



# Food Ordering and Management System

## Software Project 1

Submitted By

Name	ID	Department
Sayed, Mehedi	17-34364-1	BSc. CSSE
Shams, Tarik Bin	17-33997-1	BSc. CSSE
Alam, Md. Mahbub	17-34338-1	BSc. CSSE
Islam, Md Sameull	17-33950-1	BSc. SE

Under the Supervision of

Mohaimen-Bin-Noor

Lecturer

Department of Computer Science

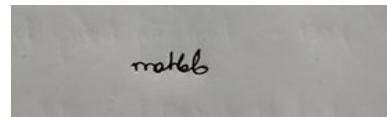
Faculty of Science and Technology

American International University-Bangladesh

November, 2020

## Declaration

This is to certify that this project is our original work. No part of this work has been submitted elsewhere partially or fully for the award of any other degree or diploma. Any material reported in this project has been properly acknowledged.

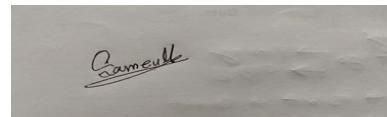
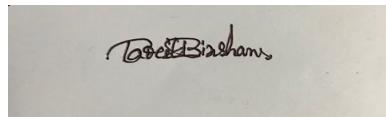


---

Sayed, Mehedi  
17-34364-1  
Bachelor of Science in Computer Science and  
Software Engineering  
American International University-  
Bangladesh

---

Alam, Md Mahbub  
17-34338-1  
Bachelor of Science in Computer Science and  
Software Engineering  
American International University-  
Bangladesh



---

Shams, Tarik Bin  
17-33997-1  
Bachelor of Science in Computer Science and  
Software Engineering  
American International University-  
Bangladesh

---

Islam, Md Sameull  
17-33950-1  
Bachelor of Science in Software Engineering  
American International University-  
Bangladesh

## Approval

The project titled “Food Order Management System” has been submitted to the following respected members of the board of examiners of the Department of Computer Science, American International University-Bangladesh in partial fulfilment of Software Project 1 Course on semester Spring 2019-2020 and has been accepted as satisfactory.

---

Mohaimen-Bin-Noor  
Lecturer  
Department of Computer Science  
Faculty of Science and Technology  
American International University-  
Bangladesh

---

Victor Stany Rozario  
Assistant Professor  
Department of Computer Science  
Faculty of Science and Technology  
American International University-  
Bangladesh

---

Dr. M M Mahbubul Syeed  
Associate Professor & Head [Undergraduate  
Program]  
Department of Computer Science  
Faculty of Science and Technology  
American International University-  
Bangladesh

---

Dr. Tafazzal Hossain  
Professor & Dean  
Department of Computer Science  
Faculty of Science and Technology  
American International University-  
Bangladesh

---

Dr. Carmen Z. Lamagna  
Vice Chancellor  
American International University-Bangladesh

## Acknowledgement

We are grateful to the Allah for giving us the effort and the good health to work on this project. This project is the result of our cordial effort and our supervisor's initiative and constant motivation. We wish to express our sincere thanks to our honorable supervisor Mohaimen-Bin-Noor, Lecturer, American International University – Bangladesh. His excellent supervision and constant support makes this project possible. We also want to thank our Vice Chancellor Dr. Carmen Z. Lamagna, our Dean Prof. Dr. Tafazzal Hossain and our head of department Dr. M M Mahbubul Syeed for their constant motivation and support.

# Contents

1	Introduction	9
1.0.1	Purpose . . . . .	9
1.0.2	Problem of the existing system . . . . .	9
1.0.3	Our Solution to the problem . . . . .	9
1.1	Project Scope . . . . .	10
2	System Study	11
2.1	List Of Domestic Applications/Systems . . . . .	11
2.2	List Of International Applications/Systems . . . . .	11
2.3	Features in the Existing Applications . . . . .	11
2.4	Features in Domestic Applications . . . . .	13
2.5	Features in International Applications . . . . .	15
3	Requirement collection and analysis	17
3.1	Overall Description . . . . .	17
3.1.1	Product Perspective . . . . .	17
3.1.2	User classes and Characteristics . . . . .	18
3.1.3	Operating Environment . . . . .	18
3.1.4	Design and implementation constraints . . . . .	18
3.2	System Features . . . . .	19
3.2.1	Customers . . . . .	19
3.2.2	Chef . . . . .	19
3.2.3	Cashier . . . . .	19
3.3	User Requirements . . . . .	19
3.3.1	Customers . . . . .	19
3.3.2	Chef . . . . .	20
3.3.3	Cashier . . . . .	21
3.4	Functional requirements . . . . .	21
3.4.1	Customers . . . . .	21
3.4.2	Chef . . . . .	22
3.4.3	Cashier . . . . .	22
3.5	Nonfunctional requirements . . . . .	22
3.6	Feasibility study . . . . .	22
3.6.1	Economic feasibility . . . . .	23
3.6.2	Technical feasibility . . . . .	23
3.6.3	Operational feasibility . . . . .	23
3.7	Cross-reference . . . . .	23
4	Modeling	24
4.1	Architecture & Design Principles . . . . .	24
4.2	Diagrammatic representation of the solution . . . . .	24
4.2.1	Structural Diagrams . . . . .	24
4.2.1.1	Class Diagrams . . . . .	24
4.2.2	Behavioral Diagrams . . . . .	25
4.2.2.1	Use case Diagrams . . . . .	25
4.2.2.2	Activity Diagrams . . . . .	28
4.3	Entity Relationship Model . . . . .	36
4.3.1	Basic ER Diagram . . . . .	36
4.3.2	Normalized ER Diagram . . . . .	37
4.4	Data Dictionary . . . . .	37

5	Development	42
5.1	User Interface Design . . . . .	42
6	Test strategy	45
6.1	Introduction . . . . .	45
6.1.1	Module . . . . .	45
6.2	Test plan . . . . .	45
6.2.1	Features to be Tested . . . . .	45
6.2.1.1	Customers Features . . . . .	45
6.2.1.2	Chef Features . . . . .	46
6.2.1.3	Cashier Features . . . . .	46
6.2.2	Features not to be Tested . . . . .	46
6.2.2.1	Customer Features . . . . .	47
6.2.2.2	Cashier Features . . . . .	47
6.3	Test Methodology . . . . .	47
6.3.1	Test Levels . . . . .	47
6.3.2	Roles and Responsibilities . . . . .	48
6.4	Test Deliverables . . . . .	48
6.4.1	Before testing phase . . . . .	48
6.4.2	During the testing . . . . .	48
6.4.3	After the testing phase . . . . .	48
6.5	Test Cases . . . . .	49
6.5.1	Customer Features . . . . .	49
6.5.2	Chef's Features . . . . .	61
6.5.3	Cashier Features . . . . .	67
6.5.4	Risk . . . . .	73
7	Deployment	74
8	Maintenance and Future Works	75
8.1	Maintenance . . . . .	75
8.2	Future Works . . . . .	75
9	User Manual	76
10	References	77

## List of Tables

1	Features in the Existing Applications . . . . .	13
2	Features in the Domestic Applications. . . . .	14
3	Features in the International Applications. . . . .	16
4	User Type Data Dictionary . . . . .	37
5	Food Category Data Dictionary . . . . .	38
6	Food Item Data Dictionary . . . . .	38
7	Order Data Dictionary . . . . .	39
8	Order Log Data Dictionary . . . . .	39
9	User Data Dictionary . . . . .	40
10	User Credential Data Dictionary . . . . .	40
11	Payment Data Dictionary . . . . .	41
12	Module . . . . .	45
13	Roles and Responsibilities . . . . .	48
14	FR_01 Test Case . . . . .	49
15	FR_02 Test Case . . . . .	50
16	FR_03 Test Case . . . . .	51
17	FR_04 Test Case . . . . .	52
18	FR_05 Test Case . . . . .	53
19	FR_06 Test Case . . . . .	54
20	FR_07 Test Case . . . . .	55
21	FR_08 Test Case . . . . .	56
22	FR_09 Test Case . . . . .	57
23	FR_10 Test Case . . . . .	58
24	FR_11 Test Case . . . . .	59
25	FR_12 Test Case . . . . .	60
26	FR_13 Test Case . . . . .	61
27	FR_14 Test Case . . . . .	62
28	FR_15 Test Case . . . . .	63
29	FR_16 Test Case . . . . .	64
30	FR_17 Test Case . . . . .	65
31	FR_18 Test Case . . . . .	66
32	FR_19 Test Case . . . . .	67
33	FR_20 Test Case . . . . .	68
34	FR_21 Test Case . . . . .	69
35	FR_22 Test Case . . . . .	70
36	FR_23 Test Case . . . . .	71
37	FR_24 Test Case . . . . .	72
38	FR_25 Test Case . . . . .	73
39	Risks during testing . . . . .	73

## List of Figures

1	Class Diagram . . . . .	24
2	Use Case Diagram . . . . .	25
3	Customer Use Case Diagram . . . . .	26
4	Chef Use Case Diagram . . . . .	27
5	Cashier Use Case Diagram . . . . .	27
6	Order Activity Diagram . . . . .	28
7	Search Food Activity Diagram . . . . .	29
8	Forget Password Activity Diagram . . . . .	30
9	Register Activity Diagram . . . . .	31
10	Add to Cart Activity Diagram . . . . .	32
11	Add to favourite Activity Diagram . . . . .	33
12	Close an item from menu Activity Diagram . . . . .	34
13	remaining Inventory Activity Diagram . . . . .	35
14	Basic ER Diagram . . . . .	36
15	Normalized ER Diagram . . . . .	37
16	Menu View . . . . .	42
17	Menu View 2 . . . . .	42
18	Profile View . . . . .	42
19	Favourites List View . . . . .	42
20	Login View . . . . .	43
21	Signup View . . . . .	43
22	Dashboard View . . . . .	43
23	Cart View . . . . .	43
24	Burger List View . . . . .	44
25	Add to Cart View . . . . .	44

# 1 Introduction

Restaurants have always played an essential role in the business, social, intellectual and artistic life of a thriving society. Restaurants are places of congregation and communication. It provides shelter and entertainments. The main role of restaurant is serving food to the customers. Food ordering and serving system is a manual process in many restaurants. Our proposed system is automated for food ordering and serving in a restaurant. Where customers come to the restaurant, they can be ordered food by a tablet. In a tablet food menu will show to the customers. They can see all available items and their prices. For payment, customers can use digital payment system like Bkash, Rocket, Debit or Credit card. After order confirmation by a customer, an order notification will go to the chef. This system will be time consuming for customers and cost effective for restaurant owners.

## 1.0.1 Purpose

The propose of the document will provide a detailed description of the requirements for Food Ordering and Management System. This will allow us for a complete understating of what we will be expected from the newly introduced system which is to be constructed. It will also explain system constraints, interface and interaction with other external application. Describing, Designing, constructing and testing the functions and specifications of the system is the primary goal of this document. The Project team will use the document to fully understand the expectation of this Food Ordering and Managing System to construct the appropriate application. The testing team will use this document as a baseline to see if the constructing team able to construct the system as user expectations.

## 1.0.2 Problem of the existing system

Nowadays people face some problem when they go to the restaurant for enjoy their meal.

- Customer can't customize their order.
- Waiter number limited in restaurant so, customer will be late to get their food.
- Customer didn't get their food processing notification.
- Waiter didn't note properly customer allergies note.
- Customer can't get their previous order history or favourite list.
- Customer can't give feedback immediately for their food.
- Customer can't filter food in many ways sitting in their table.

## 1.0.3 Our Solution to the problem

We propose a “Food Ordering and Management System” for the above mentioned problem. In the current situation, this type of system has not been activated. So we think “Food Ordering and Management System” will benefit many restaurant.

- Our system allow customers to place orders from their table which automatically gets sent to the kitchen without the need for waiters
- Customers can view their outstanding bill with the touch of a button.
- Customers can pay their bills sitting in the seats.

- Customers can customize menu as they wants.
- Customers can see their order's live status.
- Cashier can monitor every order and track of business.
- From the kitchen available orders and raw materials notification and chef approval will help to manage restaurant.
- This application will work as more then a waiter and will reduce the cost of hiring the waiter.

## 1.1 Project Scope

Food ordering management system is an android platform where customer can order sit at a table through a device. The customer will sign up with mobile number and password for login. Because for order any food customer has to login with the mobile number attach device in a table. The customer can search and filter for order food. The customer can customize the order as his/her wish with available item. The customer can view his/her previous order history and favorite list for helping the order previous favorite item. The customer can feedback for his/her experience and add notes before confirm the order. The customer can payment by card and by cash. If customer want to payment by cash, he/she has to go the cashier for order approval. The customer get notified the order status with the attached device. The cashier and chef can view the customer order history. The chef is notify the customer his/her order processing status. The chef is notify the cashier the available food item for cook. The cashier can control the food item view for customer with respect the availability. The cashier can manage and control the system.

## 2 System Study

This section illustrates the existing systems or applications that we studied before developing this system. We have observed fifteen (15) systems for this purpose. Among them seven (9) are International Systems/Applications and eight (6) are our Domestic Systems/Applications. We have marked the features that they have and extracted the features with respect to the applications.

### 2.1 List Of Domestic Applications/Systems

There are the list of domestic applications we have been study

- FoodPanda
- Uber Eats
- Hungry Naki
- Domino's Pizza
- FoodMart
- FoodBunny

### 2.2 List Of International Applications/Systems

There are the list of international applications we have been study

- Swiggy
- Burger King
- Pizza Hut (Brunei)
- Pizza Hut (UAE)
- Pizza Hut (Singapore)
- MC Delivery
- KFC
- Glovo
- Jumia Food

### 2.3 Features in the Existing Applications

SL.No	Features List
1	Automatic Select Location for delivery food by GPS
2	Select location manually for delivery food
3	Select location manually for pick up food
4	Last delivery location
5	Search restaurant
6	Search food items

7	Filter by newly launched items
8	Filter by top rated items
9	Filter by Price
10	Filter by Offers
11	Filter by Cuisines
12	Filter by items
13	Show availability restaurant
14	Show availability food items
15	Filter by delivery time
16	Filter by food items category
17	Show sub-category of food items inside of category
18	Sign up (Name, Email, Password, Mobile Number)
19	Sign in by phone number
20	Sign in by Facebook
21	Sign in by Gmail
22	Sign in by Email address and password
23	Forget password (Email)
24	Show offers
25	User can enable special offers notifications
26	Multiple address for delivery food
27	Edit quantity before place the order
28	Use promo code for discount
29	Previous Order history
30	Payment by Cash on pick up
31	Payment by mobile banking
32	Payment by internet banking
33	Payment by Credit or Debit Card
34	Payment by cash on delivery
35	Help Centre
36	Order process alert by messages
37	Quick menu
38	Customize menu
39	Add on
40	Add to favourite
41	Add to cart
42	Filter by best sellers' items
43	Favourite item list
44	Suggest extra items
45	Select offers
46	Delivery now
47	Set a time and date for post-delivery
48	Track current order by mobile number
49	Track current order by order ID
50	Meet at vehicle for pick up food item
51	Store select for pick up order

52	Delivery notes					
53	Save the address					
54	Recommended shops					
55	Filter by most popular shops					
56	Filter by rating					
57	Filter by dietary					
58	Edit account					
59	Minimum order					
60	Must select a drink for order					
61	Send feedback					
62	Sign Out					
63	Defined Allergies					

Table 1: Features in the Existing Applications.

