



Hisham Fauzi Hashmi

(01-134152-027)

Attiqua Nazir

(01-134152-087)

## **DastarKhwan**

**Bachelor of Science in Computer Science.**

Supervisor: Sir Ali Irfan

Department of Computer Science  
Bahria University, Islamabad

May 2019

## Certificate

The work in the document “**Dastarkhwan**” is written by Hisham Fauzi Hashmi and Attiqa Nazir as a confirmation to the required standard for the partial fulfillment of the degree of Bachelor of Science in Information Technology.

**Approved by:**

Supervisor: Sir Ali Irfan (Assistant Professor)

---

Internal Examiner: Name of Internal Examiner(Title)

---

External Examiner: Name of External Examiner(Title)

---

Project Coordinator: Dr. Sumaira Kausar (Sr. Assistant Professor)

---

Head of Department: Dr. Muzammil (Associate Professor)

---

10<sup>th</sup> May 2019

# Abstract:

The world we live in today has become a global village, smartphones and web technology have become very useful and popular tools for communication between individuals. Handheld devices such as mobile phones, tablets and notebooks are being used in our daily lives for multiple purposes like entertainment, business etc. Majority of these handheld devices are compatible with Global Positioning System (GPS) which allows access to the real time geolocation of the device. Location Based Services (LBS) have become handy during the past few years and are going to play a very vital role in many applications in the future. The purpose of this project is to provide better and efficient dine in services to the people at their favorite restaurants in real time using web application. According to this Project Report, the application is fully capable of connecting customers to the restaurant management. It provides a platform on which you can make table reservation at your favorite place and at your desired time. The project reduces the struggle of both the customers and the restaurant management as it provides the dine in services provided by restaurants to the customers which they can use to make a reservation at any restaurant suitable to them.

# Acknowledgments

All praise is to Allah, the most Beneficent and the most Merciful. He bestowed upon us a drop of His boundless knowledge and by the virtue of it, we were able to accomplish this challenging task.

We are thankful to our project supervisor “Mr. Ali Irfan”. His personal supervision, advice and valuable guidance played a vital role in the completion of this project. We are grateful to him for his continual help and encouragement during this project. We are also deeply indebted to our parents and families as they gave us constant support and encouragement and brought us the values of hard work and honesty

Hisham Fauzi Hashmi,  
Attiqua Nazir

Islamabad, Pakistan

May 2019

# Content:

- INTRODUCTION ..... 1**
- 1.1 Overview: ..... 1**
- 1.2 Objective:..... 1**
- 1.3 Problem Description: ..... 1**
- 1.4 Methodology:.....2**
  - 1.4.1 Development Model:.....2
  - 1.4.2 User Interaction with the application: .....2
  - 1.4.3 System Architecture: .....7
- 1.5 Project Scope: ..... 8**
- 1.6 Feasibility Study: ..... 9**
  - 1.6.1 Resource Requirement:..... 9
- 1.7 Solution Application Area:..... 10**
- 1.8 Tools/Technology:..... 10**
- 1.9 Expertise of the Team Members: ..... 10**
- 1.10 Milestones: ..... 10**
  
- LITERATURE REVIEW.....11**
- 2.1 Literature review: .....11**
  - 2.1.1 MAQAMI: .....11
  - 2.1.2 foodpanda: ..... 12
  - 2.1.3 GloriaFood: ..... 12
- 2.2 Limitations or Drawbacks: ..... 12**
- 2.3 Problem Description: ..... 13**
  
- REQUIREMENT SPECIFICATIONS ..... 14**

<b>3.1 Application Overview:</b>	<b>14</b>
<b>3.2 General Description:</b>	<b>14</b>
3.2.1 Product Function:	14
3.2.2 User Characteristics:	15
<b>3.3 Proposed System:</b>	<b>15</b>
<b>3.4 Requirement Specification:</b>	<b>16</b>
3.4.1 Functional Requirements:	16
3.4.2 Non-Functional Requirements:	19
<b>3.5 Use Cases:</b>	<b>20</b>
3.5.1 Registration Process:	21
3.5.2 Login Process:	22
3.5.3 Table Reservation:	23
3.5.4 Dine-In Food Order:	24
3.5.5 Home Delivery:	25
3.5.6 Payment Procedure:	26
3.5.7 Edit Menu:	27
3.5.8 Feedback and Rating:	28
3.5.9 Manage Restaurants and Customers:	29
<b>SYSTEM DESIGN</b>	<b>30</b>
<b>4.1 System Architecture:</b>	<b>30</b>
<b>4.2 Sequence Diagram:</b>	<b>31</b>
4.2.1 Manage Accounts and Feedbacks:	31
4.2.2 Food Order:	32
4.2.3 Table Reservation:	32
4.2.4 Signup	33
4.2.5 Login	34
4.2.6 Logout:	34
<b>4.3 Entity Relation Diagram:</b>	<b>35</b>
4.3.1 Attributes:	36
<b>4.4 Class Diagram:</b>	<b>37</b>

<b>4.5 Tables:</b> .....	<b>38</b>
4.5.1 Customer:.....	38
4.5.2 Restaurant: .....	38
4.5.3 Admin:.....	38
4.5.4 Payment: .....	38
4.5.5 Account: .....	39
4.5.6 Delivery Order:.....	39
4.5.7 Dine-In Order: .....	40
4.5.8 Order Details: .....	40
4.5.9 Feedbacks .....	40
4.5.10 Table Reservation:.....	41
4.5.11 Discount Offers: .....	41
4.5.12 Menu: .....	41
4.5.13 Category:.....	42
4.5.14 Item:.....	42
<b>SYSTEM IMPLEMENTATION</b> .....	<b>43</b>
<b>5.1 System Architecture:</b> .....	<b>43</b>
<b>5.2 Tools Used:</b> .....	<b>43</b>
5.2.1 VS Code Text Editor: .....	43
5.2.2 Microsoft Visual Studio: .....	43
5.2.3 Microsoft SQL Server Management Studio: .....	43
<b>5.3 Languages Used: .....</b>	<b>44</b>
5.3.1 JavaScript:.....	44
5.3.2 Structured Query Language (SQL):.....	44
<b>5.4 Methodology: .....</b>	<b>44</b>
5.4.1 Customer Services: .....	44
5.4.2 Restaurant Services: .....	44
5.4.3 Admin Services:.....	45
<b>SYSTEM TESTING AND EVALUATION .....</b>	<b>46</b>
<b>6.1 Interface Testing: .....</b>	<b>46</b>
<b>6.2 Usability Testing: .....</b>	<b>46</b>

<b>6.3 Integration Testing:</b> .....	<b>46</b>
<b>6.4 Exception Handling:</b> .....	<b>46</b>
<b>6.5 Test Cases:</b> .....	<b>47</b>
6.5.1.1 Login:.....	47
6.5.1.2 Sign Up:.....	47
6.5.1.3 Home Page.....	48
6.5.2.1 Login Restaurant/Customer: .....	48
6.5.2.2 Sign Up Restaurant/Customer: .....	49
6.5.2.3 Dine IN Food Order Customer/Restaurant: .....	49
6.5.2.4 Home Delivery Order: .....	50
6.5.2.5 Table Reservation .....	51
6.5.2.6 Admin Panel: .....	52
6.5.2.7 Software Performance Test Case: .....	52
<b>6.6 Compatibility Testing:</b> .....	<b>53</b>
<b>CONCLUSION</b> .....	<b>54</b>
<b>7.1 Future Work:</b> .....	<b>54</b>
7.1.1 Google Maps:.....	54
7.1.2 Membership Payment Procedure:.....	54
<b>APPENDIX A</b> .....	<b>55</b>
<b>A.1 Customer Module Homepage:</b> .....	<b>55</b>
<b>A.2 Restaurant Search:</b> .....	<b>55</b>
<b>A.3 Book Your Table:</b> .....	<b>56</b>
<b>A.4 Menus:</b> .....	<b>56</b>
<b>A.5 Restaurant Module Login</b> .....	<b>57</b>
<b>A.6 Restaurant Homepage</b> .....	<b>57</b>
<b>REFERENCES</b> .....	<b>58</b>



# List of Figures:

- Figure 1.1: Software Development Lifecycle (SDLC)
- Figure 1.2: Customer Perspective's Working
- Figure 1.3: Restaurant Management's Perspective
- Figure 1.4: Application Working
- Figure 1.5: Project Milestones
- Figure 3.1: Use Case
- Figure 4.1: System Architecture
- Figure 4.2.1: Manage Accounts and Orders
- Figure 4.2.2: Food Order
- Figure 4.2.3: Table Reservation
- Figure 4.2.4: Login
- Figure 4.2.5: Logout
- Figure 4.3: Entity Relation Diagram
- Figure 4.4: Class Diagram
- Figure A.1 Customer Module Homepage
- Figure A.2 Restaurant Research
- Figure A.3 Book Your Table
- Figure A.4 Menu
- Figure A.5 Restaurant Module Login
- Figure A.6 Restaurant Homepage

