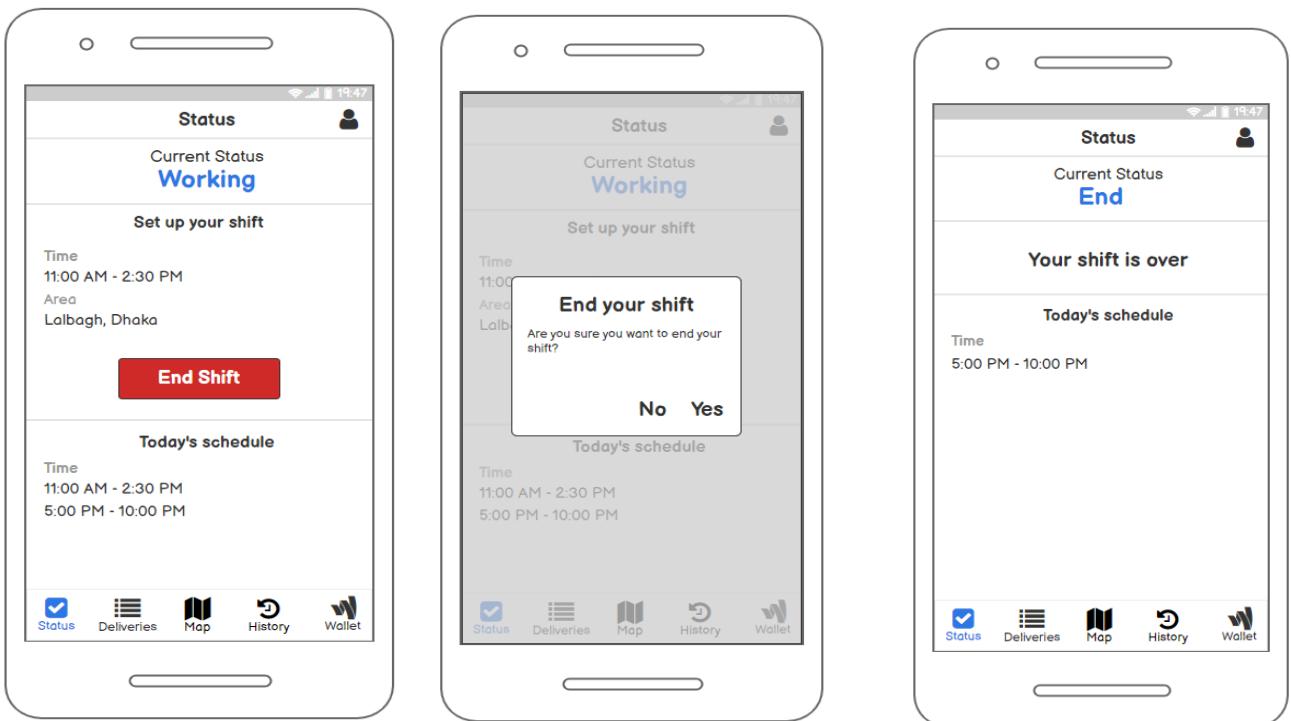
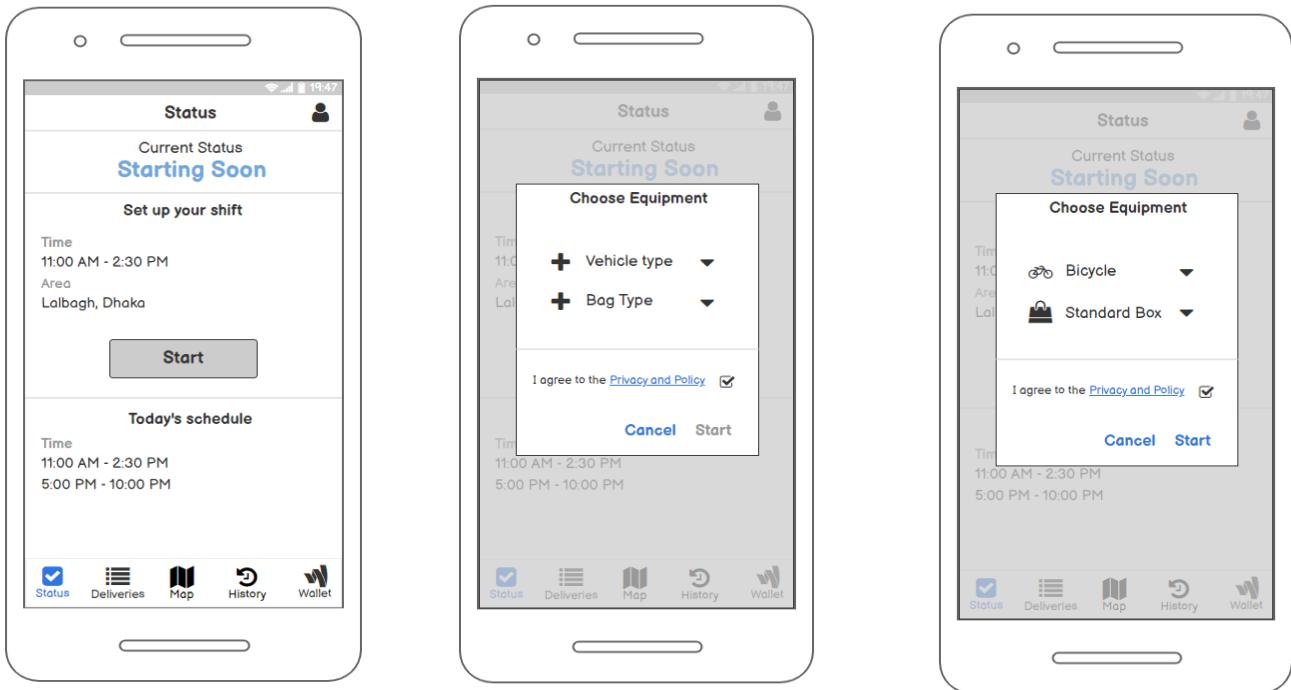
This part of the UI is basically for Restaurant manager. Here, the restaurant manager can view todays order and also can accept or deny any pending order. There is also a page for weekly business summary and performance of an item by ranking them total number of order the item receive. Also the manager can create a category or item and put them in the menu for the customer to order