## 2.4 Features in Domestic Applications

No	Food Panda	Uber Eats	Hungry Naki	Domino's Pizza	Food Mart	Food Bunny	Count
1	✓	✓	✓	✓			4
2	✓	✓	✓	✓	✓		5
3				✓	✓		2
4							0
5	✓	✓			✓	✓	4
6	✓	✓	✓	✓	✓	✓	6
7	✓						1
8	✓						1
9	✓	✓					2
10	✓						1
11	✓						1
12	✓	✓					2
13	✓	✓					2
14	✓	✓					2
15	✓	✓		✓			3
16	✓						1
17	✓						1
18	✓	✓	✓	✓	✓	✓	6
19		✓	✓	✓	✓	✓	5
20	✓				✓		2
21					✓		1
22	✓				✓		2
23	✓						1
24	✓			✓			2
25	✓						1
26	✓						1
27	✓	✓		✓			3

No	Food Panda	Uber Eats	Hungry Naki	Domino's Pizza	Food Mart	Food Bunny	Count
28	✓	✓					2
29	✓			✓	✓		3
30				✓			1
31			✓	✓	✓	✓	4
32							0
33	✓	✓	✓	✓	✓		5
34	✓		✓			✓	3
35	✓						1
36	✓						1
37				✓			1
38		✓		✓			2
39		✓		✓			2
40		✓	✓	✓			3
41		✓		✓		✓	3
42				✓			1
43				✓			1
44				✓			1
45	✓			✓			2
46				✓			1
47							0
48				✓			1
49				✓		✓	2
50				✓			1
51							0
52		✓					1
53		✓					1
54		✓					1
55		✓					1
56		✓					1
57		✓					1
58	✓	✓					2
59							0
60							0
61							0
62	✓	✓	✓	✓	✓	✓	6
63							0
Count	32	24	10	27	13	9	

Table 2: Features in the Domestic Applications.

## 2.5 Features in International Applications

No	Swiggy	Burger King	Pizza Hut (Brunei)	Pizza Hut (UAE)	Pizza Hut (Singapore)	MC Delivery	KFC	Glovo	Jumia Food	Count
1	✓		✓	✓	✓	✓	✓	✓	✓	8
2	✓		✓	✓	✓	✓	✓	✓	✓	7
3	✓				✓	✓				3
4	✓									1
5	✓		✓	✓		✓	✓	✓		6
6	✓	✓					✓	✓	✓	5
7	✓	✓					✓		✓	4
8	✓				✓		✓		✓	4
9	✓				✓	✓	✓	✓	✓	6
10	✓						✓	✓		3
11	✓									1
12	✓				✓			✓	✓	4
13	✓				✓			✓		3
14	✓			✓	✓	✓		✓		5
15										0
16	✓	✓					✓	✓	✓	5
17	✓							✓		2
18	✓	✓	✓	✓	✓	✓	✓	✓	✓	9
19	✓	✓	✓	✓		✓		✓	✓	7
20	1	✓	✓	✓				✓	✓	5
21	1	✓	✓	✓	✓			✓	✓	6
22	✓	✓	✓	✓	✓	✓	✓	✓	✓	9
23		✓		✓	✓				✓	4
24	✓			✓	✓		✓			4
25				✓						1
26										0
27	✓	✓		✓	✓	✓	✓		✓	7
28	✓	✓	✓	✓		✓	✓		✓	7
29		✓		✓						2
30				✓	✓		✓			3
31				✓			✓	✓		3
32				✓			✓	✓		3
33	✓	✓	✓	✓	✓	✓	✓	✓	✓	9
34	✓	✓	✓	✓	✓	✓	✓	✓	✓	9
35					✓					1
36				✓						1
37	✓	✓	✓	✓	✓	✓	✓		✓	8
38	✓	✓	✓	✓	✓	✓	✓		✓	8
39		✓	✓	✓	✓	✓			✓	6

No	Swiggy	Burger King	Pizza Hut (Brunei)	Pizza Hut (UAE)	Pizza Hut (Singapore)	MC Delivery	KFC	Glovo	Jumia Food	Count
40		✓	✓	✓		✓				4
41	✓	✓	✓	✓	✓	✓	✓		✓	8
42	✓									1
43		✓	✓	✓		✓				4
44					✓	✓	✓		✓	4
45	✓	✓	✓	✓		✓	✓		✓	7
46			✓	✓	✓					3
47			✓	✓	✓					3
48			✓	✓	✓					3
49	✓	✓	✓	✓	✓	✓	✓			7
50				✓						1
51			✓	✓	✓		✓			4
52			✓	✓					✓	3
53			✓	✓						2
54								✓		1
55	✓							✓		2
56	✓						✓	✓		3
57	✓									1
58		✓	✓	✓				✓		4
59	✓							✓		2
60		✓				✓		✓		3
61	✓									1
62					✓			✓		2
63								✓		1
total	34	24	26	38	28	22	27	22	31	

Table 3: Features in the International Applications.

### 3 Requirement collection and analysis

Software requirements describe the background, business goals and constraints of the project. We choose agile methodology for our requirement collection because Agile methodology is where demands and solutions evolve through the collaborative effort of self-organizing and cross-functional teams and their customers. Our user can change their requirement in the mean time of development phase. Because agile is built around the principle that project scope can and should change during the project. Agile software development refers to software development methodologies centered round the idea of iterative development. We develop each module individually and deliver to the customer when each module development is finished. It's iteration process will continue until every module finish to develop. In the meantime of development process user can change their specification because not every restaurant order and serve process are same and their requirement should be not will be the same. It is commonly used to deliver complex projects due to its adeptness. It emphasizes collaboration, flexibility, continuous improvement, and high quality results. The ultimate value in Agile development is that it enables teams to deliver value faster, with greater quality and predictability, and greater aptitude to respond to change.

#### 3.1 Overall Description

##### 3.1.1 Product Perspective

The Food ordering and management system is an Android based application and hence will require an android device and internet connection. The application will be able to connect to server and will have an SQLite database server. It will use JAVA and XML as language and android stdio as development tool for it's development. This system provides simple mechanism for users to order food without waiter. The followings are the main features that are included in second hand book buy and sell system.

- User Account : The system allows the customer to create their accounts in the system and provide features of updating and viewing profiles.Cashier will create an account for chef to updating and viewing.
- Login : After registration customers can login into the system to order foods.Chefs can login to approve orders.
- Previous Order List : Customers can see their previous order list.
- Quick order : Customers can order their previous order quickly.
- Search : Customers can search any food item.Also can filter by newly launched items,top rated items,price,cooking time.
- Offers : Customers can see current offers and cashier can add,modify and delete any offer.
- Notification : Customers will get changing status of cooking their orders and chef and cashier will get order place notification and raw material availability notification. Chef changing status of cooking will notify customers.
- Order Notes : Customers can add any notes to their order like allergies, like or dislikes.
- Feedback : Customers can review and give feedback notes to any item will anonymous. Chef and Cashier will able see those feedback.
- Sign Out : Every types of account can sign out.

### 3.1.2 User classes and Characteristics

There are three types of users that interact with the system: customers, chefs and cashier. Each of these three types of users has different use of the system, so each of them has their own requirements.

- Customers: Customers can order any items from the menu. Before ordering they have to register to the system by phone number and password. They can pay for their order by Mobile banking or Debit and Credit card. After ordering they can see food processing step with notifications.
- Chefs: When a customer order some food, the full order history is forwarded to the chef. Chef can update food processing status.
- Cashier: Cashier helps to the customer for payment with digital banking system and check transaction is successful or not. If any customer want to pay with hand cash, cashier will collect the cash. Cashier can see every customer's orders details. Cashier should report to the owner's about profit of the restaurant after a month.

### 3.1.3 Operating Environment

This is a Android base application. So, customer can access the system on mobile or tablet of the restaurant.

- OS: Android
  - Version : Up to 5.0
- Develop Tool:
  - Language : JAVA, XML
  - Database : SQLite
  - Authentication : API
  - IDE: Android Studio

### 3.1.4 Design and implementation constraints

- The information of all the users must be stored in a database that is accessible by the cashier.
- Mobile number and password used for the authentication of customers.
- Customers will get the OTP after forget the password for recover their profile.
- The system only the android platform friendly.
- Since red and green color blindness is most common, red and green color shall not be used for important texts.
- This android environment shall not contain blue text which will confuse users to be links.
- This application will be developed using the android studio IDE.
- This android application must be comply with government regulations for usage by visually impaired persons.

## 3.2 System Features

Description of features are

### 3.2.1 Customers

- Customers have to register to the system by providing necessary information's (phone number). Registered customers can access all the functionalities provide on the portal. They can have their own profiles through this feature.
- They can login to the system by providing their phone number and password.
- They can reset their password if they forget.
- An authenticated customer can order one or multiple food. Also can add food items to their favorite list.
- Customer can pay online with credit/debit card or mobile banking.
- They can add note when they place an order.
- They can search food by filtering option. (New item, most popular, price off etc.)
- Customer can give feedback about the food.
- They can track the food processing step with notification.
- Customer can see his/her order history.

### 3.2.2 Chef

- When the order will be forward to the chief, Chief will update the food processing status.

### 3.2.3 Cashier

- Cashier can login to the system to get all the features.
- As there is online payment system but customer can pay bill with hand cash. So cashier will collect bill for the order.
- Cashier also can check the food processing status of all orders
- He/she can check order list with other details.
- Also they can count the average sell / order / profit etc.
- Cashier can monitor all things.

## 3.3 User Requirements

### 3.3.1 Customers

- As a customer, I want to sign up with phone number so that I can login to the system.
- As a customer, I want to search food item so that I can get the food item easily.
- As a customer, I want to sign in with phone number so that I can use the system as a valid user.

- As a customer, I want to filter new launched items so that I can see the new item.
- As a customer, I want to filter item so that I can get those item foods. [Ex. Thai, Chinese etc.].
- As a customer, I want to filter the price of foods so that I can get food on my budget.
- As a customer, I want to filter foods offer so that I can claim them.
- As a customer, I want to add favorite food item to the favorite list so that I can easily order my favorite food. (Don't need to search multiple time).
- As a customer, I want to check my favorite list so that I can order my favorite food (I can analysis that which I have like most).
- As a customer I want to add food item to the cart so that I can order food.
- As a customer, I want to get food processing notification so that I can check the food processing status.
- As a customer, I want to see the review of the food item so that I can get idea about the food quality.
- As a customer, I want to send feedback about food so that fault can be overcome next if there is any problem with food.
- As a customer, I want to write order note when I will order foods so that I can recommend my needs.
- As a customer, I want to reset my password (so that) if I forget password.
- As a customer, I want to sign out so that I can exit the system and other customer can login again.
- As a customer, I want to show my previous order history so that I can analysis that what I have ordered most / I can re-order them easily.
- As a customer, I want to use online payment system (credit/debit/mobile banking) card so that I can pay the bill online easily.

### 3.3.2 Chef

- As a chef, I want to approve the orders so that i can know which food has to make and is it possible or not.
- As a chef, I want to change the status of cooking process so that customers can know their orders status.
- As a chef, I want to notify the customer when food cooking is finished so that customers can take their meals.
- As a chef, I want to get notification of remaining inventory so that i can approve the order.
- As a chef, I want to close the item after inventory finish so that customers can't order the item.
- As a chef, I want to see the reviews so that i can know the shortage in the food.
- As a chef ,I want to see the order notes so that i can prepare the food according to the customers will.

### 3.3.3 Cashier

- As a cashier. I want to have track of every orders so that i can know which order has been finished,cooking and finished.
- As a cashier, I want to add offers in the offers list so that customers can get offers.
- As a cashier, I want to give discount in any order so that customers keep coming.
- As a cashier, I want to get notification of remaining inventory so that i can buy or order them quickly.
- As a cashier, I want to create a profile or hire a chef in the system so that he can do his role.
- As a cashier, I want to fire chef so that he/she can leave the job.
- As a cashier, I want to see the reviews of the foods so that i can tell chef for improvement.
- As a cashier, I want to update inventory so that chef can know the remaining inventory.
- As a cashier. I want to close the item if the inventory is not available so that customers can't order the item.

## 3.4 Functional requirements

### 3.4.1 Customers

- The customers shall be able to register to the system with their phone number.
- The customers shall be able to login to the system.
- The customers shall be able to search food items.
- The customers shall be able to filter item with category.
- The customers shall be able to filter item with items prices.
- The customers shall be able to filter food offers.
- The customers shall be able to add favorite food items.
- The customers shall be able to check their favorite food items list.
- The customers shall be able to add food to cart.
- The customers shall be able to get food processing notifications.
- The customers shall be able to see review of foods.
- The customers shall be able to send feedback about the food.
- The customers shall be able to write order note.
- The customers shall be able to reset password.
- The customers shall be able to sign out from the system.
- The customers shall be able to re-ordered items which had been ordered previously.
- The customers shall be able to pay by online payment system.

### 3.4.2 Chef

- The chefs shall be able to approve the orders.
- The chefs shall be able to change food processing status.
- The chefs shall be able to notify the customers when food cooking is finished.
- The chefs shall be able to notify the remaining inventory.
- The chefs shall be able to close items when inventory is finished.
- The chefs shall be able to see orders note.

### 3.4.3 Cashier

- The cashier shall be able to track every orders.
- The cashier shall be able to add offers.
- The cashier shall be able to give discount.
- The cashier shall be able to get notification of remaining inventory.
- The cashier shall be able to create profile of a new chef.
- The cashier shall be able to fire a chef.
- The cashier shall be able to see reviews of foods.
- The cashier shall be able to update inventory.
- The cashier shall be able to close item when inventory is finished.

## 3.5 Nonfunctional requirements

- The front-page load time must be no more than 3 seconds for users.
- Passwords shall never be viewable at the point of login or at any other time.
- The system should be capable enough to handle 500 users without affecting its performance.
- System shall be able to process a notification in 1 second or less.
- Correct identification of users attempting access to the systems and protection of the systems from unauthorized users.
- Only cashier can see all orders and payment history.
- Only chef can update orders processing status.
- System should be portable for different Android versions.

## 3.6 Feasibility study

A feasibility study is an assessment of the practicality of a proposed project or system. There are some feasibility have been studied for this project.

### 3.6.1 Economic feasibility

Analysis of a project's costs and revenues in an effort to determine whether it or not it is logical and possible to complete.

### 3.6.2 Technical feasibility

Technical feasibility helps organizations determine whether the technical resources meet capacity and the team is capable to build the system. Technical feasibility also involves evaluation of the hardware, software, and other technical requirements of the proposed system.