# List of Tables:

Table 3.5.1	Registration Process
Table 3.5.2	Login Process
Table 3.5.3	Table Reservation
Table 3.5.4	Dine-In Food Order
Table 3.5.5	Home Delivery
Table 3.5.6	Payment Procedure
Table 3.5.7	Edit Menu
Table 3.5.8	Feedback and Rating
Table 3.5.9	Manage Restaurants and Customers
Table 4.3.1	Attributes
Table 4.5.1	Customer
Table 4.5.2	Restaurant
Table 4.5.3	Admin
Table 4.5.4	Payment
Table 4.5.5	Account
Table 4.5.6	Delivery Order
Table 4.5.7	Dine-In Order
Table 4.5.8	Order Details
Table 4.5.9	Feedback and Ratings
Table 4.5.10	Table Reservation
Table 4.5.11	Discount Offers
Table 4.5.12	Menu
Table 4.5.13	Category
Table 4.5.14	Item
Table 6.5.1.1	Login
Table 6.5.1.2	Sign Up
Table 6.5.1.3	Home Page
Table 6.5.2.1	Login Restaurant/Customer
Table 6.5.2.2	Sign Up Restaurant/Customer
Table 6.5.2.3	Dine IN Food Order Customer/Restaurant
Table 6.5.2.4	Home Delivery
Table 6.5.2.5	Table Reservation
Table 6.5.2.6	Admin Panel
Table 6.5.2.7	Software Performance Test Case
Table 6.6	Compatibility Test

# Acronyms and Abbreviations:

HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
AJAX	Asynchronous JavaScript and XML
SQL	Structured Query Language
HTTP	Hypertext Transfer Protocol
JS	JavaScript
ASP	Active Server Pages
UC	Use Case
DB	Database
ERD	Entity Relationship Diagram
GUI	Graphical User In

# Chapter 1:

## Introduction

### 1.1 Overview:

People in their daily lives, face a very unusual and discomforting situation, when they visit their favorite restaurants with their families on different occasions and don't find any table available for them to Dine IN which is a total waste of their time and money and it is also very disappointing for the people as well as the restaurant management to refuse their customers. People who get a table in time also have to face an annoying problem i.e. late serving of food. We often see people fighting over that with the restaurant management as their time is wasted especially during office working hours.

The main goal of Dastarkhwan is to provide a web application to the people which they will use to make table reservations at their favorite restaurants, in their desired timings. People can place an order for food as well through the application and will get a time duration for the order to be ready and will get a notification from the restaurant management that their order is prepared, and they can Dine IN any time in their allocated time. People can also order food for home delivery from the same web application and get it delivered at their homes.

### 1.2 Objective:

*“To develop a web application and a mobile application that will help the people to make table reservations, order food at the same time or get the food delivered at home via home delivery.”*

### 1.3 Problem Description:

On different occasions, we have seen people with their families standing at local restaurants on weekends or public holidays, waiting for an empty table. If they are lucky enough to find one in time, their order is not ready in time due to slow progress rate, which is annoying for the customers and an embarrassment for the restaurant management. Some people don't even find an empty table and have to leave in disappointment. We will provide a platform to the people that they will use to make table reservations for their families at their favorite restaurants from their homes using our web application. They can order food alongside the table reservation as well, so they don't have to wait for the food to be prepared. The restaurant management will

be notifying them when their meal is ready. They can also just order the food and get it delivered at home.

On the other hand, the restaurant management will be benefiting from this web application as well. Usually, they have a deal with all the customers at once but through this web application, they will know about the number of customers dining in and at what time. This web application will help them in managing their customers and serving them up to their expectations.

## 1.4 Methodology:

We will develop a web-based application using the platform of ASP.net MVC with the required HTML, CSS, BOOTSTRAP and JAVASCRIPT in order to achieve our desired goal.

### 1.4.1 Development Model:

The system which is to be developed is a web-based application, which is the reason why it needs to be updated after certain time periods to add new features to it, so it has to be made using the incremental model so that new features can be added to it at every increment.

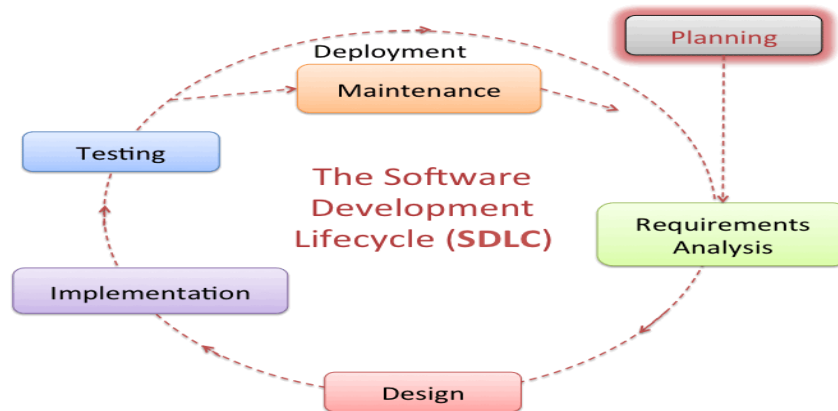


Figure 1.1: Software Development Lifecycle (SDLC)

### 1.4.2 User Interaction with the application:

Mainly, there are three types of interactions with our application.

1. Customer
2. Restaurant
3. Admin

### **1.4.2.1 Customer Perspective:**

Customers can utilize this system for making Dine In reservations or order food. There are different services which they can avail.

#### **Dine IN Reservations:**

Dine Reservations allow the customers to:

- Search for the Restaurants in their desired locations.
- For Dine IN, make Table Reservations for their families / friends.
- For Dine IN, customers can also place order online, so they don't have to wait for the food to be prepared.

#### **Home Delivery:**

- Place online orders for the food to be delivered at their homes.
- Check Order Status

#### **Feedback and Account Management:**

- Feedback on the services and the quality standards.
- Customers can manage their account details, view their order history and total engagements.

#### **Payment Options:**

The payment procedure for customers has been provided below:

##### **❖ Manual Payment Procedure:**

Customers can clear their billable after Dine IN or home delivery to the restaurant management / delivery guy.

#### **Customer Perspective's Working:**

- There will be a login or signup form. Customers will make an account before making a request.
- Provided restaurant or item name, customer can select any restaurant of his / her choice.
- Customer can book the table by selecting the no. of persons then the system will book the table/tables according to the persons and tables available at that time.
- Customer will be able to see the menu of restaurant according to its categories.

- Customers will be billed according to their order.

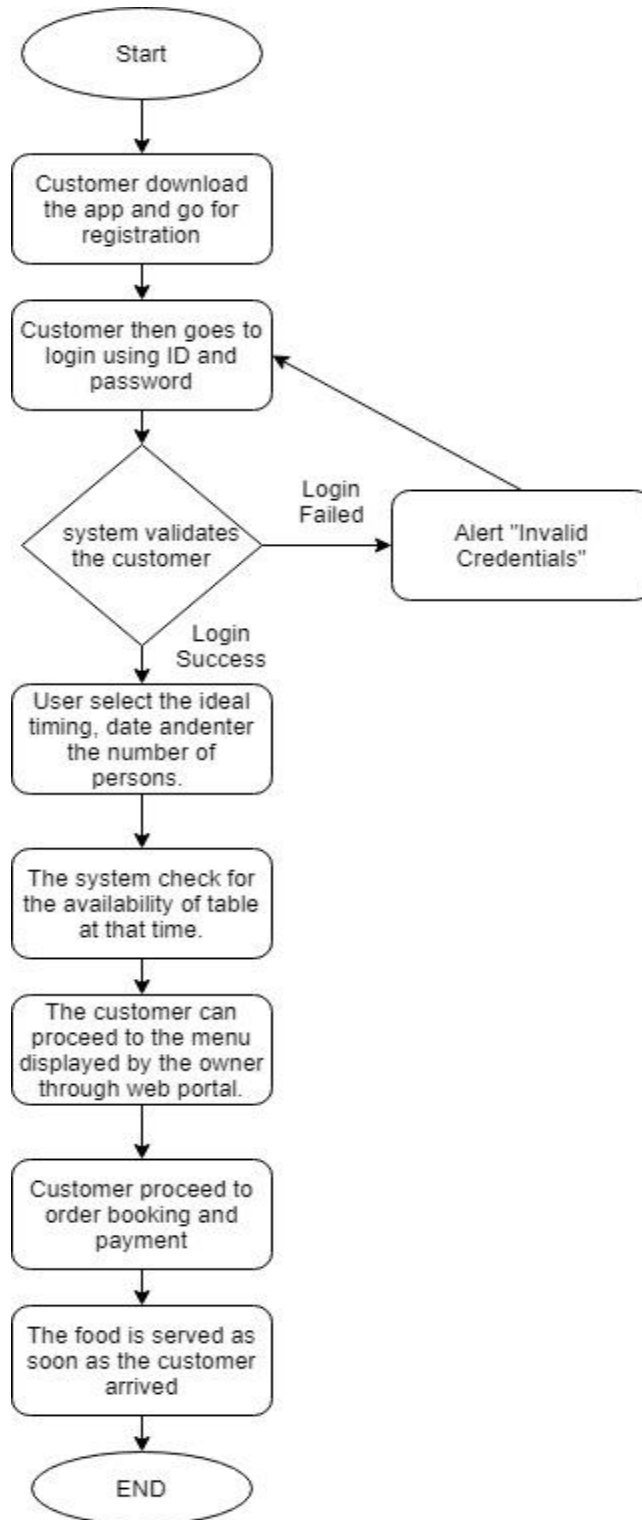


Figure 1.2: Customer Perspective's Working

### **1.4.2.2 Restaurant Management Perspective:**

Restaurant Management will be able to use the system and avail the following services.

- Get registered to the website and put their restaurant information about their restaurant.
- Put their menu information on the website and add their Dine IN details.
- Manage Dine IN reservations.
- Manage Online Food orders.
- Manage Home Delivery orders.
- Manage Promotional offers.
- Manage Payments.
- View Reservation details and customer engagements.

### **Restaurant Management Perspective's Working:**

- Restaurant should be able to register to the site giving location and menu of his restaurant.
- After logging in the restaurant will be able to view the table and order bookings placed by customers according to their arrival time.
- Management will be notified when an order is placed, and the food will be ready according to the arrival time of the customer.
- The management can edit the menu information such as price, items available currently and check if the order is served or not.
- They will also view the customer feedback.