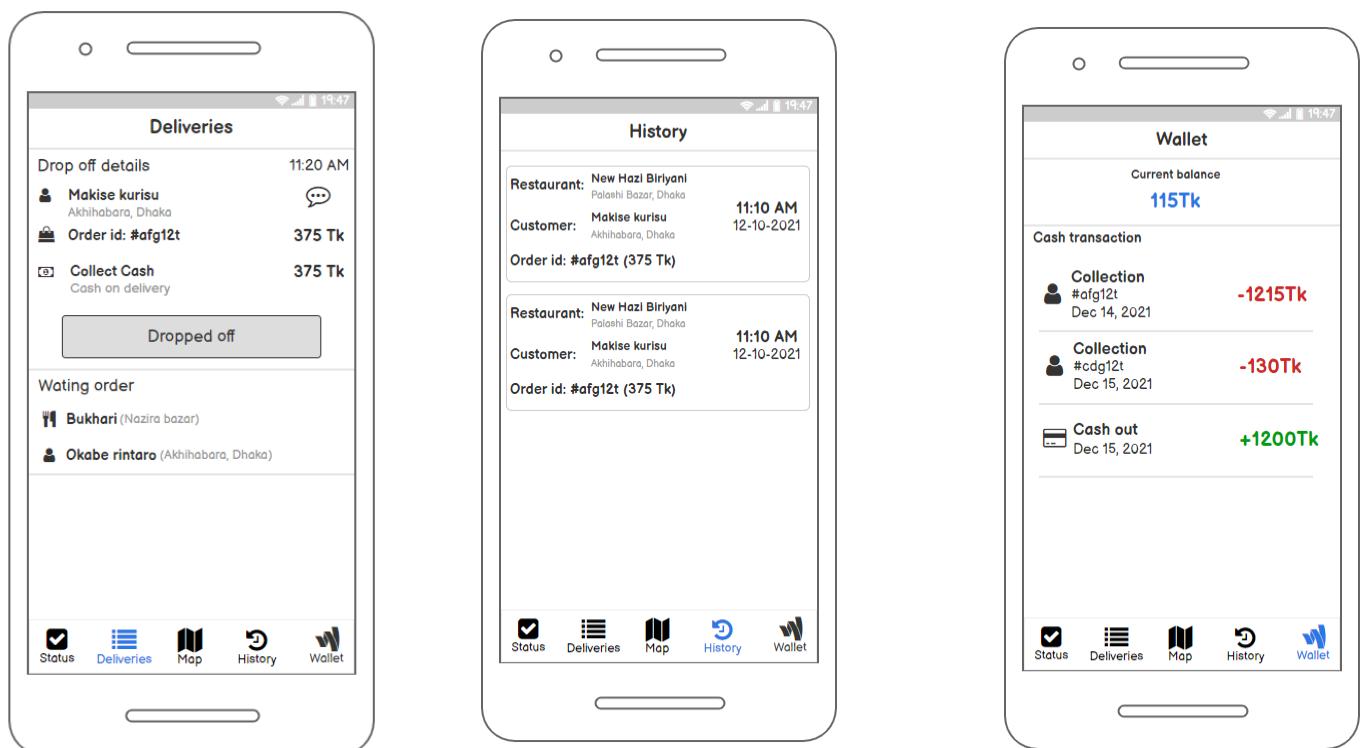