### 3.6.3 Operational feasibility

Operational Feasibility, we consider whether the current system become implemented using existing human resource or not. To find functional feasibility we determine whether the proposed solution can participate in existing operations and whether the right information in the right time is, provide to end users. Operational feasibility of our proposed system is modularized.

## 3.7 Cross-reference

In our country, there are no mobile application for restaurants where customers can order food sitting at the table with a android device in the restaurant without waiter. But in our country there are many application where customers can order food from anywhere for home delivery. We want to make an android application where customers can order food and customize it sitting at the table in the restaurant. Customers also can create a favorite list so that they can order next time easily and quickly. Customers can give feedback for the experience and add notes for the order.

There are three actors in this android application. They are cashier, chef and customer. For this android application documentation we use Microsoft Word 2016. We also use Google docs in Google drive, so that every member in our group can read and write same document. We search for sample of SRS and IEEE standard format of SRS in Google with internet browser like Google Chrome, Firefox etc.

## 4 Modeling

### 4.1 Architecture & Design Principles

A software architecture describes its group of components, their connections, interactions among them and deployment configuration of all components. This software architecture can be defined in UML (Unified Modeling Language). UML stands for Unified Modeling Language. It is a pictorial language used to make software blueprints. The elements are like components which can be associated in different ways to make a complete UML picture, which is known as a diagram. We have two broad categories of diagrams