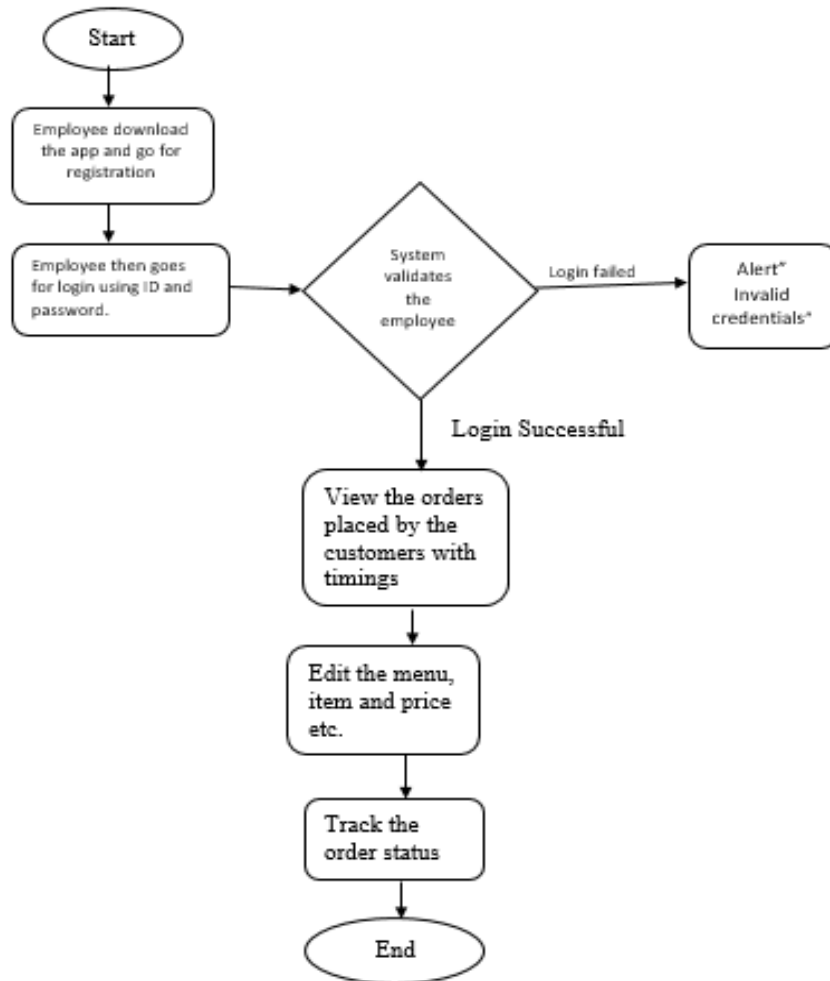


Figure 1.3: Restaurant Management Perspective's Working

### 1.4.2.3 Admin Perspective:

The admin module will be like a CMS that will allow the admin to:

- Manage Restaurants and Customers Information.
- Manage customer engagements and history.
- Manage customer feedback to improve the application.
- Block / Unblock restaurants and customers.

### Admin Perspective's Working:

The admin module will be like a CMS that will include:

- Admin will be able to manage the restaurants and customers details, who are engaging through this application.

- Admin will have the access to block any restaurant and customer for having bad conduct or bad faith.
- Admin will be working on the feedback provided by the customers and make improvements in the application for the customers to have a better experience.

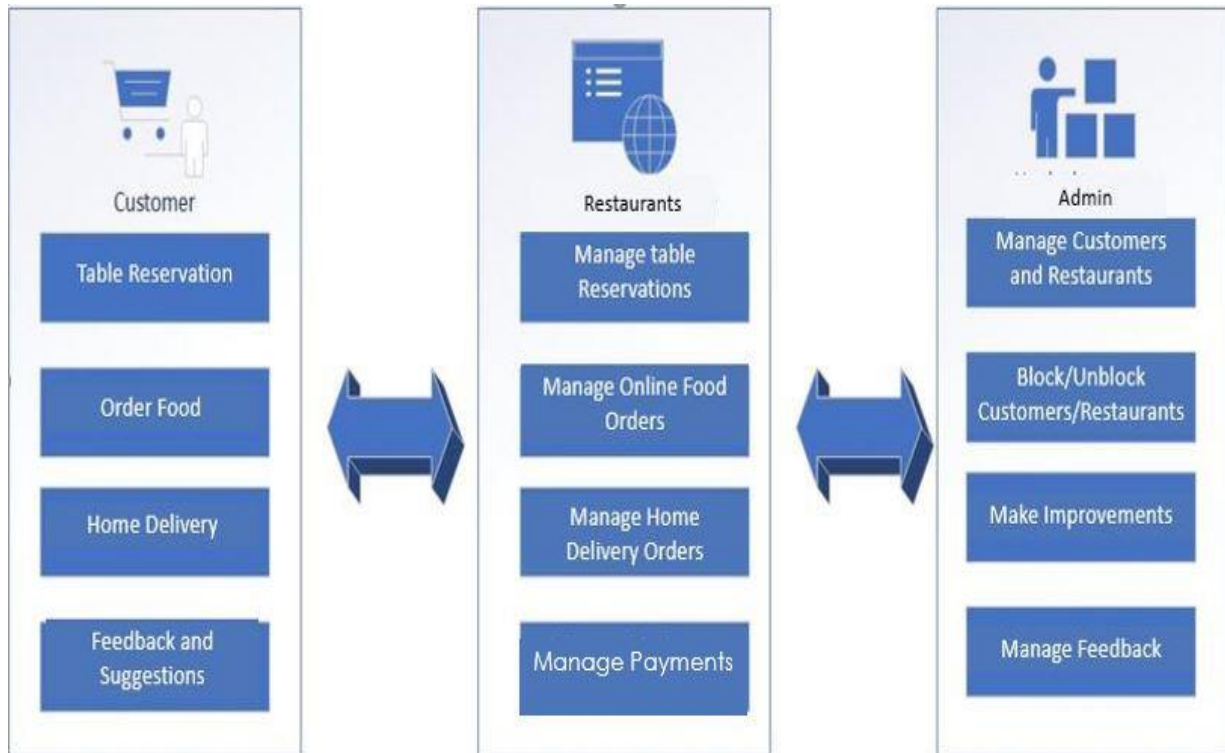


Figure 1.4: Application Working

### 1.4.3 System Architecture:

System Architecture includes:

- Presentation Layer
- Business Layer
- Database Layer

#### 1.4.3.1 Presentation Layer:

View Section includes following specifications:

- Sign in form.
- Registration form.
- “Notification”, notification section includes replies.
- Customer can provide their table and order booking details.
- Customer and restaurants profile.

### 1.4.3.2 Business Layer:

Business Layer contains the functionality of all modules of the system.

#### Customer:

- Takes username and password from Sign in form and checks validity through function and authenticate it.
- When customer will authenticate it then he/she will be able to make table reservation, place online order for serving or place order for home delivery.

#### Restaurant Management:

- Takes username and password from Sign in form and checks validity through function and authenticate it.
- When restaurant management will authenticate it then he/she will be able to check table reservation, online orders for serving or orders placed for home delivery.

#### Admin:

- An option will be provided to block/unblock restaurants and customers.
- An option will be provided to see the customer engagement history.
- Will be able to view to customer feedback.

### 1.4.3.3 Database Layer:

Model will be created in database. In tables we will add attributes and perform queries in it, which will respond in the above mentioned “business layer” section and can be viewed in the “view section”.

## 1.5 Project Scope:

The web application will comprise of all the features which will allow the customers to save time, energy and resources that they are going to use in finding empty tables at the restaurants by allowing them to book the table at the restaurant of their choice, order food for dine in or home delivery through this application.

Restaurant management will get the alert notification about the table reservation and the order placed, so they will inform their staff to get the table and the order prepared in time for the customers or for the delivery of food to their homes. The food will be served as soon as the customer arrives so there will be no wastage of time. This will also help the restaurant to increase their sales as there will be no shortage of tables

and speed of service will be much better. Chances of serving of wrong order will be reduced and all the customers will be well managed.

## **1.6 Feasibility Study:**

Yes, we will be able to complete all the proposed functional requirements in time.

### **1.6.1 Resource Requirement:**

It explains the resources (hardware and software) required for the development of application.

## **Application Development:**

For the development of this application we need a PC of the following hardware and software requirements.

### **❖ Hardware:**

- Personal Machine with minimum 8 GB Ram, 2GHz Core i5 Processors.
- Android mobile with Android 4.4 "KitKat" version.

### **❖ Software:**

- Windows Server/10
- IIS Web Server (ASP.NET)
- MS SQL Server
- Android Studio

## **User:**

For using our application and website, user requires the following resources:

- Android mobile with Android 4.4 "KitKat" version for android application.
- Internet Facility.
- Personal Machine with minimum 8 GB Ram, 2GHz Core i5 Processors for a web application.

## 1.7 Solution Application Area:

Our target are the youth, families and people from all walks of life as they all want to have a nice, stress free meal in their favorite restaurant without waiting for an empty table. This platform will be helping them by letting them make table reservation for them in their favorite restaurants, order food online either dine in or have it delivered to you at home, just by using this web application.

Restaurant management will also be benefiting from it as it will help them to serve their customers well and keep their house in order

## 1.8 Tools/Technology:

For developing this application, following tools are required:

- Visual studio 2015/2017
- MS SQL server
- IIS Web services
- Android Studio

## 1.9 Expertise of the Team Members:

We have studied the course of Web Engineering, in which we were taught web development using the platform of ASP.net and got a hands-on experience of making a project using the same platform. For this project, all the latest tools and techniques will be used. The Group member for this project are highly interested and have great desire to work in collaboration with each other.