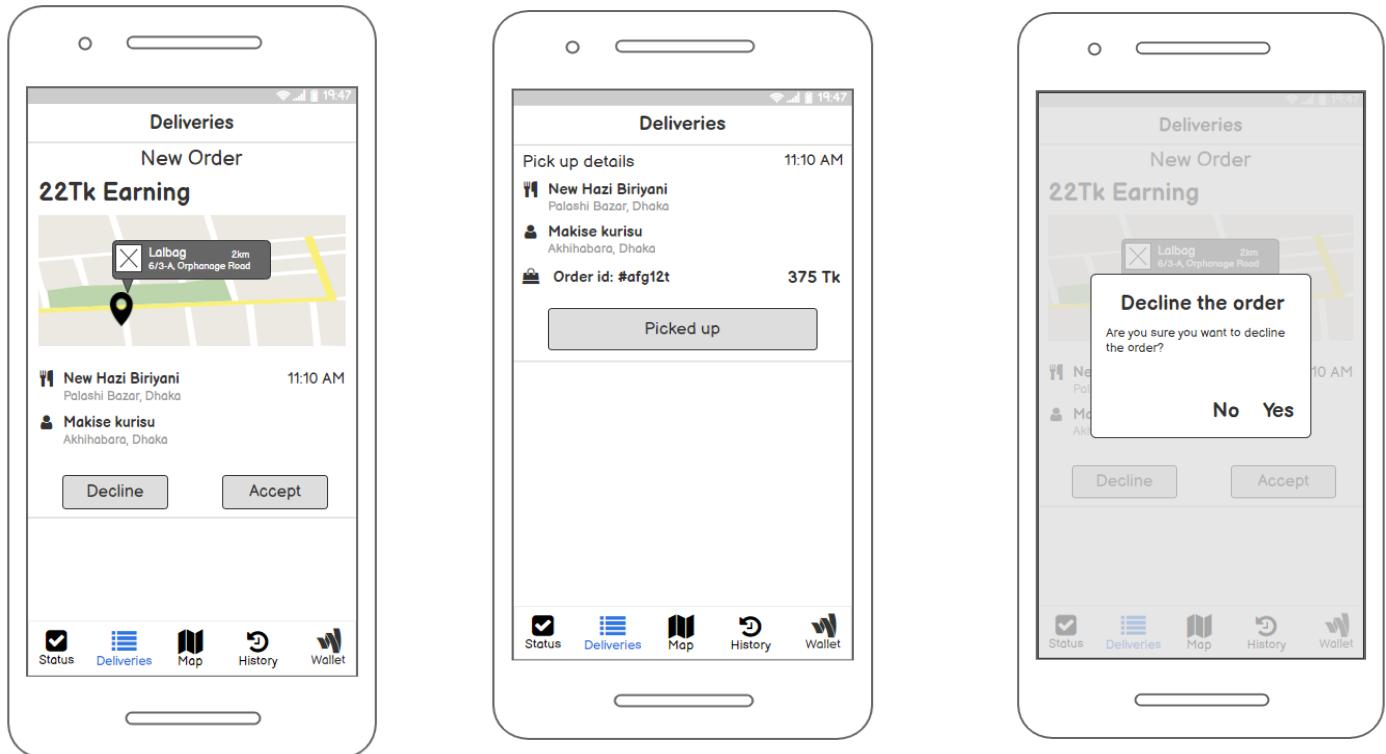