- Structural Diagrams
- Behavioral Diagrams

### 4.2 Diagrammatic representation of the solution

#### 4.2.1 Structural Diagrams

##### 4.2.1.1 Class Diagrams

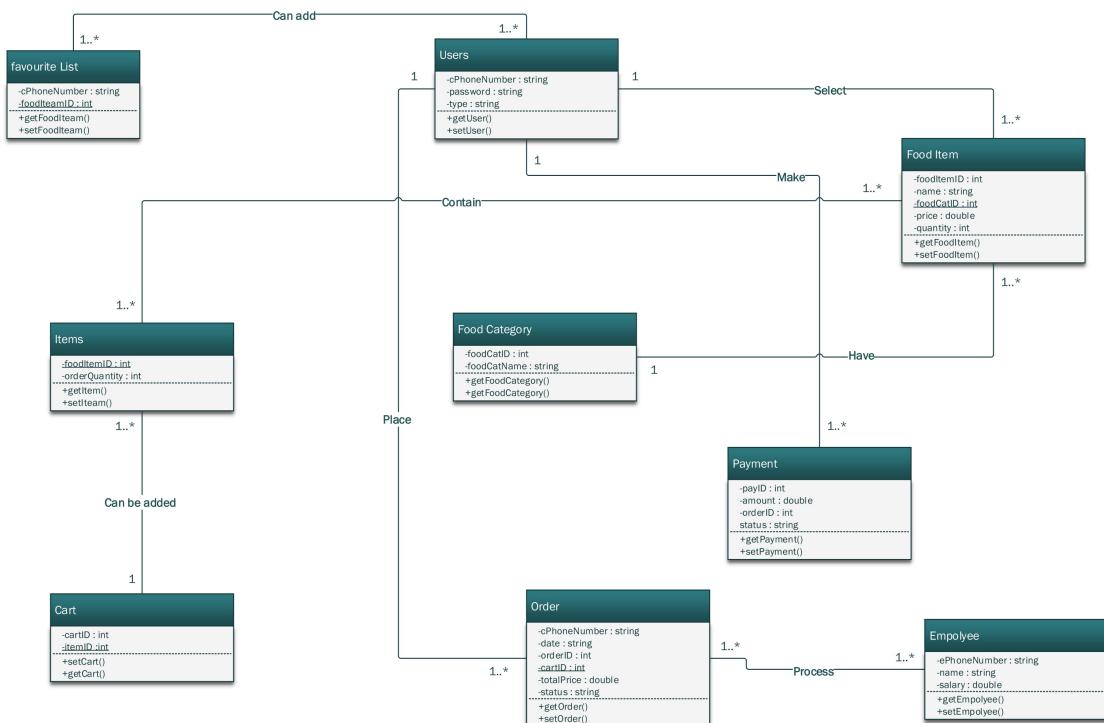


Figure 1: Class Diagram

## 4.2.2 Behavioral Diagrams

### 4.2.2.1 Use case Diagrams

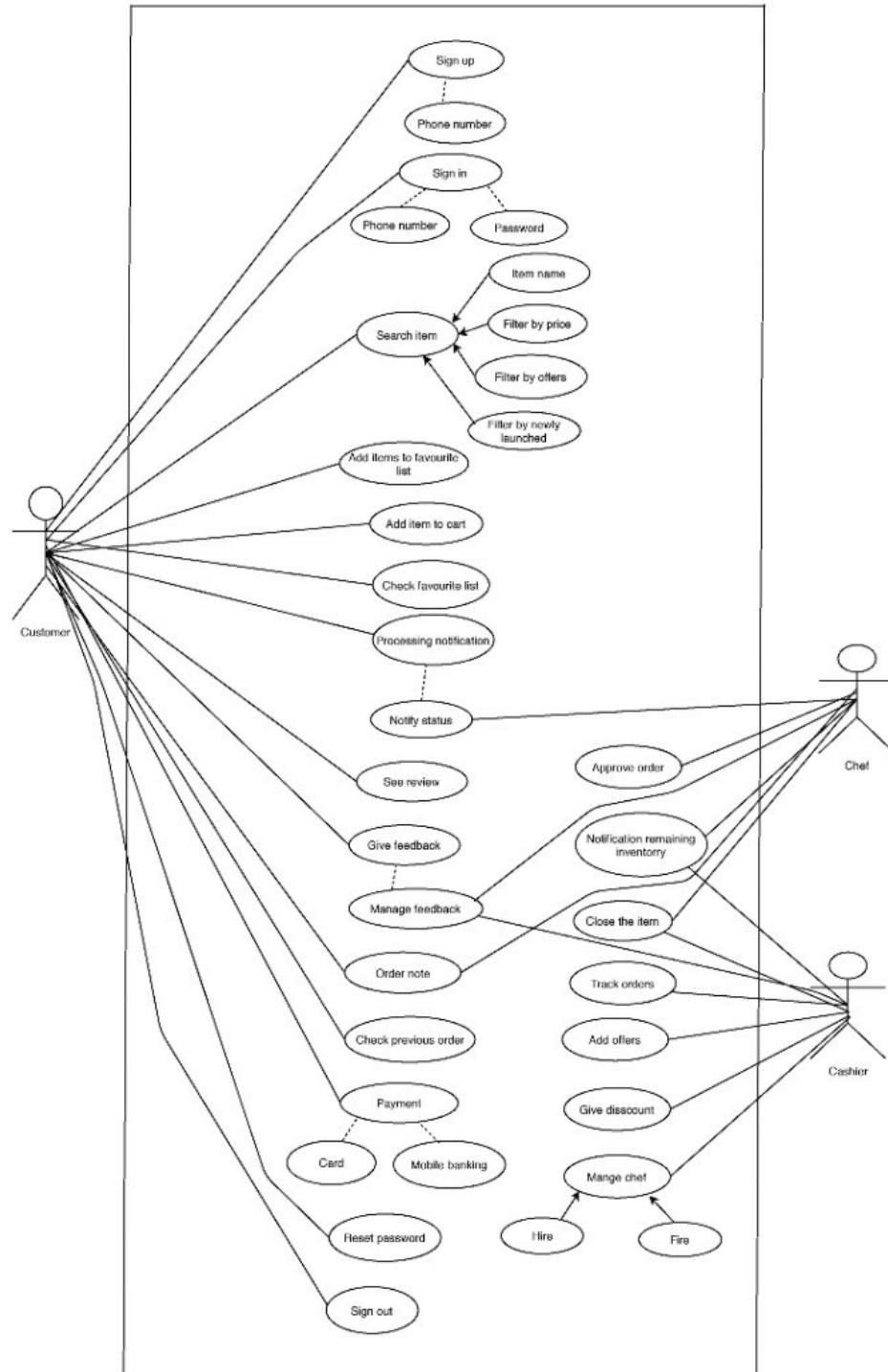


Figure 2: Use Case Diagram

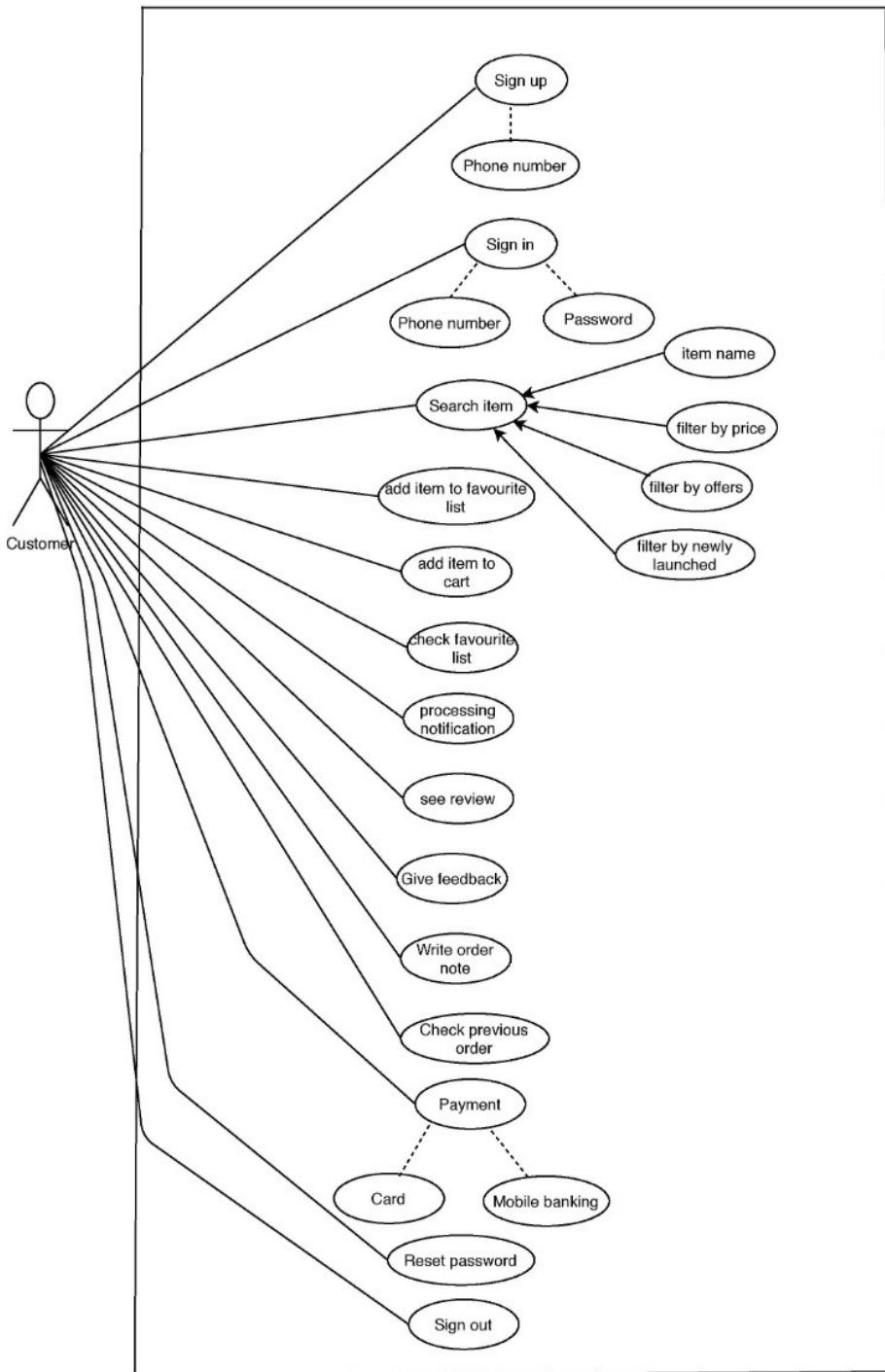


Figure 3: Customer Use Case Diagram



Figure 4: Chef Use Case Diagram

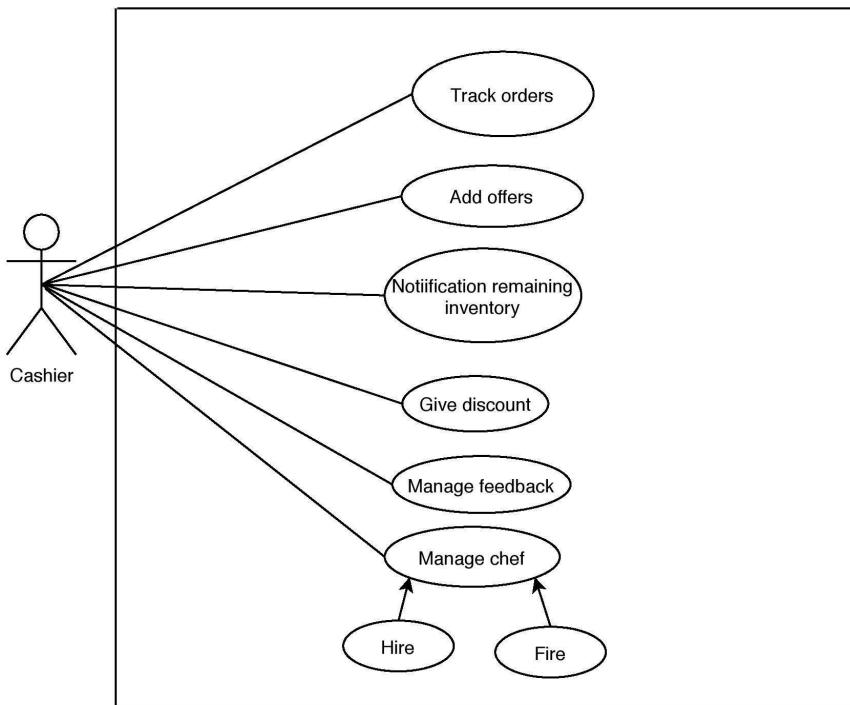


Figure 5: Cashier Use Case Diagram

#### 4.2.2.2 Activity Diagrams

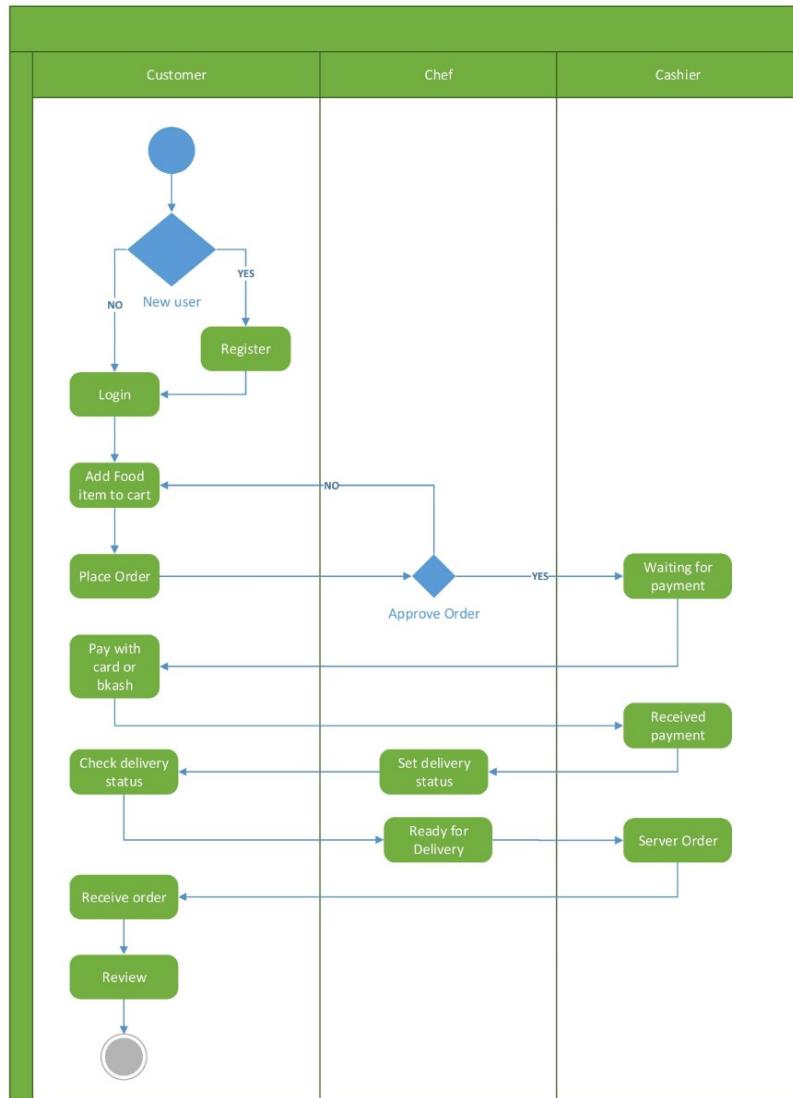


Figure 6: Order Activity Diagram

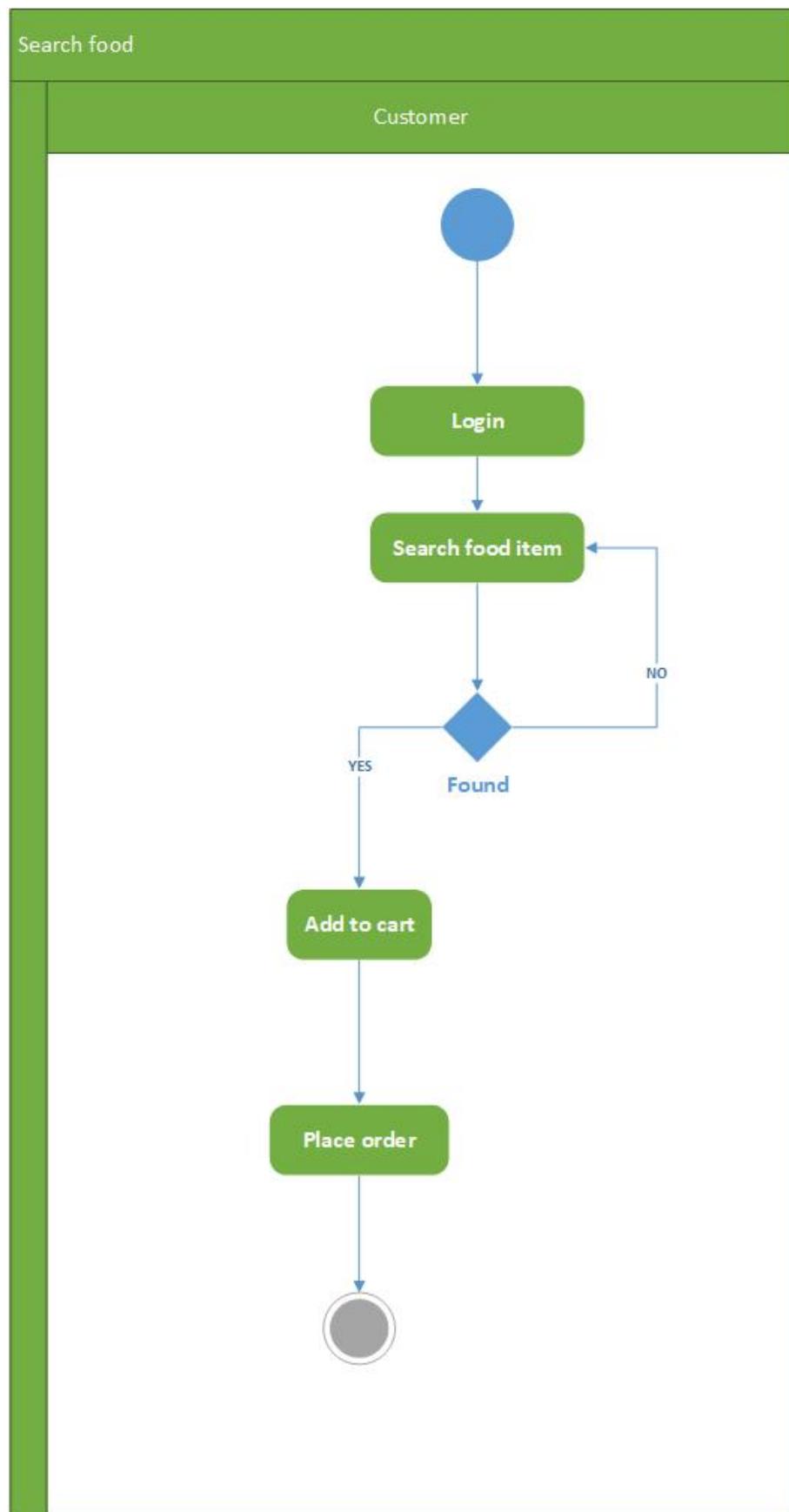


Figure 7: Search Food Activity Diagram

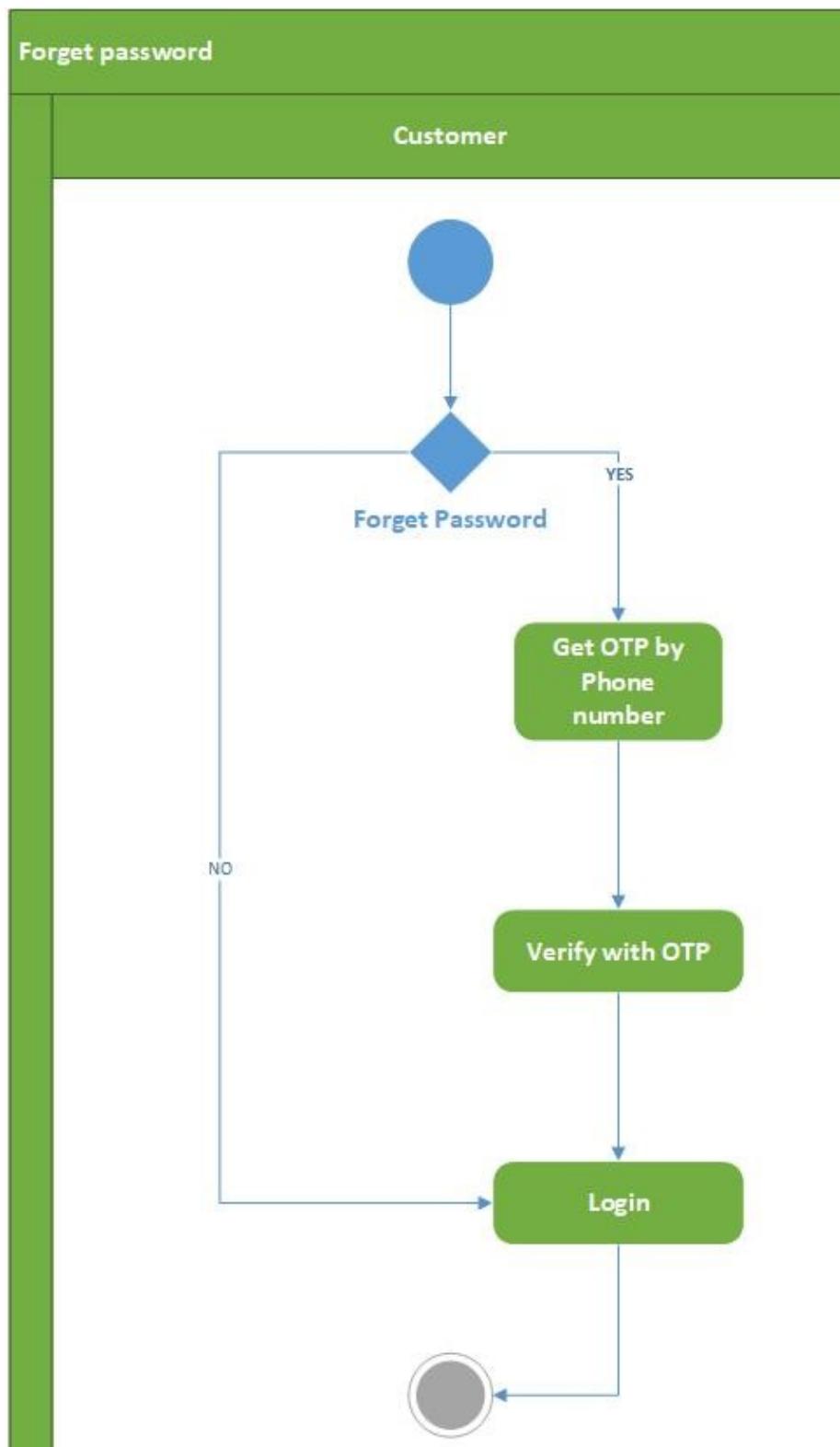


Figure 8: Forget Password Activity Diagram

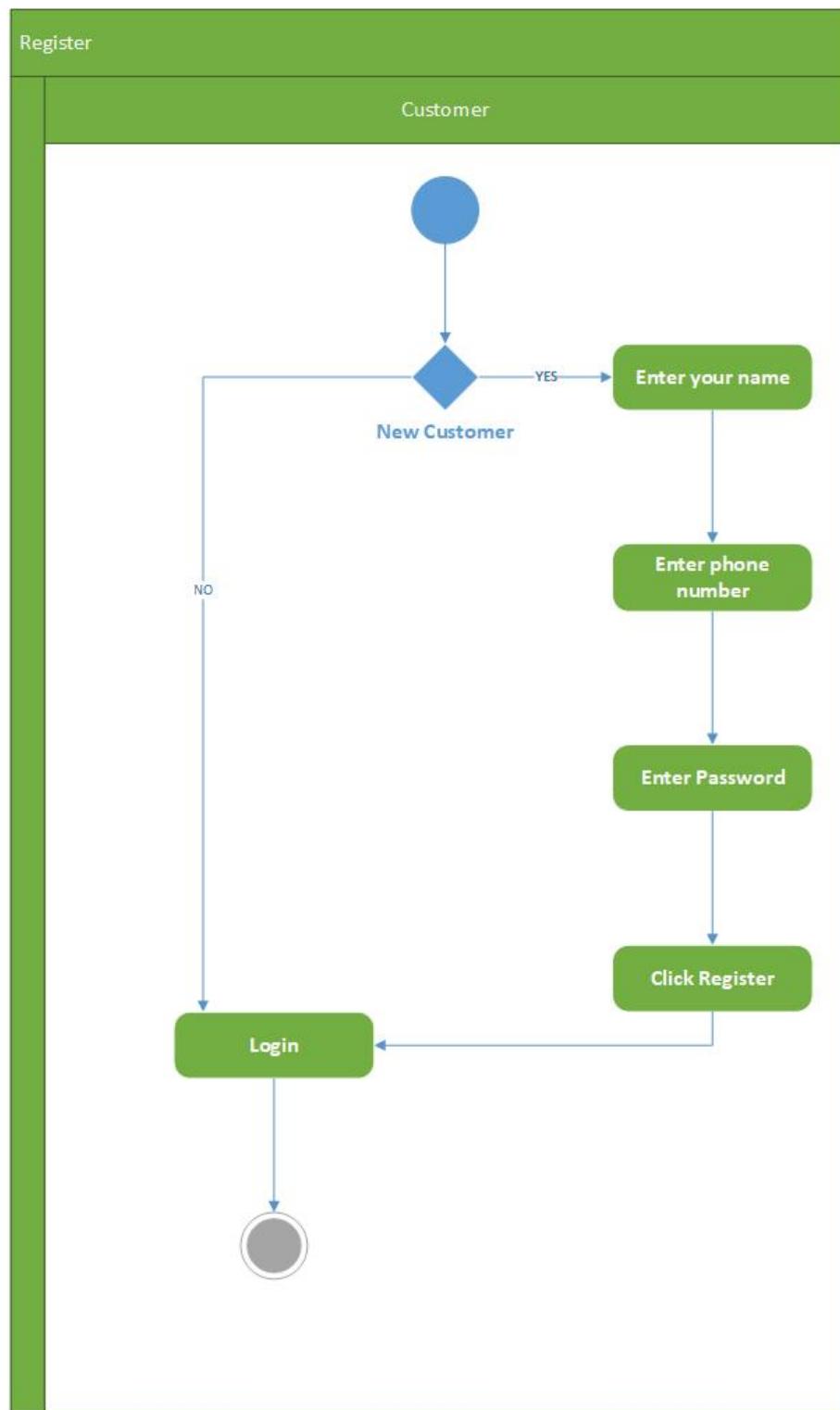


Figure 9: Register Activity Diagram

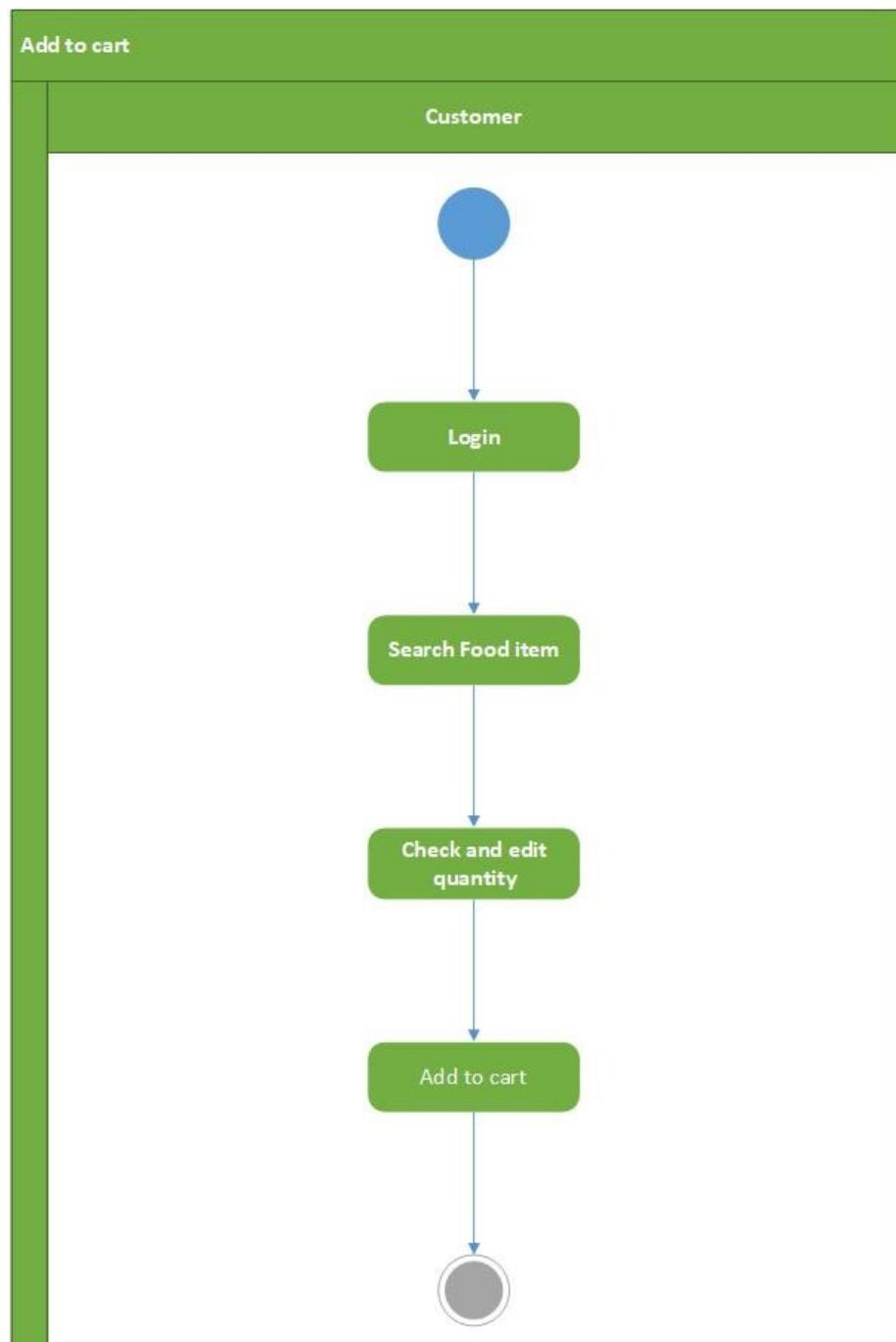


Figure 10: Add to Cart Activity Diagram

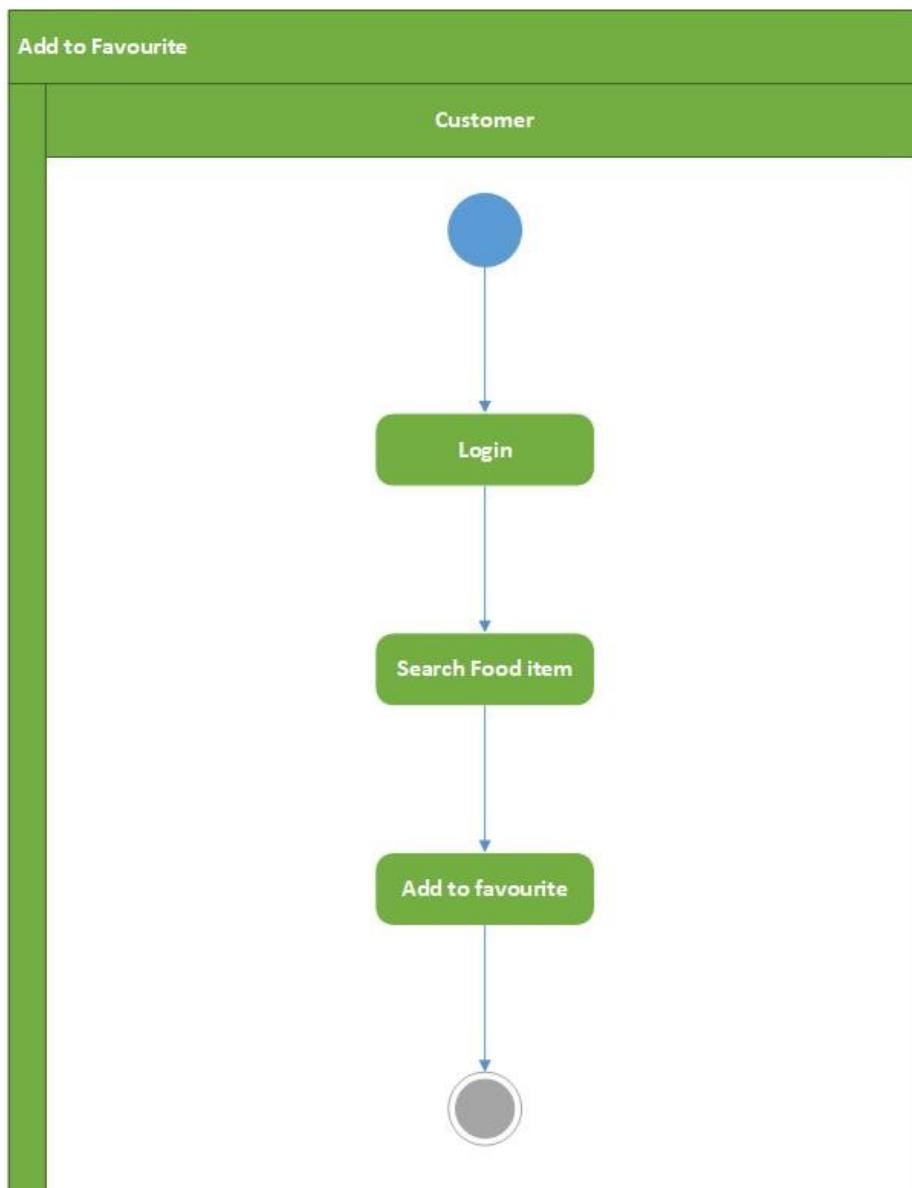


Figure 11: Add to favourite Activity Diagram

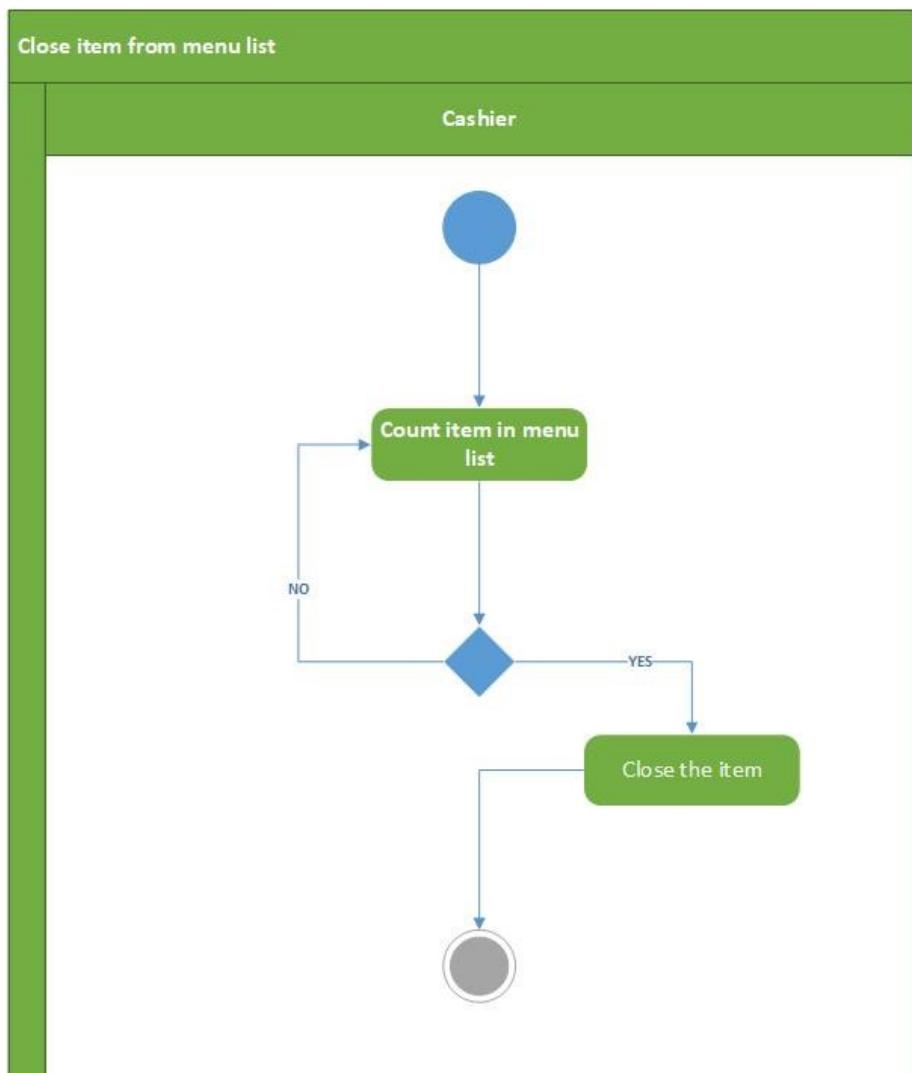


Figure 12: Close an item from menu Activity Diagram

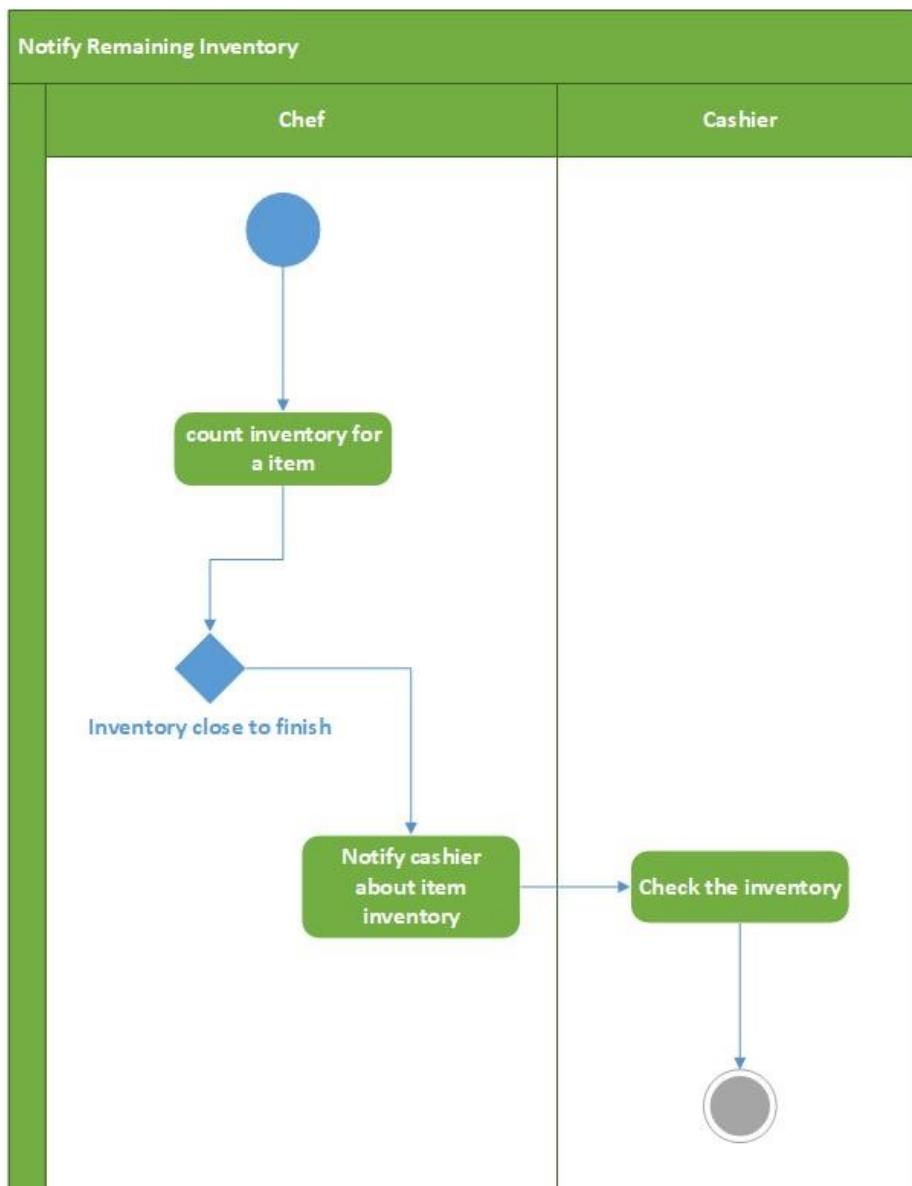


Figure 13: remaining Inventory Activity Diagram

## 4.3 Entity Relationship Model

### 4.3.1 Basic ER Diagram

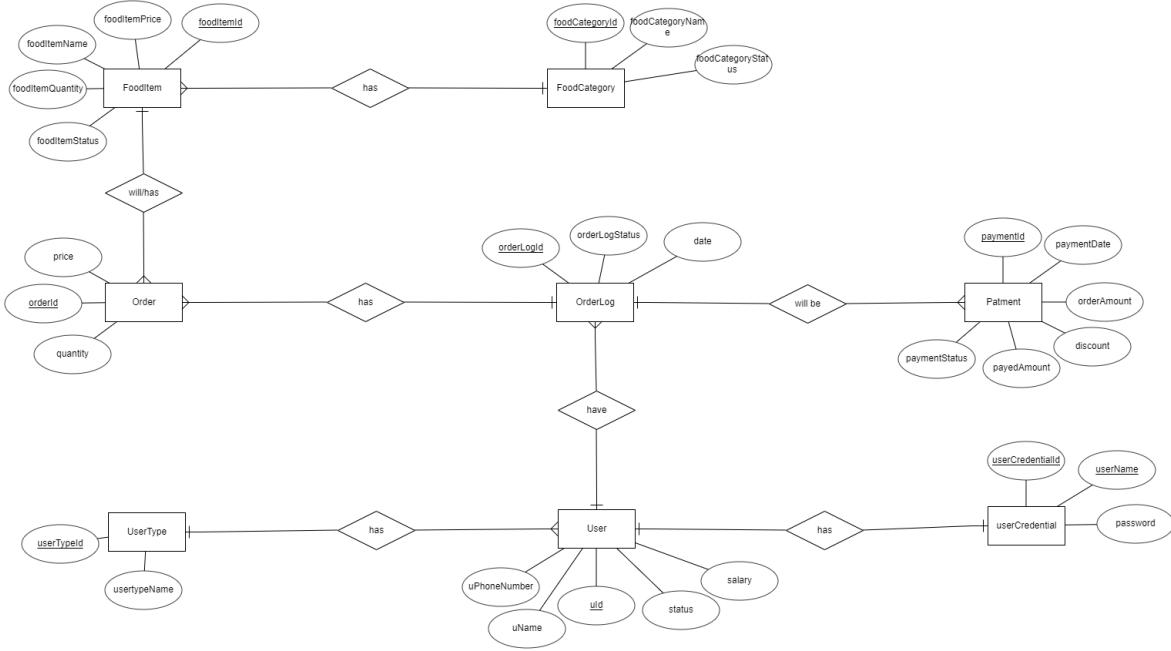


Figure 14: Basic ER Diagram

#### 4.3.2 Normalized ER Diagram

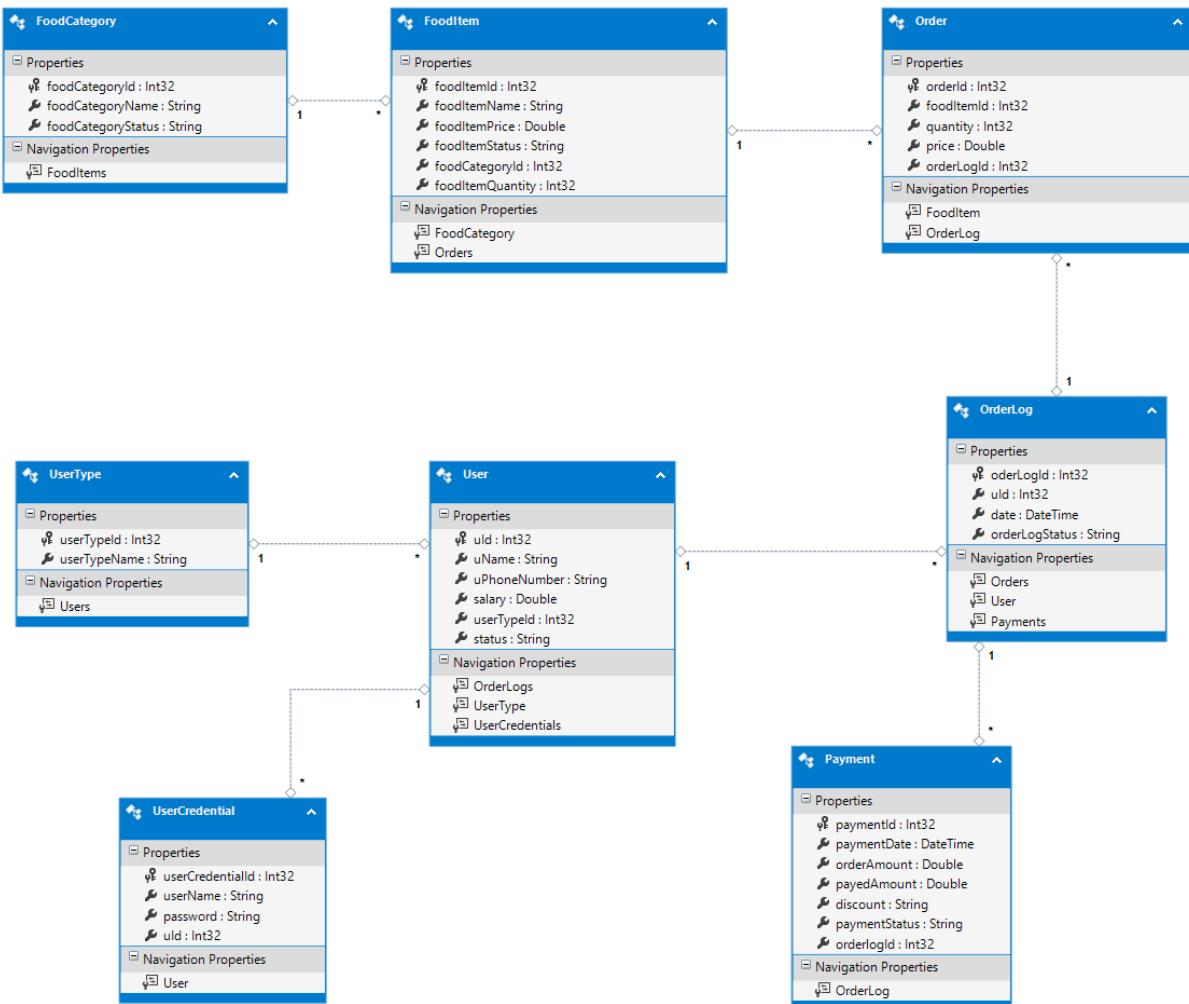


Figure 15: Normalized ER Diagram

#### 4.4 Data Dictionary

User Type					
Key	Field Name	Type	Size	Description	Example
pk	userTypeId	int		Primary Key	1
	userTypeName	string	100	Name of User Type	Chef

Table 4: User Type Data Dictionary