## 1.10 Milestones:

Task name	Start	Finish
Project Proposal	09/10/2018	10/1/2018
Project Presentation	10/8/2018	10/12/2018
Customer Module Implementation	11/23/2018	02/28/2019
Restaurant Module Implementation	02/03/2018	03/29/2019
Admin Module Implementation	04/05/2019	4/29/2019
Final Project Report	05/01/2019	05/06/2019
Project Completion	05/07/2019	05/10/2019
Final Presentation	05/13/2019	05/15/2019

Figure 1.5: Project Milestone

# Chapter 2:

## Literature Review

### 2.1 Literature review:

Dastarkhwan can provide online opportunity for customers to place food orders and make table reservations on just simple clicks on Android and web-based application. The application will provide a single platform to the restaurants where they can simply signup and connect with their customers and also increase their sales as the system will reserve the tables according to the number of customers arriving and there will be no wastage of seats.

In this chapter, we have done some research on the similar existing systems and on how we can use their knowledge to develop a better online system. We have compared our work to them and we have proposed some improvements.

### 2.2 Existing System:

The existing systems related to our project are:

#### 2.1.1 MAQAMI:

MAQAMI is a web and mobile based application that tends to empower people and provide them with the opportunity to browse the best restaurants in their city from a single platform. It provides the information about the restaurants in your vicinity which are related to the food item you searched. You can see the menus and contact information of the restaurants which can be used to make a reservation at these restaurants. They also tend to provide information about the deals and promotional offers at different restaurants and allow the users to get the food delivered at their homes by using the contact information of the restaurants.

The goal of this application is to provide information about different restaurants in the city and provide the people with a centralized library of eateries which they can use to get dine in experience at different food places rather than only few places and that too within their budget.

### **2.1.2 foodpanda:**

foodpanda is a food delivery system that is based on mobile and web application. It is famous for its diversity in food items and a wide range of restaurants. It provides information about the restaurants in your close vicinity as well as in your city. You can order food from any eatery and get it delivered at your doorstep and all of it is done just by placing the order via mobile or web application of foodpanda. It also provides information about the promotional deals and discounts at different food places and different items.

foodpanda tends to provide food at your doorstep and save your time and resources by providing one platform from where all the restaurants in your city are accessible and you have the option to choose the restaurant and order food from that restaurant.

### **2.1.3 GloriaFood:**

GloriaFood is a web-based application that tends to provide free online food ordering system for restaurants. It allows the restaurants to sign up to their application and use their webservices to get themselves online and get their businesses up and running. It is beneficial for local food outlets as they have the opportunity to establish themselves and take orders online directly from the customers. They also allow the restaurant management to put online table reservation option at their page.

GloriaFood's sole purpose is to provide an opportunity to small businesses to flourish and make a name for themselves. By getting themselves online and connected to the world around them, small businesses have a chance to improve themselves and compete with the bigger food outlets in their vicinity.

## **2.2 Limitations or Drawbacks:**

The systems mentioned do not provide all the features required to make them a perfect platform such as MAQAMI only allows the users to search food items and on the basis of those food items, it displays the eateries, but their search system is not efficient and lacks accuracy as well. Also, it doesn't allow the users to make table reservation or order food delivery directly via their web portal or mobile app but instead, it provides contact information of the restaurants that can be used by the users to make table reservations and food delivery orders. foodpanda on the other hand is only for food delivery and GloriaFood is only a platform for small business to

get them online and running. It doesn't provide a wide range of restaurants from a single platform. Each restaurant uses their services separately.

So, there is a need of a system that provides all the extensive features that are needed to make a perfect platform for the people, our system will be providing an opportunity to the people to make table reservation, place order, get food delivered at their doorstep, track their order from one single platform without the expense of any other resource.

### **2.3 Problem Description:**

The major problems with all the previously made systems are that all of them are single purpose, they do not provide multiple purpose functionalities. MAQAMI provides only information about the restaurants i.e. their menus and contact information, foodpanda is only a food delivery system while GloriaFoods is only a platform for promoting local businesses and getting them online not for providing people a platform where they can access a library of restaurants and dine in or order food from the restaurant of their choosing. Our goal is to achieve a system that has the provides services to both the restaurants and the people. It will allow the restaurant to get online, promote and manage their businesses well, it will help the restaurant in management in maintaining better food service and quality as well as managing the number of people dining in at their restaurant at a particular time. Likewise, it will allow the people to access the restaurants of their choice and make a table reservation, place order or order food via home delivery. All that from one platform, they don't have to waste their time and other resources on finding a decent place for themselves and their families to get food from.



# Chapter 3:

## Requirement Specifications

This chapter will describe all the requirements for the development of admin, customer and restaurant module.

### 3.1 Application Overview:

An application is to be designed that constitutes of all the features that are required by the customers or restaurants. Our system ought to eliminate the manual procedures regarding table reservation and food delivery.

This system will appear to be a responsive web service to both the customers as they can make table reservation, order food online and get it delivered at their doorstep. Upon a certain action or query, a notification will be sent to the customers and the restaurant management.

System Admin will have the access to the web portal through which he could see all the activities of the customers and the restaurants and manage them. He will also be able to check the customer restaurant interactions to keep a check on the activities of people on this web portal.

### 3.2 General Description:

#### 3.2.1 Product Function:

This web application will be viewed with three different perspectives, from a Customer's Perspective, from a Restaurant's Perspective and from System Admin's Perspective.

##### 3.2.1.1 Customer's Perspective:

- There will be a login or signup form. Customers will make an account before making a request.
- Provided restaurant or item name, customer can select any restaurant of his / her choice.
- Customer can book the table by selecting the no. of persons then the system will book the table/tables according to the persons and tables available at that time.

- Customer will be able to see the menu of restaurant according to its categories.
- Customers will be billed according to their order.

### **3.2.1.2 Restaurant's Perspective:**

- Restaurant should be able to register to the site giving location and menu of his restaurant.
- After logging in the restaurant will be able to view the table and order bookings placed by customers according to their arrival time.
- Management will be notified when an order is placed, and the food will be ready when the customer will arrive.
- The management can edit the menu information such as price, items available currently and check if the order is served or not.
- They will also manage the customer feedback.

### **3.2.1.3 Admin's Perspective:**

The admin module will be like a CMS that will include:

- Admin will be able to manage the restaurants and customers details, who are engaging through this application.
- Admin will have the access to block any restaurant and customer for having bad conduct or bad faith.
- Admin will be working on the feedback provided by the customers and make improvements in the application for the customers to have a better experience.

### **3.2.2 User Characteristics:**

This web application does not require any specific or special characteristics from the users (Customers and Restaurants). Users are expected to be familiar with this web-based system. This web portal does not need expertise of portal and will be used by a technical individual. The system would provide a user-friendly interface which will not need any special knowledge.

### **3.3 Proposed System:**

The proposed system will eliminate the problem faced by the people when they visit different restaurants. People won't have to spend their resources in going to a restaurant a find a table for themselves as our application will be allowing them to reserve a table at their favorite restaurants online and feel free to go and dine in at their allocated time.

The system will be working as a web-based system and will be available to the customers and the restaurant management whenever they have registered themselves on this web application.

The system is supposed to work as an online system for the table reservation, online food order and home delivery that will also help in maintaining their record. The system mentioned can also work manually.

## **3.4 Requirement Specification:**

### **3.4.1 Functional Requirements:**

Functional Requirements is the explanation of all the accumulated requirements while keeping focus on all the stakeholders. Some of those requirements are mentioned below.

#### **I. Customer Sign Up:**

- **Description:**  
The attributes that will be required from the customers in order to get them registered will be First name, last name, username, email, id, password and confirm password.
- **Output:**  
Registration complete.
- **Processing:**  
User details are matched with the information in the database and password constraint's validity is checked as well.
- **Input:**  
User information.

#### **II. Customer Sign in:**

- **Description:**  
The application allows the users to sign in to the application and access its resources.
- **Output:**  
Profile page is displayed to the user.

- **Processing:**  
The system validates the login credentials entered by the user and compares them with the ones present in the database, if the credentials added are correct then the users can access the applications resources. Otherwise, they are asked to re-enter the correct login credentials.
- **Input:**  
Enter valid username and password.

### III. Select Restaurant:

- **Description:**  
The users have to enter the select the restaurant they want to dine in or order food from the list provided to them according to their search either by restaurant name or by item name.
- **Output:**  
Users are provided with the list of restaurants.
- **Processing:**  
The restaurant name or the item name added by the user will be checked with the details in the database and the restaurant or list of the restaurants will be displayed to the user.
- **Input:**  
Enter restaurant names or enter the area in which you want to check for restaurants.

### IV. Table Reservation and Place Order:

- **Description:**  
The users are provided with the option to make a table reservation for them at the restaurant of their choice. They will be allowed to place their food order at the same time along with the table reservation or place it on their arrival.
- **Output:**  
User will be provided the details of their reserved table and food order.
- **Processing:**  
The user will provide the information about the number of people for whom they want to reserve a table. The system will automatically reserve a table considering the total available seats, the best suitable table will be reserved for them so that other customers can be managed in the best suitable ways.

- **Input:**  
Enter the number of people, other personal information and food order.

## V. Restaurant Module:

- **Description:**  
The application allows the restaurants to manage their page and provide services to the user at the best of their abilities.
- **Output:**  
Restaurant Profile Page.
- **Processing:**  
The restaurants will be able to manage their page, edit their profile and menus, and add information about new deals. They will be able to check customer interactions, number of orders in process and seat reservation details.
- **Input:**  
Check Table Reservation details and food order details. Edit Profile and Menus.