### 3.Rider Module:

Finally in the rider module, rider can start his/her shift by pressing the start button and also can choose the vehicle type and bag type.Also, the rider can end his shift



In this pages, A rider can accept any order or decline and also can talk with customer.The rider can show his past cash in or cash out money thorough the wallet page.



## The feedback provided in the class for Mock UI:

- 1.Sir told us to change the UI for restaurant module from app view to website view.As it will be easy for restaurant owner to view the menu list and easily can add/remove/edit item, category and price of the item.So, We implemented a website for restaurant module with using express framework and the details can be found in the snippets of the implemented use case/module section.
- 2.Sir also told us to add addon on the item section so that a customer can also select any optional item with an item.We also fix that in the later part of our implementation.We add this in the class diagram, ER diagram.
- 3.We considered to implemented to rider module later on.But as the workload was becoming higher, sir advised us not to implement this on this course and we can implement this in level 4 term 1 course.
- 4.Sir also told us to make a page to edit item and edit category, we also implemented this in the final part.

## Snippets of implemented use-case/module:

We basically implemented the add new category, add new item, edit item name, add new item and also add item to the category use cases. This page are part of the restaurant module. The restaurant manager will handle this pages to create/ edit item. The framework and language we used to implement this are –

- i) Node js
- ii) Express framework
- iii) Html, Css

We can see the details snippets of the implemented use case below-

### 1. Menu management page

The screenshot shows a web application interface for 'Hunger Express'. At the top, there's a navigation bar with links for Dashboard, Reports, Orders, Menu Management (which is currently active), and More. Below the navigation, there are two main sections. On the left, a sidebar lists categories: 'Category' (with 'Add new' button) and 'Item' (with 'Add new' button). Under 'Category', there's a list of items: 'Chicken biriyani', 'Chicken biriyani' (with an edit icon), 'Chicken Burger', 'French fry', 'Burger', 'Chicken biriyani', and 'hellloofafa'. On the right, the main content area displays a card for 'kacchi Biriyani' with a 'Add Item' button. Below it, another card shows 'Mutton Kacchi' with a description: 'Basmati rice, 2 piece mutton, 2 piece potato' and 'Count :1'. It has a 'Tk 140' price button and a 'Remove Item' button. To the right of these cards is a large image of a chicken biriyani dish. Below these cards is another card for 'Pizza' with a colorful graphic of a lion's face, a 'Description' field, and a 'Count :5' field.

## 2.Add new category

The screenshot shows the Hunger Express application interface. On the left, there is a sidebar with a list of categories: Hello, Buy 1 get 1, abcd, tafjalfdjlfajf, fafafafda, BurgerExpress, kacchi Biriyani, Mutton jali kabab, PizzaX, and Burger. Each item has a blue edit icon next to it. At the top right of the sidebar is a blue 'Add new' button. In the center, a modal window titled 'Add new category' is open. It contains a text input field with the placeholder 'create', a 'Category Name' input field, and a blue 'Submit' button. Below the modal, there is a progress bar. In the background, there is a blurred view of a menu item page with a biriyani dish and a colorful lion illustration.

## 3.Edit category

The screenshot shows the Hunger Express application interface. On the left, there is a sidebar with a list of categories: Hello, Buy 1 get 1, abcd, tafjalfdjlfajf, fafafafda, BurgerExpress, kacchi Biriyani, Mutton jali kabab, PizzaX, and Burger. Each item has a blue edit icon next to it. At the bottom of the sidebar are 'Item' and 'Add new' buttons. In the center, a modal window titled 'Edit category' is open. It contains an 'Edit category name' input field and a blue 'Submit' button. Below the modal, there are buttons for 'Tk 2' and 'Remove Item'. In the background, there is a blurred view of a menu item page with a biriyani dish and a colorful lion illustration. To the right, there is another blurred view of a menu item page for 'Chicken biriyani'.

## 4.Add new item

Hunger Express

Dashboard Reports Orders Menu Management More

### Create New Item

Item Name

Item Description

Category

Quantity (between 1 and 100):

Upload New Photo  
 No file chosen

Price, Weight and Sizes

Item type Name

Weight

Price(in Tk)

## 5.Item details

The screenshot displays the 'Hunger Express' application interface. At the top, there is a dark header bar with the brand name 'Hunger Express' on the left and navigation links for 'Dashboard', 'Reports', 'Orders', 'Menu Management', and 'More' on the right.

The main content area shows a large image of a dish labeled 'Mutton Kacchi'. Below the image, the item name 'Mutton Kacchi' is displayed in bold. A detailed description follows: 'Basmati rice, 2 piece mutton, 2 piece potato'. The item has a rating of '4.9' and a count of '1'.

Below this section, there are three blue buttons with white text: 'Available Options', 'Categories', and 'AddOns', each accompanied by a downward arrow indicating they are dropdown menus.

Further down, there is a section titled 'Reviews'. It includes a placeholder 'Customer Name' and a review entry with a rating of '4.2' and the text 'Very tasty food and fast service, saved my day'.

## Conclusion:

We tried to implement all the module and their corresponding UI, class diagram, Er diagram, sequence diagram. The tools we used to make different part are balsamiq.com for mock ui design, bpmn.io for bpmn diagram and draw.io to draw sequence diagram, Er diagram. There was

some problem in our implementation which then we corrected as sir told us to do. We are hopeful to implement the whole project in our next course.