Food Category					
Key	Field Name	Type	Size	Description	Example
pk	foodCategoryId	int		Primary Key	1
	foodCategoryName	string	100	Name of the food category	Fast Food
	foodCategoryStatus	string	100	Category running or not	Valid

Table 5: Food Category Data Dictionary

Food Item					
Key	Field Name	Type	Size	Description	Example
pk	foodItemId	int		Primary Key	1
	foodItemName	string	100	Name of the food item	Chicken Burger
	foodItemPrice	double		Price of the food item	280.50
	foodItemStatus	string	15	Status of food item	Valid
fk	foodCategoryId	int		Foreign Key from table food category	1
	foodItemQuantity	int		Quantity of the remaining	10

Table 6: Food Item Data Dictionary

Order Data Dictionary					
Key	Field Name	Type	Size	Description	Example
pk	orderId	int		Primary Key	1
fk	foodItemId	int		Foreign Key from table food item	1
	quantity	int		Quantity of the item	5
	price	double		price of ordered item	125.80
fk	orderLogId	int		Foreign Key from table order log	1

Table 7: Order Data Dictionary

Order Log					
Key	Field Name	Type	Size	Description	Example
pk	orderLogId	int		Primary Key	1
fk	uId	int		Foreign Key from table user	1
	date	date/time		date of the order	10/03/2020
	orderlogStatus	string	100	status of the order	cooking

Table 8: Order Log Data Dictionary

User					
Key	Field Name	Type	Size	Description	Example
pk	uId	int		Primary Key	1
	uName	string	100	Name of the user	Niloy
unique	uPhoneNumber	string	14	Phone Number of user	+8801731569019
	salary	double	nullble	salary for the employee	30000
fk	userTypeId	int		Foreign Key from table user type	1
	status	string	50	status of the user	valid

Table 9: User Data Dictionary

User Credential					
Key	Field Name	Type	Size	Description	Example