## VI. Logout:

- **Description:**  
The application allows the users logout from their accounts after they are one using the application, securing it from other people.
- **Output:**  
The system logouts from the user account to the main page.
- **Processing:**  
User logouts from their accounts after they are done using the application.
- **Input:**  
Clicking the logout button.

### 3.4.2 Non-Functional Requirements:

Following are the Non-Functional Requirements for this system:

#### **i. Performance:**

The system's working should be in accordance to the provided use cases and systematic diagrams. The system will be sending notifications to the admin regarding the certain actions by the users such as making table reservations, food orders and order delivery details.

#### **ii. Operability:**

System must be user friendly, interactive and viable for the users.

#### **iii. Maintainability:**

The system should be capable of updating itself with new features.

#### **iv. Reliability:**

In case of some error or failure, system must be able catch and handle it without informing the users about it.

#### **v. Robustness:**

The system itself must be capable of handling system failures and get themselves recovered.

#### **vi. Security:**

The system must be capable of keeping the data secured.

### 3.5 Use Cases:

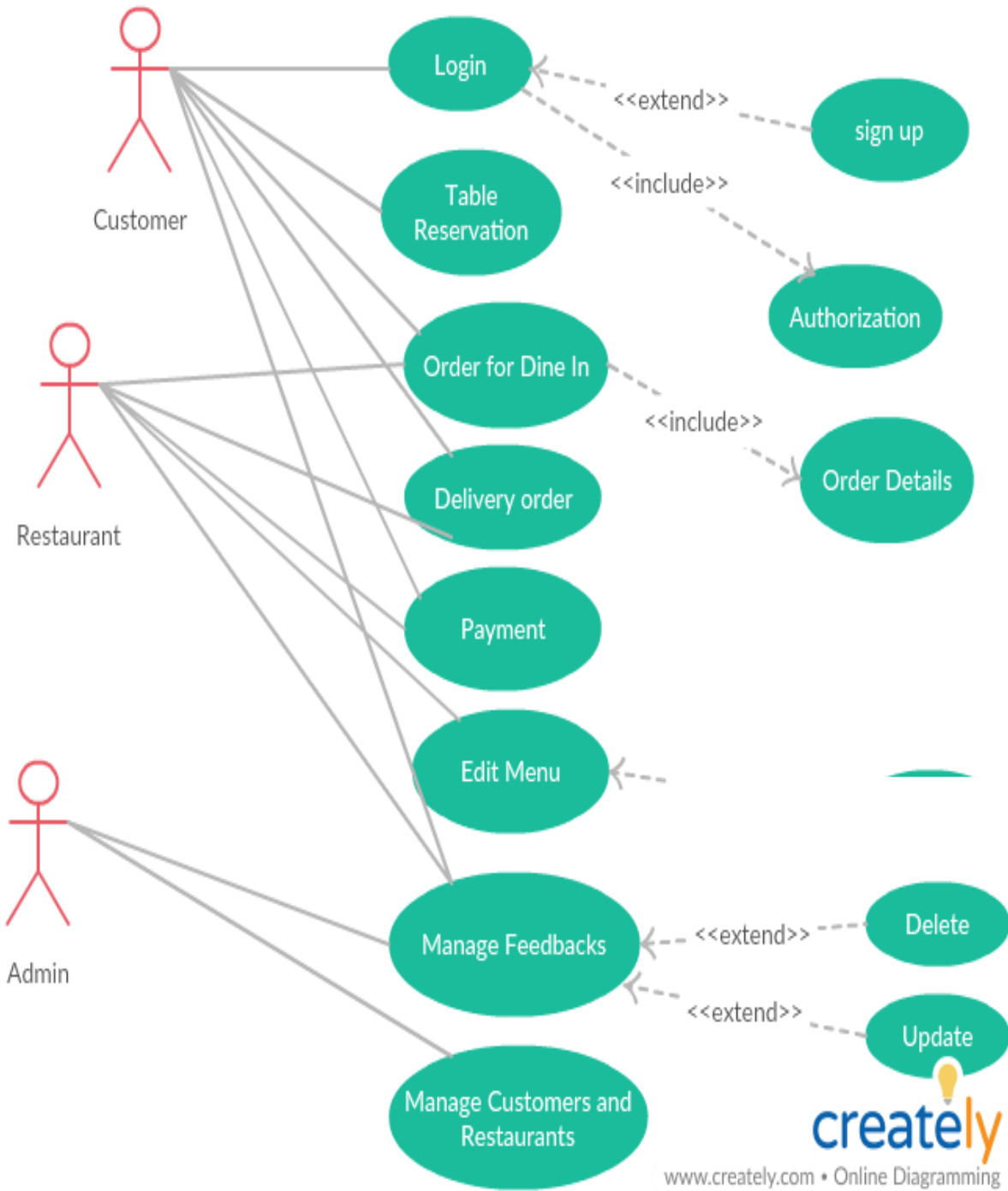


Figure 3.1: Use Case

### 3.5.1 Registration Process:

Table 3.5.1 Registration Process

<b>Use-Case Name:</b>	Registration process	
<b>Use-case ID:</b>	DK-BU01	
<b>Priority:</b>	High	
<b>Primary business Actor:</b>	Customer, Restaurant, Admin	
<b>Description:</b>	The use case describes the registration of any person or restaurant requesting to use our application. The member's demographic information as well as his or her account standing is validated. They will be allowed to use this application when they successfully register themselves	
<b>Basic Flow:</b>	<p>User opens to "Sign Up" page.</p> <ul style="list-style-type: none"> <li>• The user enters personal information and password to register in system. Restaurants also need to enter their location.</li> <li>• The system validates and sees if existing records are entered or not.</li> <li>• After the successfully entering the information, the user account information page is displayed and assign a unique ID to the user.</li> </ul>	
<b>Alternative flow:</b>		
<b>Precondition:</b>	No users are registered to our application.	
<b>Trigger:</b>	The use case is initiated when the user opens the signup page.	
<b>Course Event:</b>	<b>Action of actor</b>	<b>Response of System</b>
	Step 1: User fills the required fields in the signup form.	<p>Step 2: The system responds to all the information provided.</p> <p>Step 3: The system verifies all the information provided.</p> <p>Step 4: Availability of the previous record of the same information is checked and no duplication of records are checked.</p> <p>Step 5: If no such records are found the new account creates.</p>



		Step 6: Notification is sent to the user that his account is created.
<b>Post-condition:</b>	The account has been successfully created.	

### 3.5.2 Login Process:

Table 3.5.2 Login Process

<b>Use-Case Name:</b>	Login Process	
<b>Use-case ID:</b>	DK-BU02	
<b>Priority:</b>	High	
<b>Primary business Actor:</b>	Customer, Restaurant, Admin	
<b>Description:</b>	The use case explains login of any person requesting to use our application. The user will give information account required for the login to authorize from the system. They will be allowed to use this application when they login successfully.	
<b>Basic Flow:</b>	User opens to "Login" page. <ul style="list-style-type: none"> <li>• The user adds username and password in the previous user area of the "Login In".</li> <li>• The system verifies the password and enrollment and shows the account of user information page.</li> </ul>	
<b>Alternative flow:</b>	The user enters to the "Login In" page. <ul style="list-style-type: none"> <li>• The user adds enrollment and password.</li> <li>• The system figures out whether the enrollment/password is incorrect and asks the user to retry.</li> <li>•</li> </ul>	
<b>Precondition:</b>	No user login to our application.	
<b>Trigger:</b>	This use case is started whenever a current user tries to use this application.	
<b>Course Event:</b>	<b>Action of actor</b>	<b>Response of System</b>
	Step 1: User adds all the required data on the page.	Step 2: The system replies to all the data given. Step 3: This system verifies all the entered data. Step 4: Authorize user if exists. Step 5: If no such records are found

		the user is given an alert to retry. Step 6: If the authorization is successful then the user can now use the application.
<b>Post-condition:</b>	The user has been login successfully and now he/she can use the application.	

### 3.5.3 Table Reservation:

Table 3.5.3 Table Reservation

<b>Use-Case Name:</b>	Table Reservation Process
<b>Use-case ID:</b>	DK-BU03
<b>Priority:</b>	High
<b>Primary business Actor:</b>	Customer
<b>Description:</b>	The use case explains the table reservations made by customers. The customer will enter the number of persons and some personal information. The system will reserve the empty table according to the number of persons.
<b>Basic Flow:</b>	<p>User opens to "Table Reservation" page.</p> <ul style="list-style-type: none"> <li>• The user enters the number of persons, date, time of arrival and the checkout time</li> <li>• The system will look for the table or tables according to the number of persons and notify the table no to the customer.</li> <li>• The system will retrieve the personal information of the customer through the account and notify the restaurant with the reservation details.</li> <li>• The restaurant can see all the reservations with details.</li> </ul>
<b>Alternative flow:</b>	<p>The user enters to the "Table Reservation" page.</p> <ul style="list-style-type: none"> <li>• The user enters the number of persons, date, time of arrival and the checkout time</li> <li>• The system will look for the empty table or tables according to the number of persons and notify the table no to the customer.</li> <li>• If no table is available at the specified date or time, then the system asks the user to change the date or time.</li> </ul>

	<ul style="list-style-type: none"> <li>If the user wants to cancel the reservation. then click the cancel reservation button. Now system will set those tables as empty.</li> </ul>	
<b>Precondition:</b>	No user has made the table reservation.	
<b>Trigger:</b>	This use case is started whenever a current user tries to make table reservation	
<b>Course Event:</b>	<b>Action of actor</b>	<b>Response of System</b>
	Step 1: User adds all the required data on the page.	Step 2: The system replies back to all the data given. Step 3: This system notifies the table no to the customer. Step 4: The system notifies the restaurant with the reservation details.
<b>Post-condition:</b>	The table has been reserved successfully for the customer.	