pk	userCredentialId	int		Primary Key	1
fk	uId	int		Foreign Key from table user	1
unique	userName	string	50	unique username for users	niloy12345
	password	string	15	password for login	Abcd12345

Table 10: User Credential Data Dictionary

Payment					
Key	Field Name	Type	Size	Description	Example
pk	paymentId	int		Primary Key	1
fk	orderLogId	int		Foreign Key from table orderLog	1
	orderAmount	double		ordered price	250
	payedAmount	double		payed price	220
	discount	int		discount percentage	18%
	paymentStatus	string	50	payment status of the order	payed

Table 11: Payment Data Dictionary

## 5 Development

### 5.1 User Interface Design

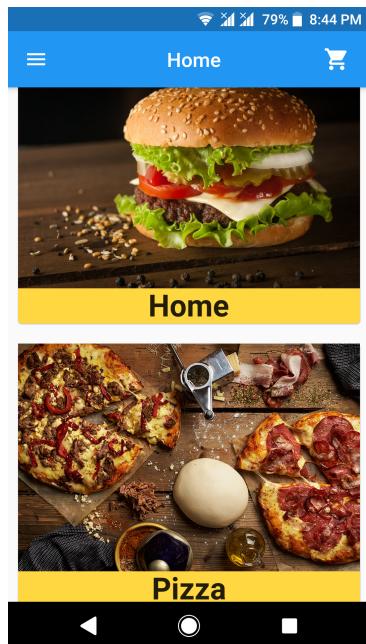


Figure 16: Menu View

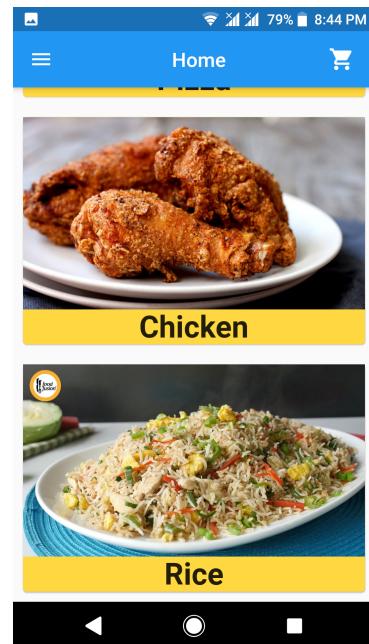


Figure 17: Menu View 2

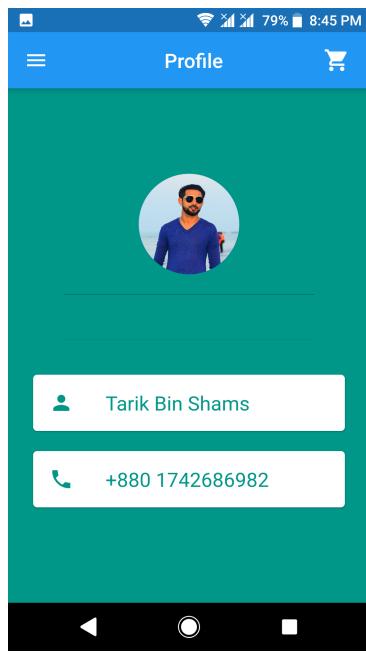


Figure 18: Profile View

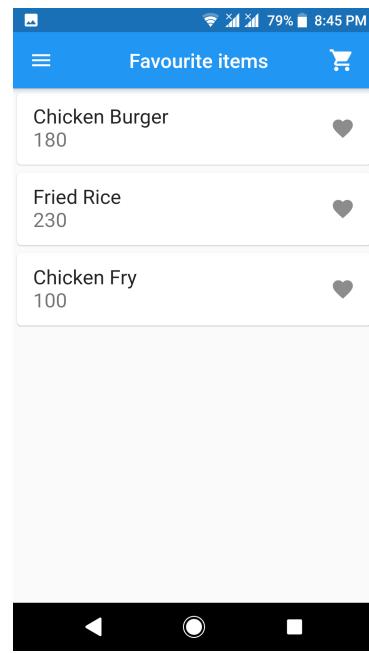


Figure 19: Favourites List View

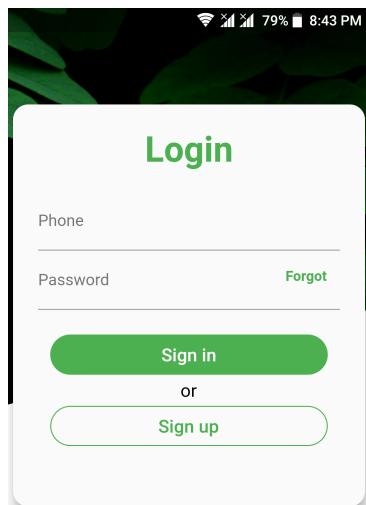


Figure 20: Login View

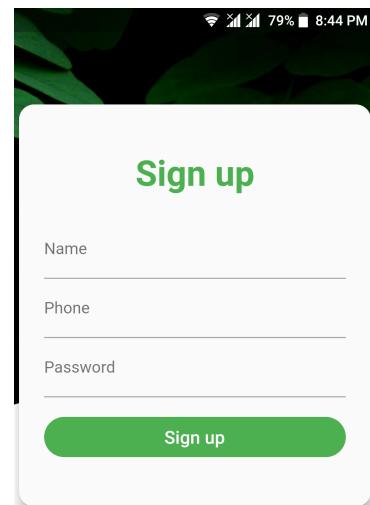


Figure 21: Signup View

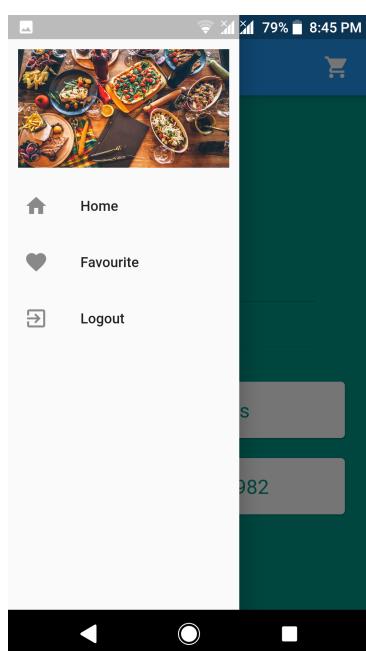


Figure 22: Dashboard View

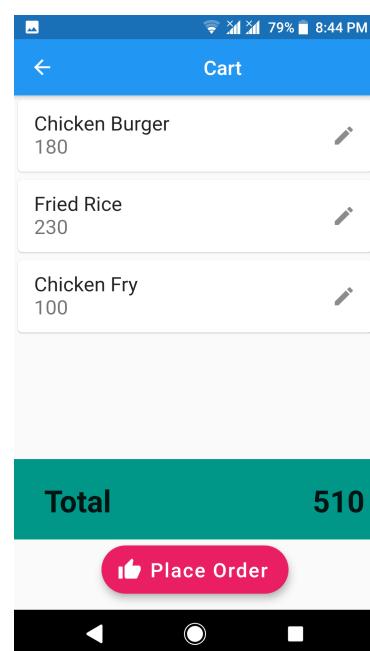


Figure 23: Cart View

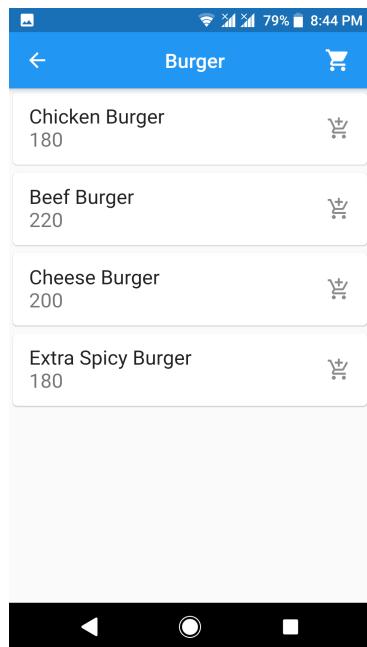


Figure 24: Burger List View

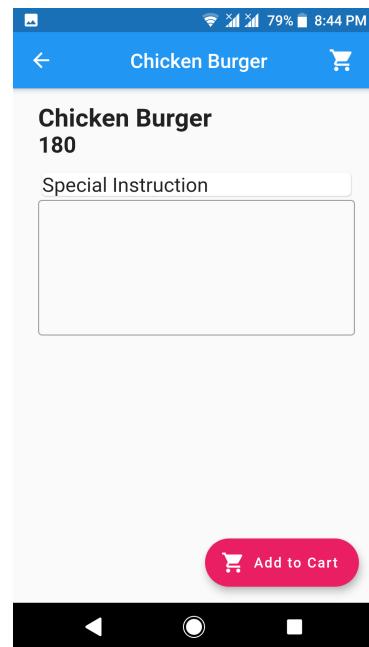


Figure 25: Add to Cart View

## 6 Test strategy

### 6.1 Introduction

A test strategy is an outline that describes the testing approach of the software development cycle. The purpose of testing is to make sure the functions work properly with user requirements. Testing strategies inform with development method, testing objectives, resources for testing, testing environment for test process to the project manager, QA team and developers.