### 3.5.4 Dine-In Food Order:

Table 3.5.4 Dine-In Food Order

<b>Use-Case Name:</b>	Dine-In Food Order Process
<b>Use-case ID:</b>	DK-BU04
<b>Priority:</b>	High
<b>Primary business Actor:</b>	Customer, Restaurant
<b>Description:</b>	The use case explains the dine-in food order placed by customers. The customer will search the desired restaurant, navigate the menu and then place the order by selecting the item and entering the order details with the arrival time. The application will notify the order details to the restaurant.
<b>Basic Flow:</b>	Customer clicks on the item <ul style="list-style-type: none"> <li>The item details are displayed on the page.</li> <li>The customer can select the quantity and amount of the item and also the different options for order details like flavor etc.</li> <li>Customer clicks on the Dine-In order option.</li> <li>Customer will enter the date and arrival time.</li> <li>Restaurant will receive the order details.</li> </ul>

	<ul style="list-style-type: none"> <li>• Food will be served when the customer will arrive.</li> </ul>	
<b>Precondition:</b>	No customer has made Dine-In order	
<b>Trigger:</b>	This use case is started whenever a current user tries to place Dine-In food order.	
<b>Course Event:</b>	<b>Action of actor</b>	<b>Response of System</b>
	<p>Step 1: Customer select the desired food item and enter the details.</p> <p>Step 3: Customer select the dine-in order option.</p> <p>Step 5: Customer will provide the required information.</p>	<p>Step 2: The system will display the payment details and food order details.</p> <p>Step 4: The system will ask for the date and time of arrival.</p> <p>Step 6: Order will be shown to the restaurant with all the details.</p>
<b>Post-condition:</b>	The Dine-In food order has been placed successfully.	

### 3.5.5 Home Delivery:

Table 3.5.5 Home Delivery

<b>Use-Case Name:</b>	Home Delivery Order Process
<b>Use-case ID:</b>	TA-BU05
<b>Priority:</b>	High
<b>Primary business Actor:</b>	Customer, Restaurant
<b>Description:</b>	The use case explains the home delivery order placed by customers and received by restaurant. The customer will search the desired restaurant, navigate the menu and then place the order by selecting the item and entering the order details. The system will notify the order details to the restaurant and arrival time to the customer.
<b>Basic Flow:</b>	<p>Customer clicks on the item</p> <ul style="list-style-type: none"> <li>• The item details are displayed on the page.</li> <li>• The customer can select the quantity and amount of the item and also the different options for order details like flavor etc.</li> <li>• Customer clicks on the home delivery order option.</li> <li>• Customer will enter the address and some personal details.</li> </ul>

	<ul style="list-style-type: none"> <li>Home delivery order will be shown to the restaurant with all the details.</li> </ul>	
<b>Precondition:</b>	No customer has placed Home delivery order.	
<b>Trigger:</b>	This use case is started whenever a current user tries to food delivery order	
<b>Course Event:</b>	<b>Action of actor</b>	<b>Response of System</b>
	Step 1: Customer select the desired food item with the details. Step 3: Customer select the home delivery food order option.	Step 2: The system will display the payment details with delivery fee and food order details. Step 4: The system will ask for the address, name and contact no etc. Step 5: the system will notify the restaurant with the order details.
<b>Post-condition:</b>	The Home delivery food order has been placed successfully.	

### 3.5.6 Payment Procedure:

Table 3.5.6 Payment Procedure

<b>Use-Case Name:</b>	Payment Procedure	
<b>Use-case ID:</b>	DK-BU06	
<b>Priority:</b>	High	
<b>Primary business Actor:</b>	Customer, Restaurant	
<b>Description:</b>	The use case explains the payment procedure for dine-in or delivery order. The customer can pay with cash.	
<b>Basic Flow:</b>	<b>Manual Payment Procedures:</b> <ul style="list-style-type: none"> <li>Customers can clear their bills after Dine IN or home delivery to the restaurant management / delivery guy.</li> </ul>	
<b>Precondition:</b>	Customer has not paid yet.	
<b>Trigger:</b>	This use case is started whenever a customer wants to pay the bill.	
<b>Course Event:</b>	<b>Action of actor</b>	<b>Response of System</b>

	<p>Step 1: User clicks on “Payment page”.</p> <p>Step 3: User clicks on ‘Make Payments’ link in the Screen.</p> <p>Step 4: User enters the following information.</p> <p>Applicant Information:</p> <ul style="list-style-type: none"> <li>• First Name</li> <li>• Last Name</li> <li>• Mode of Payment</li> </ul> <p>i.e. cash on delivery</p>	<p>Step 2: The system will display the payment details.</p> <p>Step 4: The system will notify the restaurant about the billing amount.</p>
<b>Post-condition:</b>	The payment has been cleared successfully.	

### 3.5.7 Edit Menu:

Table 3.5.7 Edit Menu

<b>Use-Case Name:</b>	Edit Menu Process	
<b>Use-case ID:</b>	DK-BU07	
<b>Priority:</b>	High	
<b>Primary business Actor:</b>	Restaurant	
<b>Description:</b>	The use case explains the process of adding or removing food items from the menu. When the restaurant will edit its menu, the changes will be shown in our application automatically	
<b>Basic Flow:</b>	Restaurant click opens the edit menu page. Categories with items details and prices are shown on the page. When the restaurant is registered with the application the menu details will also be provided. Now the restaurant can add, remove or update the category or food item through the application.	
<b>Precondition:</b>	Restaurant hasn't made any changes in the menu.	
<b>Trigger:</b>	This use case is started whenever a restaurant wants to edit the menu.	
<b>Course Event:</b>	<b>Action of actor</b>	<b>Response of System</b>
	<p>Step 1: Staff member click open the edit menu page.</p> <p>Step 3: Restaurant clicks on the add or remove category or item option.</p> <p>Step 4: Restaurant will enter all the details or choose the category or item.</p>	<p>Step 2: System will show all categories with items details and prices on the page.</p> <p>Step 4: System will ask the restaurant to enter the item or</p>

	Step 6: Restaurant will enter the changes for updating the existing details of category or foods item.	category details to be added and also the menu to which it belongs (if there are more than one) or to choose the item or category to remove. Step 5: The system will add or remove the item for the menu. Step 7: The system will make the changes in the menu accordingly.
<b>Post-condition:</b>	Menu has been updated successfully.	

### 3.5.8 Feedback and Rating:

Table 3.5.8 Feedback and Rating

<b>Use-Case Name:</b>	Feedbacks Process	
<b>Use-case ID:</b>	DK-BU08	
<b>Priority:</b>	High	
<b>Primary business Actor:</b>	Restaurant, Customer, Admin	
<b>Description:</b>	The use case explains the feedbacks and ratings given by the customers and checked by restaurant and admin. Customer gives the feedback and rating based on their experience with application or restaurant. Restaurant can improve their performance and admin can improve the application by reading the customer views.	
<b>Basic Flow:</b>	Whenever the customer place order, the application will ask to give the feedback for the restaurant and the application. Customer can skip this process. Restaurants can give feedback about the customer and the application or check their feedbacks. Admin can also check the feedbacks.	
<b>Precondition:</b>	No customer has given feedback and rating.	
<b>Trigger:</b>	This use case is started whenever a customer gives rating or feedback.	
<b>Course Event:</b>	<b>Action of actor</b>	<b>Response of System</b>

	<p>Step 1: Customer place the order.</p> <p>Step 3: Customer gives rating or feedback.</p> <p>Step 4: Restaurant gives the feedback about customer or the application</p>	<p>Step 2: System will ask for rating and feedback.</p> <p>Step 5: If the feedback is for the application then the system will show it to the admin only otherwise system it will be shown to both restaurant and admin.</p>
<b>Post-condition:</b>	Feedback has been given by customer and checked by restaurant and admin.	

### 3.5.9 Manage Restaurants and Customers:

Table 3.5.9 Manage Restaurants and Customers

<b>Use-Case Name:</b>	Manage Restaurants and Customers Process	
<b>Use-case ID:</b>	DK-BU08	
<b>Priority:</b>	High	
<b>Primary business Actor:</b>	Admin	
<b>Description:</b>	Based on the feedbacks of customers or restaurants admin make improvement in the application. This use case will basically act as a user care service.	
<b>Basic Flow:</b>	Admin will check the feedbacks given by customers. If there are any complaints, then the admin will verify those complaints and take measures accordingly. Admin can make improvement in application.	
<b>Precondition:</b>	No complaints have been made.	
<b>Trigger:</b>	This use case is started whenever admin wants to resolve any issue.	
<b>Course Event:</b>	<b>Action of actor</b>	<b>Response of System</b>
	<p>Step 1: Admin opens the “feedback” page.</p> <p>Step 3: If there are any complaints then admin will firstly verify the complaint.</p> <p>Step 4: If the complaint is valid then the admin will take measures accordingly.</p>	<p>Step 2: System will show the feedbacks given by the customers.</p>
<b>Post-condition:</b>	Problem has been solved successfully	



# Chapter 4:

## System Design

### 4.1 System Architecture:

The diagram given below provides the display of the general architecture of our system. It will show us how our system will be executing its functions.

In this architecture, model view controller will be used.

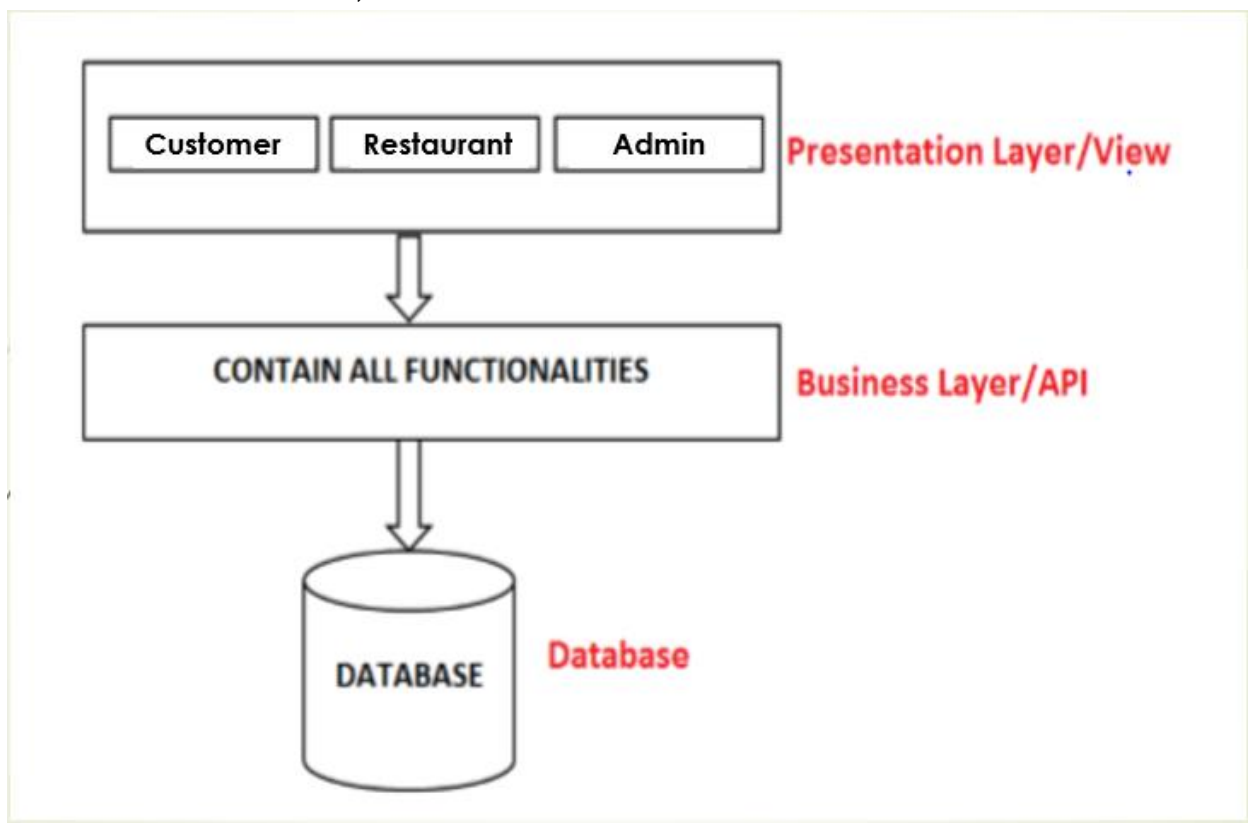


Figure 4.1: System Architecture

- **Presentation Layer:**

In the MVC structure, the presentation layer is the layer at the front-end and comprises of the UI. The web browser provides access to the UI which displays the related content and important data to the user in the system.

- **Business Layer:**

Business Layer comprises of the functional business logic which derives application capabilities core.

- **Database Layer:**

Data storage system or database and the data access layer are the prospects of the Database Layer. For accessing the data, application layer is used through the API calls.

## 4.2 Sequence Diagram:

Sequence diagram shows interaction between different system objects. The tools used for drawing the sequence diagram are Visual Paradigm and Creatly.

### 4.2.1 Manage Accounts and Feedbacks:

Admin will manage the customer and restaurant accounts. Admin will view the feedback of the customer and can block/unblock accounts.

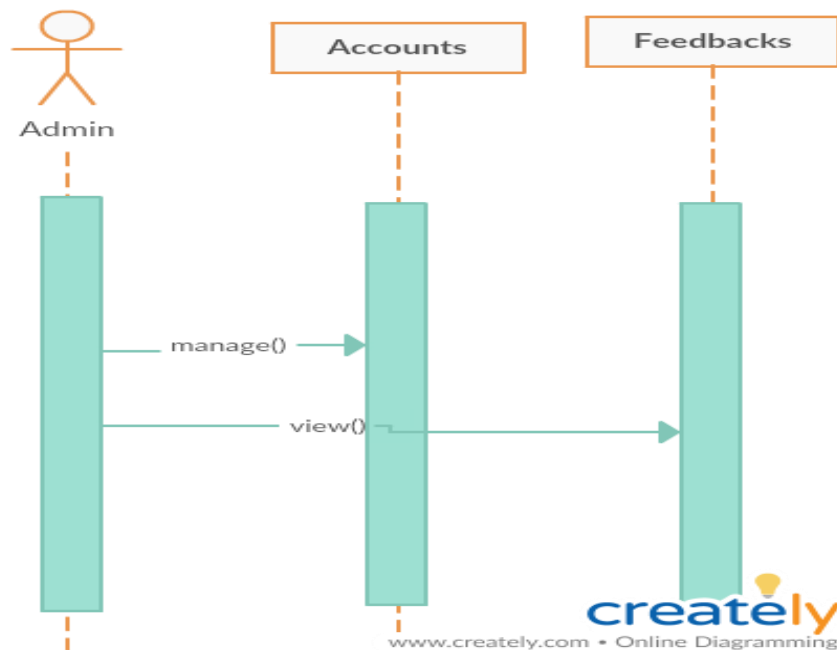


Figure 4.2.1 Manage Accounts and Feedbacks

### 4.2.2 Food Order:

Customer will view the menu and place order. Restaurant will receive the order and food will be served on time. Customer will then give payment to the restaurant.

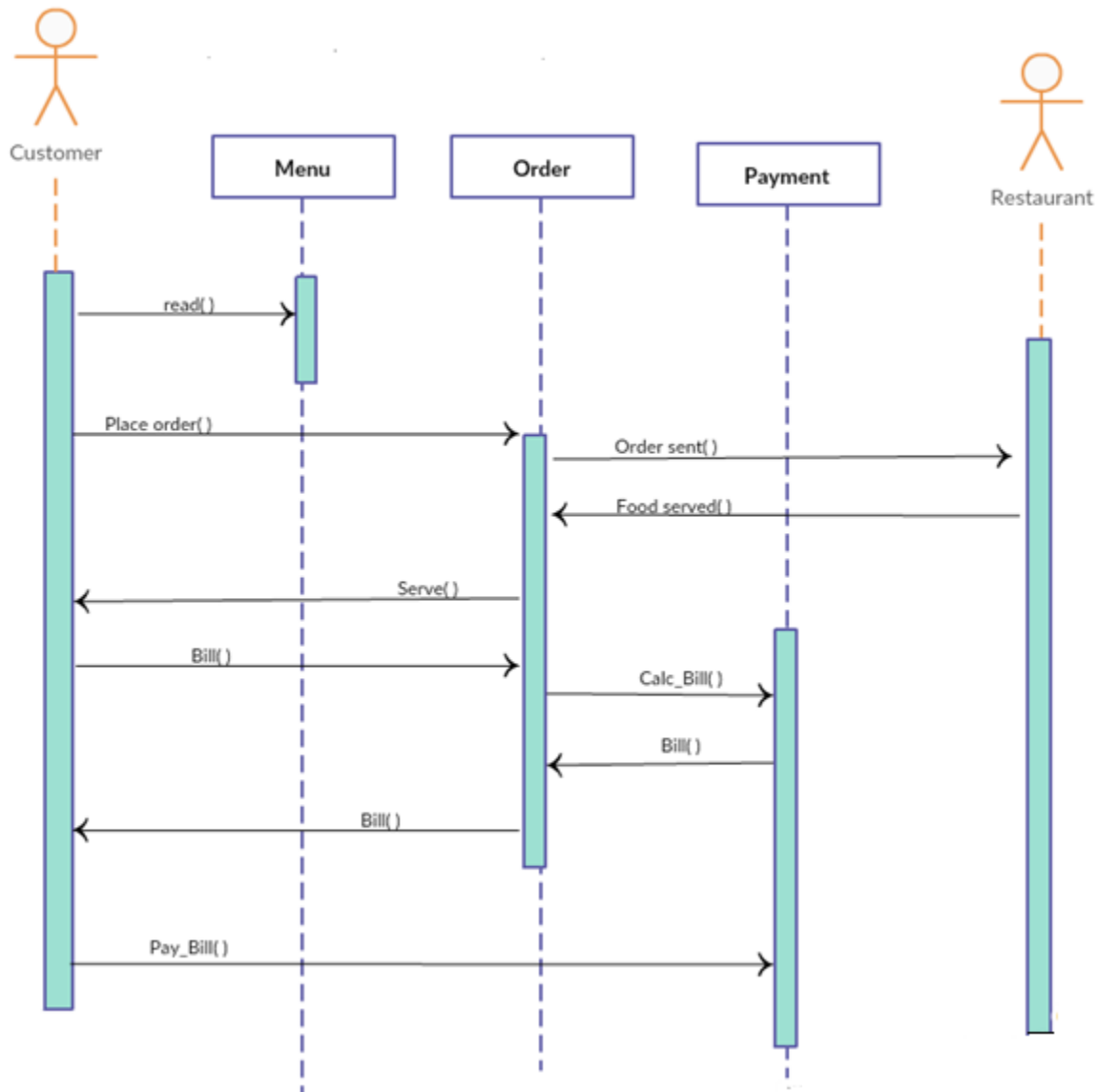


Figure 4.2.2 Food Order

### 4.2.3 Table Reservation:

Customer will make table reservation and enter the number of persons arriving along with the other details. The system will reserve the available tables according to the number of customers. Customers will receive a notification if there is no table available.