#### 6.1.1 Module

In this system have some module for all types of user. We analyze functional requirement for module select all three types of user.

Serial No	Module Name	Applicable Roles
1	Register	Customer
2	Login	Cashier, Chef and Customer
3	Update profile	Cashier, Chef and Customer
4	Search item	Customer
5	Filter item	Customer
6	Favourite food list	Customer
7	Reorder item	Customer
8	Cart	Customer
9	Processing notification	Customer and Chef
10	Review of foods	Customer and Cashier
11	Approve order	Chef
12	Order note	Customer and Chef
13	Payment	Customer and Cashier
14	Remaining inventory	Cashier and Chef
15	Update inventory	Cashier
16	Close item	Cashier and Chef
17	Track orders	Cashier
18	Add offers	Cashier
19	Hire and Fire Chef	Cashier
20	Sign out	Cashier, Chef and Customer

Table 12: Module

### 6.2 Test plan

#### 6.2.1 Features to be Tested

There is the features list we test in the testing process.

##### 6.2.1.1 Customers Features

- FR\_1. The customers shall be able to register to the system with their phone number.

- FR\_2. The customers shall be able to login to the system.
- FR\_3. The customers shall be able to search food items.
- FR\_4. The customers shall be able to add food to cart.
- FR\_5. The customers shall be able to get food processing notifications.
- FR\_6. The customers shall be able to see review of foods.
- FR\_7. The customers shall be able to send feedback about the food.
- FR\_8. The customers shall be able to write order note.
- FR\_9. The customers shall be able to reset password.
- FR\_10. The customers shall be able to sign out from the system.
- FR\_11. The customers shall be able to re-ordered items which had been ordered previously.
- FR\_12. The customers shall be able to pay by online payment system.

#### 6.2.1.2 Chef Features

- FR\_13. The chefs shall be able to approve the orders.
- FR\_14. The chefs shall be able to change food processing status.
- FR\_15. The chefs shall be able to notify the customers when food cooking is finished.
- FR\_16. The chefs shall be able to notify the remaining inventory.
- FR\_17. The chefs shall be able to close items when inventory is finished.
- FR\_18. The chefs shall be able to see orders note.

#### 6.2.1.3 Cashier Features

- FR\_19. The cashier shall be able to track every orders.
- FR\_20. The cashier shall be able to add offers.
- FR\_21. The cashier shall be able to get notification of remaining inventory.
- FR\_22. The cashier shall be able to create profile of a new chef.
- FR\_23. The cashier shall be able to fire a chef.
- FR\_24. The cashier shall be able to update inventory.
- FR\_25. The cashier shall be able to close item when inventory is finished.

#### 6.2.2 Features not to be Tested

In testing process we can not test all features, functional requirement because complete testing is not feasible for any system. We can not complete our all testing features for testing environment, data, time, cost for the testing.

#### 6.2.2.1 Customer Features

- The customers shall be able to filter item with category.
- The customers shall be able to filter item with items prices.
- The customers shall be able to filter food offers.
- The customers shall be able to add favorite food items.
- The customers shall be able to check their favorite food items list

#### 6.2.2.2 Cashier Features

- The cashier shall be able to give discount.
- The cashier shall be able to see reviews of foods.

### 6.3 Test Methodology

Software testing methodology includes all types of testing for meet the user requirements. Testing method defined test objectives, test strategy and test deliverable. We used Agile methodology for testing in this system. Because Agile methodology focuses on responding to change rather than extensive planning. The main purpose of such process is to deliver new software features fast and with the best quality. Therefore, this approach is less cost-intensive: Fixing the errors early in the development process. In Agile methodology, software is developed in incremental, rapid cycles. Incremental testing is used in agile development methods. This ensures that any bugs in the system are fixed before the next release.

#### 6.3.1 Test Levels

We used four types of testing in our testing process.

- Unit Testing
- System Testing
- Regression Testing
- Acceptance Testing

### 6.3.2 Roles and Responsibilities

No.	Roles	Responsibilities
1.	Test Manager	Manage the whole project, Define project directions, Acquire appropriate resources
2.	Tester	Identifying and describing appropriate test techniques/tools/automation architecture, Verify and assess the Test Approach, Execute the tests, Log results, Report the defects.
3.	Developer in Test	Implement the test cases, test program, test suite etc.
4.	Test Administrator	Builds up and ensures test environment and assets are managed and maintained, Support Tester to use the test environment for test execution
5.	QA Members	Check to confirm whether the testing process is meeting specified requirements

Table 13: Roles and Responsibilities

## 6.4 Test Deliverables

Three phase should be complete during the testing process.

### 6.4.1 Before testing phase

- Test plans document.
- Test cases documents
- Test Design specifications.

### 6.4.2 During the testing

- Testing tool simulators
- Test data
- Logs-(Error and Execution)

### 6.4.3 After the testing phase

- Test case execution reports
- Defect reports
- Release notes

## 6.5 Test Cases

### 6.5.1 Customer Features

- FR\_01: The customers shall be able to register to the system with their phone number and password.

Project Name: Food Ordering Management System	Test Designed By: Tarik Bin Shmas			
Test Case ID: TP_FOMS_01	Test Designed date: Friday 13 <sup>th</sup> November, 2020			
Test Priority(Low, Medium, High): High	Test Executed by: Tarik Bin Shams			
Module Name: Register	Test Execution date: Friday 13 <sup>th</sup> November, 2020			
Test Title: Register with phone number and password				
Description: Test the system's registration page				
Precondition: Have a valid phone number and 4 length's password				
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)
<ul style="list-style-type: none"> <li>– Go to the Application</li> <li>– Enter phone number</li> <li>– Enter password</li> <li>– Click submit</li> </ul>	Phone number: 01811829982 Password: 2256	User should login with the system		
Post Condition: User has been successfully registered with the system.				

Table 14: FR\_01 Test Case

- FR\_02: The customers shall be able to login to the system.

Project Name: Food Ordering Management System		Test Designed By: Tarik Bin Shmas					
Test Case ID: TP_FOMS_02		Test Designed date: Friday 13 <sup>th</sup> November, 2020					
Test Priority(Low, Medium, High): High		Test Executed by: Tarik Bin Shams					
Module Name: Login		Test Execution date: Friday 13 <sup>th</sup> November, 2020					
Test Title: Login with phone number and password							
Description: Test the system's login page							
Precondition: Have a valid phone number and 4 length's password							
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)			
<ul style="list-style-type: none"> <li>– Go to the application</li> <li>– Enter phone number</li> <li>– Enter password</li> <li>– Click login</li> </ul>	Phone number: 01811829982 Password: 2256	User should log-in with the system					
Post Condition: User has been validated with database and successfully registered with the system.							

Table 15: FR\_02 Test Case

- FR\_03: The customers shall be able to search food items.

Project Name: Food Ordering Management System	Test Designed By: Tarik Bin Shmas					
Test Case ID: TP_FOMS_03	Test Designed date: Friday 13 <sup>th</sup> November, 2020					
Test Priority(Low, Medium, High): Medium	Test Executed by: Tarik Bin Shams					
Module Name: Search item	Test Execution date: Friday 13 <sup>th</sup> November, 2020					
Test Title: Searching food items from food list						
Description: Test the system's search page						
Precondition: Have food items in the list						
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)		
<ul style="list-style-type: none"> <li>- Go to the application</li> <li>- Enter food name on the Search box</li> </ul>	Search: Burger	Show all types of burger				
Post Condition: Food items have been searched with database and successfully showed on the search page.						

Table 16: FR\_03 Test Case

- FR\_04: The customers shall be able to add food to cart.

Project Name: Food Ordering Management System	Test Designed By: Tarik Bin Shmas			
Test Case ID: TP_FOMS_04	Test Designed date: Friday 13 <sup>th</sup> November, 2020			
Test Priority(Low, Medium, High): High	Test Executed by: Tarik Bin Shams			
Module Name: Cart	Test Execution date: Friday 13 <sup>th</sup> November, 2020			
Test Title: Add to cart				
Description: User can add food items to cart. If any item is in cart notify that already added				
Precondition: Have a food items list				
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)
<ul style="list-style-type: none"> <li>– Go to the application</li> <li>– Click add to cart in a food item</li> </ul>	Search: Clicked add to cart	User should add food cart items to cart		
Post Condition: User has been added food items to the cart.				

Table 17: FR\_04 Test Case

- FR\_05: The customers shall be able to get food processing notifications.

Project Name: Food Ordering Management System		Test Designed By: Tarik Bin Shmas			
Test Case ID: TP_FOMS_05		Test Designed date: Friday 13 <sup>th</sup> November, 2020			
Test Priority(Low, Medium, High): High		Test Executed by: Tarik Bin Shams			
Module Name: Processing notification		Test Execution date: Friday 13 <sup>th</sup> November, 2020			
Test Title: Getting order processing notification					
Description: User get notification of his/her order processing					
Precondition: Have a confirmed order					
Test Steps	Test Data	Expected Results	Actual results		
<ul style="list-style-type: none"> <li>– Go to the application</li> <li>– Confirmed an order with two food items.</li> </ul>	Food item id: 106, 110 have been ordered	User get the notification of his/her order processing			
Post Condition: User has been notified about order processing.					

Table 18: FR\_05 Test Case

- FR\_06: The customers shall be able to see review of the foods.

Project Name: Food Ordering Management System	Test Designed By: Alam Md. Mahbub					
Test Case ID: TP_FOMS_06	Test Designed date: Friday 13 <sup>th</sup> November, 2020					
Test Priority(Low, Medium, High): High	Test Executed by: Alam Md. Mahbub					
Module Name: Review of foods	Test Execution date: Friday 13 <sup>th</sup> November, 2020					
Test Title: The customer can see review of all foods						
Description: Test if customer click review button then seen the all previous customer review.						
Precondition: User must have a valid account.						
Test Steps	Test Data	Expected Results	Actual results			
<ul style="list-style-type: none"> <li>– Go to the application</li> <li>– Login to the customer account using valid user id and password.</li> <li>– Select any food item.</li> <li>– Click over the review button.</li> </ul>	Chicken Burger	User can see all previous review of that specific item				
Post Condition: Customer can order that item or skip that item .						

Table 19: FR\_06 Test Case

- FR\_07: The customers shall be able to send feedback about the food.

Project Name: Food Ordering Management System	Test Designed By: Alam Md. Mahbub			
Test Case ID: TP_FOMS_07	Test Designed date: Friday 13 <sup>th</sup> November, 2020			
Test Priority(Low, Medium, High): High	Test Executed by: Alam Md. Mahbub			
Module Name: Review of foods	Test Execution date: Friday 13 <sup>th</sup> November, 2020			
Test Title: Customer can feedback about the ordered food				
Description: Test if customer ordered any food.				
Precondition: Customer must confirm any order.				
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)
<ul style="list-style-type: none"> <li>– Click feedback button.</li> <li>– Type there feed-back.</li> </ul>	Text: This item was testy.9 out of 10.	Find a text field.		
Post Condition: Show a popup message and store this feedback on the database.				

Table 20: FR\_07 Test Case

- FR\_08: The customers shall be able to write order note.

Project Name: Food Ordering Management System	Test Designed By: Alam Md. Mahbub					
Test Case ID: TP_FOMS_08	Test Designed date: Friday 13 <sup>th</sup> November, 2020					
Test Priority(Low, Medium, High): High	Test Executed by: Alam Md. Mahbub					
Module Name: Order note	Test Execution date: Friday 13 <sup>th</sup> November, 2020					
Test Title: User Can write order note if they need any extra instruction.						
Description: Test if customer select any food item.						
Precondition: User must select any food item						
Test Steps	Test Data	Expected Results	Actual results			
<ul style="list-style-type: none"> <li>– Order any food item</li> <li>– Enter note if needed</li> <li>– Enter submit button</li> </ul>	Allergies	User can type note text box				
Post Condition: Customer can order that item or skip that item.						

Table 21: FR\_08 Test Case

- FR\_09: The customers shall be able to reset password.

Project Name: Food Ordering Management System		Test Designed By: Alam Md. Mahbub		
Test Case ID: TP_FOMS_09		Test Designed date: Friday 13 <sup>th</sup> November, 2020		
Test Priority(Low, Medium, High): Medium		Test Executed by: Alam Md. Mahbub		
Module Name: Update profile		Test Execution date: Friday 13 <sup>th</sup> November, 2020		
<p>Test Title: User can reset their password</p> <p>Description: Test if customer has valid account of the system.</p> <p>Precondition: Customer must login to their account.</p>				
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)
<ul style="list-style-type: none"> <li>– Click over the ‘My Profile’ button.</li> <li>– Click over the ‘Reset password’ button.</li> <li>– Enter new password on the text box</li> </ul>	Password	User find the home screen of the system.		
Post Condition: User has been validated with database with new password.				

Table 22: FR\_09 Test Case

- FR\_10: The customers shall be able to sign out from the system.

Project Name: Food Ordering Management System	Test Designed By: Alam Md. Mahbub			
Test Case ID: TP_FOMS_10	Test Designed date: Friday 13 <sup>th</sup> November, 2020			
Test Priority(Low, Medium, High): High	Test Executed by: Alam Md. Mahbub			
Module Name: Sign out	Test Execution date: Friday 13 <sup>th</sup> November, 2020			
Test Title: User can sign out his/her account				
Description: Test if customer has valid account.				
Precondition: User must login to their account.				
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)
<ul style="list-style-type: none"> <li>- Go to the application</li> <li>- Click over the Sign out button.</li> </ul>		User can find the home screen		
Post Condition: Customer can see the home screen of that system.				

Table 23: FR\_10 Test Case

- FR\_11: The customers shall be able to re-ordered items which had been ordered previously.

Project Name: Food Ordering Management System		Test Designed By: Mehedi Sayed					
Test Case ID: TP_FOMS_11		Test Designed date: Friday 13 <sup>th</sup> November, 2020					
Test Priority(Low, Medium, High): Medium		Test Executed by: Mehedi Sayed					
Module Name: Reorder item		Test Execution date: Friday 13 <sup>th</sup> November, 2020					
Test Title: Re-ordered items							
Description: User can order which had been ordered previously from this option							
Precondition: Completed a previous order							
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)			
<ul style="list-style-type: none"> <li>– Go to the application.</li> <li>– Clicked the previous order.</li> <li>– Show the order.</li> <li>– Place the order.</li> </ul>	Food item id: 106, 110 have been ordered	Order has been placed					
Post Condition: Order has been placed.							

Table 24: FR\_11 Test Case

- FR\_12: The customers shall be able to pay by online payment system.

Project Name: Food Ordering Management System		Test Designed By: Mehedi Sayed					
Test Case ID: TP_FOMS_12		Test Designed date: Friday 13 <sup>th</sup> November, 2020					
Test Priority(Low, Medium, High): High		Test Executed by: Mehedi Sayed					
Module Name: Payment		Test Execution date: Friday 13 <sup>th</sup> November, 2020					
Test Title: Online Payment							
Description: User can pay their bill by their online bank or mobile banking account.							
Precondition: Order added to cart.							
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)			
<ul style="list-style-type: none"> <li>– Go to the application.</li> <li>– Go to the cart.</li> <li>– Confirm Order.</li> <li>– Payment option selection.</li> </ul>	Food item id: 106, 110 have been ordered	payment will be done					
Post Condition: User's Payment will be successful							

Table 25: FR\_12 Test Case

### 6.5.2 Chef's Features

- FR\_13: The chefs shall be able to approve the orders.

Project Name: Food Ordering Management System		Test Designed By: Mehedi Sayed					
Test Case ID: TP_FOMS_13		Test Designed date: Friday 13 <sup>th</sup> November, 2020					
Test Priority(Low, Medium, High): Medium		Test Executed by: Mehedi Sayed					
Module Name: Approve order		Test Execution date: Friday 13 <sup>th</sup> November, 2020					
Test Title: Chef's approval for order.							
Description: Customer's placed order should be approve by chef							
Precondition: Customer should placed an order.							
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)			
<ul style="list-style-type: none"> <li>– Go to the application.</li> <li>– Go to placed order.</li> <li>– Confirm Order.</li> </ul>	Food item id: 106, 110 have been ordered	Order will start processing					
Post Condition: User will get notification of confirming order.							

Table 26: FR\_13 Test Case

- FR\_14: The chefs shall be able to change food processing status.

Project Name: Food Ordering Management System		Test Designed By: Mehedi Sayed			
Test Case ID: TP_FOMS_14		Test Designed date: Friday 13 <sup>th</sup> November, 2020			
Test Priority(Low, Medium, High): High		Test Executed by: Mehedi Sayed			
Module Name: Processing notification		Test Execution date: Friday 13 <sup>th</sup> November, 2020			
Test Title: Cooking status changed by chef.					
Description: Chef can update the cooking status after changing a state of cooking.					
Precondition: Customer should complete payment.					
Test Steps	Test Data	Expected Results	Actual results		
<ul style="list-style-type: none"> <li>– Go to the application.</li> <li>– Go to Confirmed order.</li> <li>– Change Status.</li> </ul>	Food item id: 106, 110 have been ordered	Cooking Status will shown			
Post Condition: User will be able to see the status of cooking.					

Table 27: FR\_14 Test Case

- FR\_15: The chefs shall be able to notify the customers when food cooking is finished.

Project Name: Food Ordering Management System		Test Designed By: Mehedi Sayed					
Test Case ID: TP_FOMS_15		Test Designed date: Friday 13 <sup>th</sup> November, 2020					
Test Priority(Low, Medium, High): High		Test Executed by: Mehedi Sayed					
Module Name: Processing notification		Test Execution date: Friday 13 <sup>th</sup> November, 2020					
Test Title: Notify the customer when cooking is finished.							
Description: After chef changed cooking status to complete the customer will get an notification.							
Precondition: Chef should start cooking.							
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)			
<ul style="list-style-type: none"> <li>– Go to the application.</li> <li>– Go to Ongoing order.</li> <li>– Change Status to complete.</li> </ul>	Food item id: 106, 110 have been ordered	Get notification of cooking finished					
Post Condition: User will get a notification of finished cooking.							

Table 28: FR\_15 Test Case

- FR\_16: The chefs shall be able to notify the remaining inventory.

Project Name: Food Ordering Management System	Test Designed By: Alam, Md. Mahbub					
Test Case ID: TP_FOMS_16	Test Designed date: Friday 13 <sup>th</sup> November, 2020					
Test Priority(Low, Medium, High): High	Test Executed by: Alam, Md. Mahbub					
Module Name: Remaining inventory	Test Execution date: Friday 13 <sup>th</sup> November, 2020					
Test Title: The chefs shall be able to notify the remaining inventory						
Description: Test if inventory is close to empty the chef will notify the cashier.						
Precondition: Inventory is close to empty.						
Test Steps	Test Data	Expected Results	Actual results			
<ul style="list-style-type: none"> <li>– Go to the application</li> <li>– Notify the cashier about the remaining inventory</li> </ul>	Food item: Chicken remain 4 out of 25	Chef can notify to the cashier with remaining inventory				
Post Condition: The chef notified the cashier with the remaining inventory.						

Table 29: FR\_16 Test Case

- FR\_17: The chefs shall be able to close items when inventory is finished.

Project Name: Food Ordering Management System	Test Designed By: Alam, Md. Mahbub							
Test Case ID: TP_FOMS_17	Test Designed date: Friday 13 <sup>th</sup> November, 2020							
Test Priority(Low, Medium, High): High	Test Executed by: Alam, Md. Mahbub							
Module Name: Close item	Test Execution date: Friday 13 <sup>th</sup> November, 2020							
Test Title: The chef shall be able to close the item when the inventory is finished								
Description: Test if the inventory will finished then the chef close the item from menu list								
Precondition: The specific item inventory should be finished								
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)				
<ul style="list-style-type: none"> <li>– Go to the application</li> <li>– Enter the quantity of chicken is to zero</li> </ul>	Chicken = 0 Food item :The chicken burger has closed	Chef can closed the finished item from menu list if inventory is finished.						
Post Condition: The chef has been able to close the item from menu list when the inventory is finished.								

Table 30: FR\_17 Test Case

- FR\_18: The chefs shall be able to see orders note.

Project Name: Food Ordering Management System	Test Designed By: Alam, Md. Mahbub					
Test Case ID: TP_FOMS_18	Test Designed date: Friday 13 <sup>th</sup> November, 2020					
Test Priority(Low, Medium, High): High	Test Executed by: Alam, Md. Mahbub					
Module Name: Order note	Test Execution date: Friday 13 <sup>th</sup> November, 2020					
Test Title: The chef shall be able to see orders note.						
Description: The chef see orders note with customer orders						
Precondition: After confirm order chef can see order note.						
Test Steps	Test Data	Expected Results	Actual results			
<ul style="list-style-type: none"> <li>- Go to the application</li> <li>- Check the order list for orders note</li> </ul>	Orders list: Table-1	The chef can see the orders note.				
Post Condition: The chef has been able to see the orders note from customer order list.						

Table 31: FR\_18 Test Case

### 6.5.3 Cashier Features

- FR\_19: The cashier shall be able to track every orders.

Project Name: Food Ordering Management System	Test Designed By: Alam, Md. Mahbub					
Test Case ID: TP_FOMS_19	Test Designed date: Friday 13 <sup>th</sup> November, 2020					
Test Priority(Low, Medium, High): Medium	Test Executed by: Alam, Md. Mahbub					
Module Name: Track orders	Test Execution date: Friday 13 <sup>th</sup> November, 2020					
Test Title: The cashier shall be able to track every orders						
Description: The cashier can track every orders from order list						
Precondition: The cashier able to track order after confirm the order.						
Test Steps	Test Data	Expected Results	Actual results			
<ul style="list-style-type: none"> <li>– Go to the application</li> <li>– Select any order from order list for track</li> </ul>	Orders list: Table-2	The cashier can track every orders				
Post Condition: The cashier has been track every orders from order list						

Table 32: FR\_19 Test Case

- FR\_20: The cashier shall be able to add offers

Project Name: Food Ordering Management System	Test Designed By: Alam, Md. Mahbub			
Test Case ID: TP_FOMS_20	Test Designed date: Friday 13 <sup>th</sup> November, 2020			
Test Priority(Low, Medium, High): Medium	Test Executed by: Alam, Md. Mahbub			
Module Name: Add offers	Test Execution date: Friday 13 <sup>th</sup> November, 2020			
Test Title: The cashier shall be able to add offers				
Description: The cashier add offer for customer menu list				
Precondition: The cashier add offer after certain times of confirm order or in a holiday or other time.				
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)
<ul style="list-style-type: none"> <li>– Go to the application</li> <li>– Select food items from item list</li> <li>– Add offers</li> </ul>	Food item: Chicken Cheese Burger Buy 1 Get 1 Free	The cashier can add offers		
Post Condition: The cashier has been able to add offers for customer menu				

Table 33: FR\_20 Test Case

- FR\_21: The cashier shall be able to get notification of remaining inventory.

Project Name: Food Ordering Management System		Test Designed By: Alam, Md. Mahbub					
Test Case ID: TP_FOMS_21		Test Designed date: Friday 13 <sup>th</sup> November, 2020					
Test Priority(Low, Medium, High): High		Test Executed by: Alam, Md. Mahbub					
Module Name: Remaining inventory		Test Execution date: Friday 13 <sup>th</sup> November, 2020					
Test Title: Get notification of remaining inventory							
Description: Test if inventory is empty then the cashier will get a notification.							
Precondition: Cashier must have a valid account.							
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)			
<ul style="list-style-type: none"> <li>- Go to the application</li> <li>- Login to the cashier account using valid user id and password.</li> <li>- Click over the notifications button.</li> </ul>		Cashier will get notifications of remaining product					
Post Condition: Cashier can brought the ingredients							

Table 34: FR\_21 Test Case

- FR\_22: The cashier shall be able to create profile of a new chef.

Project Name: Food Ordering Management System		Test Designed By: Alam, Md. Mahbub					
Test Case ID: TP_FOMS_22		Test Designed date: Friday 13 <sup>th</sup> November, 2020					
Test Priority(Low, Medium, High): High		Test Executed by: Alam, Md. Mahbub					
Module Name: Hire and Fire chef		Test Execution date: Friday 13 <sup>th</sup> November, 2020					
Test Title: Create profile for a new chef							
Description: Testing by creating a new chef account							
Precondition: chef must have a necessary information.(User name,Password)							
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)			
<ul style="list-style-type: none"> <li>– Click the create new account.</li> <li>– Enter username and password.</li> <li>– Click Submit.</li> </ul>	User Name: Chef01 Password: ACBD123	Account will be created successfully					
Post Condition: Chef should change password later.							

Table 35: FR\_22 Test Case

- FR\_23: The cashier shall be able to fire a chef.

Project Name: Food Ordering Management System		Test Designed By: Alam, Md. Mahbub					
Test Case ID: TP_FOMS_23		Test Designed date: Friday 13 <sup>th</sup> November, 2020					
Test Priority(Low, Medium, High): High		Test Executed by: Alam, Md. Mahbub					
Module Name: Hire and Fire chef		Test Execution date: Friday 13 <sup>th</sup> November, 2020					
Test Title: Firing Chefs							
Description: The cashier shall be able to fire a chef							
Precondition: Chefs must have a valid account							
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)			
<ul style="list-style-type: none"> <li>– Click Fire a chef.</li> <li>– Enter username of chef.</li> <li>– Click Submit.</li> </ul>	User Name: Chef01	Chef will be Fired successfully					
Post Condition: Cashier could hire the chef again							

Table 36: FR\_23 Test Case

- FR\_24: The cashier shall be able to update inventory.

Project Name: Food Ordering Management System		Test Designed By: Alam, Md. Mahbub					
Test Case ID: TP_FOMS_24		Test Designed date: Friday 13 <sup>th</sup> November, 2020					
Test Priority(Low, Medium, High): High		Test Executed by: Alam, Md. Mahbub					
Module Name: Update inventory		Test Execution date: Friday 13 <sup>th</sup> November, 2020					
Test Title: Test update inventory.							
Description: The cashier shall be able to update inventory.							
Precondition: Cashier must have a valid account							
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)			
<ul style="list-style-type: none"> <li>– Click Update Inventory</li> <li>– Enter quantity of ingredients.</li> <li>– Click Submit.</li> </ul>	Chicken : 10 Cheese : 50	Quantity will be updated					
Post Condition: Chef could see the updated quantity.							

Table 37: FR\_24 Test Case

- FR\_25: The cashier shall be able to close item when inventory is finished.

Project Name: Food Ordering Management System	Test Designed By: Alam, Md. Mahbub			
Test Case ID: TP_FOMS_25	Test Designed date: Friday 13 <sup>th</sup> November, 2020			
Test Priority(Low, Medium, High): High	Test Executed by: Alam, Md. Mahbub			
Module Name: Close Item	Test Execution date: Friday 13 <sup>th</sup> November, 2020			
Test Title: Closing items when inventory finished				
Description: The cashier shall be able to close items when inventory finished.				
Precondition: Cashier must have a valid account				
Test Steps	Test Data	Expected Results	Actual results	Status (Pass/Fail)
<ul style="list-style-type: none"> <li>– Click Close items</li> <li>– Enter items names.</li> <li>– Click Submit.</li> </ul>	Chicken Burger	Item will not show in menu		
Post Condition: Customer can not able to see the items in the menu.				

Table 38: FR\_25 Test Case

#### 6.5.4 Risk

Risk	Details	Mitigation
Skills	Team member lack the required skills for android testing.	Plan training course to skill up your members
Resources	The project schedule is too tight. It's hard to complete this project on time	Set Test Priority for each of the test activity.
Leadership	Test Manager has poor management skill	Plan leadership training for manager
Concentration	A lack of cooperation negatively affects your employees' productivity	Encourage each team member in his task, and inspire them to greater efforts.
Defects	Defects are found at a late stage of the cycle	Defect management plan is in place to ensure prompt communication and fixing of issues.

Table 39: Risks during testing

## 7 Deployment

Software deployment is the process of delivering completed software to the user and should ensure that all the flaws and bugs have been identified and fixed. We deploy our project after complete to develop every module. We will install 3 different APK file for 3 types user. For customer use we will install in 10 devices or android tablet. For cashier will install in his mobile with a account. Then he can hire chef as he pleased and provide them the software and user-name, password. After deployment software development life cycle is not finished. Maintenance is the final step of the life cycle, and this is when remaining fixes will be delivered.

## 8 Maintenance and Future Works

### 8.1 Maintenance

Software maintenance is the most expensive phase of software development life cycle. We maintain our project after deploy to the user with following:

- Performance monitoring.
- Bug fixes.
- Audits against with new version of android.
- Code updates for ensuring the compatibility with new version of android.

Adding new feature updates to our project based on user feedback and usage patterns helps retain them and keep them interested.

### 8.2 Future Works

This project is still basic stage there are more things to do in future. We want to implement kitchen live stream for customer, employee HR management system, food home delivery, advance customer table booking and also more things alongside solving some obvious bugs after maintenance.

## 9 User Manual

We deploy our 3 different project APK file in 3 different user android tablet. But user should know how to use our application. For Cashier, Open application with cashier android device and control everything alongside with create chef account. For Chef, Open application with Chef account information then see order list, approve order according to resource. For Customer, Open customer application and register with phone number if not register yet. After register get OTP from server then provide password for login. Then login with phone number and password. After login browse the application for order. Before leaving customer should sign out from the device. Otherwise after 30 min it will automatically sign out.

## 10 References

Those are the applications we used as references

- FoodPanda
- Uber Eats
- Hungry Naki
- Domino's Pizza
- FoodMart
- FoodBunny
- Swiggy
- Burger King
- Pizza Hut (Brunei)
- Pizza Hut (UAE)
- Pizza Hut (Singapore)
- MC Delivery
- KFC
- Glovo
- Jumia Foodt