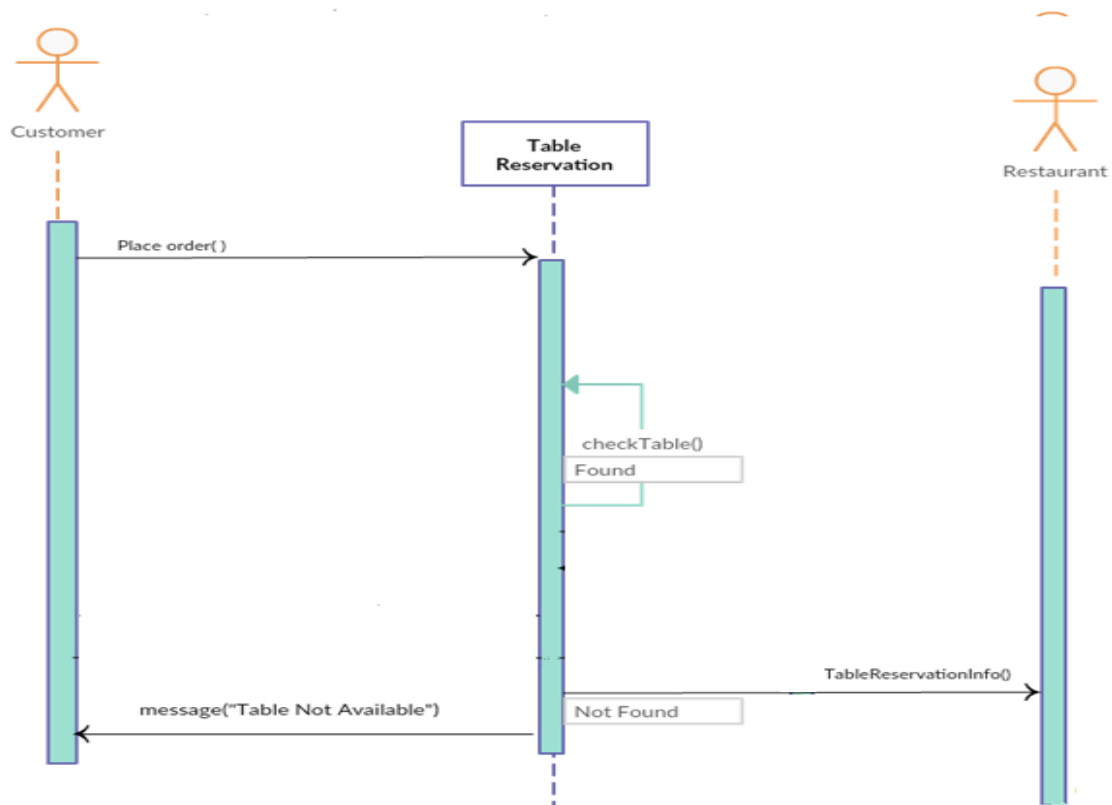


Figure 4.2.3 Table Reservation

## 4.2.4 Signup

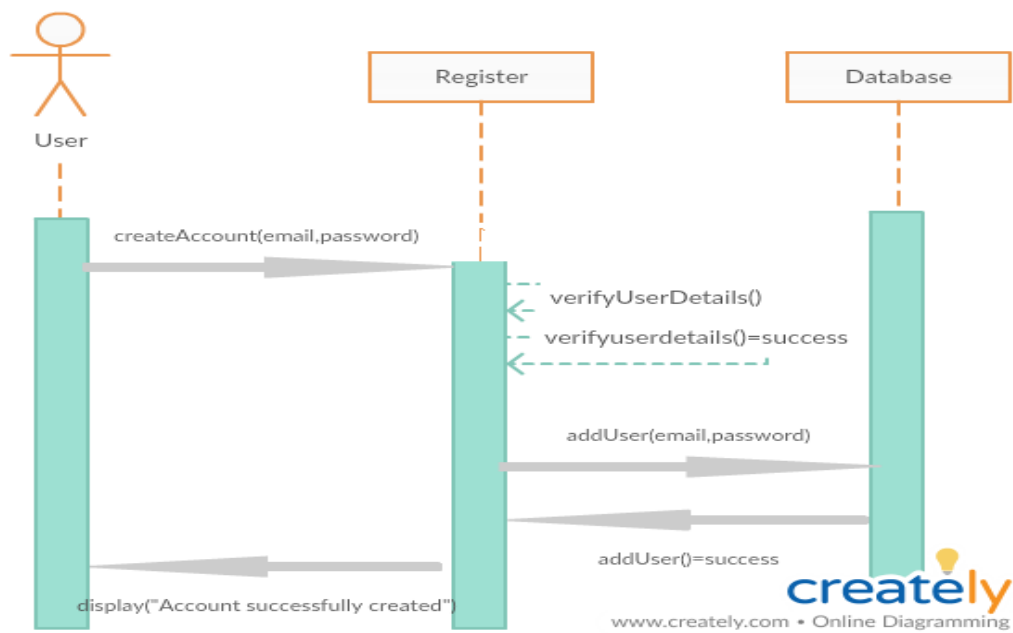


Figure 4.2.4 Sign Up

## 4.2.5 Login

User will provide the username and password to request for login. System will verify the details. If the information is found in the database, then home page will be display to the user otherwise error message will be displayed.

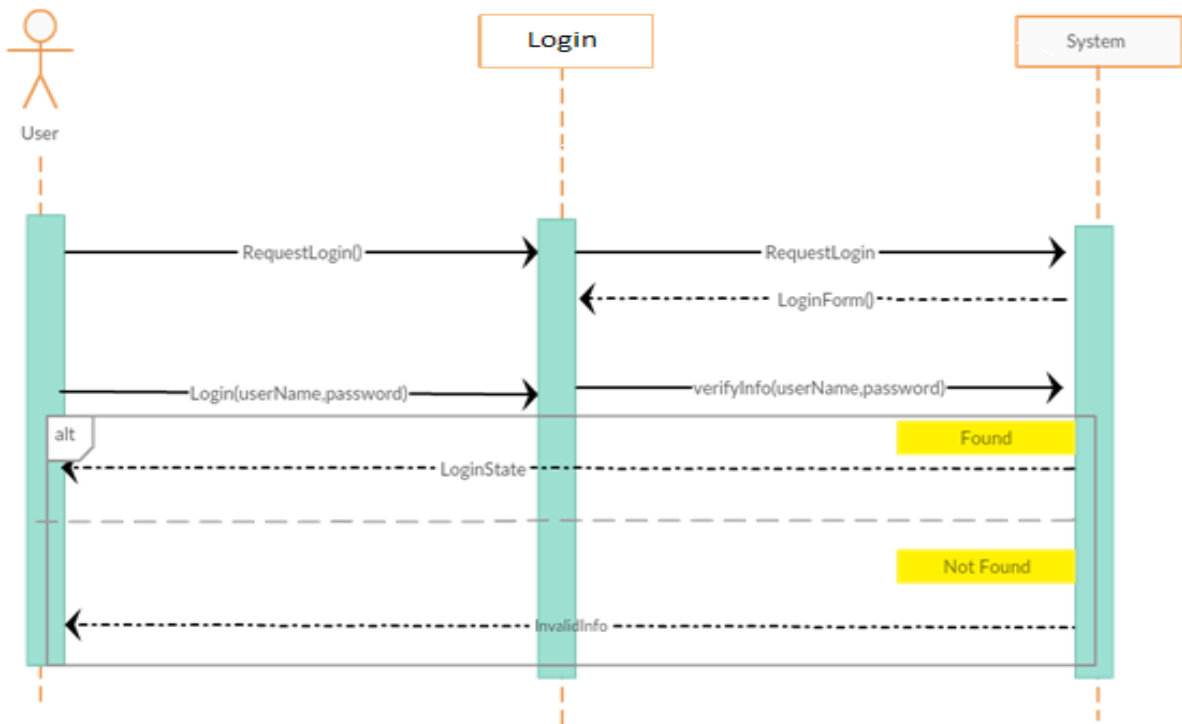


Figure 4.2.5 Login

## 4.2.6 Logout:

User can logout from his/ her account by simply clicking on the logout option.

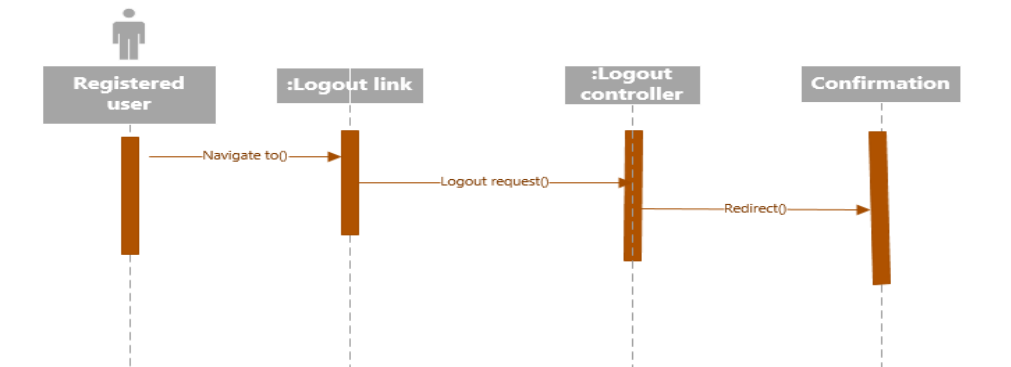


Figure 4.2.6 Logout

### 4.3 Entity Relation Diagram:

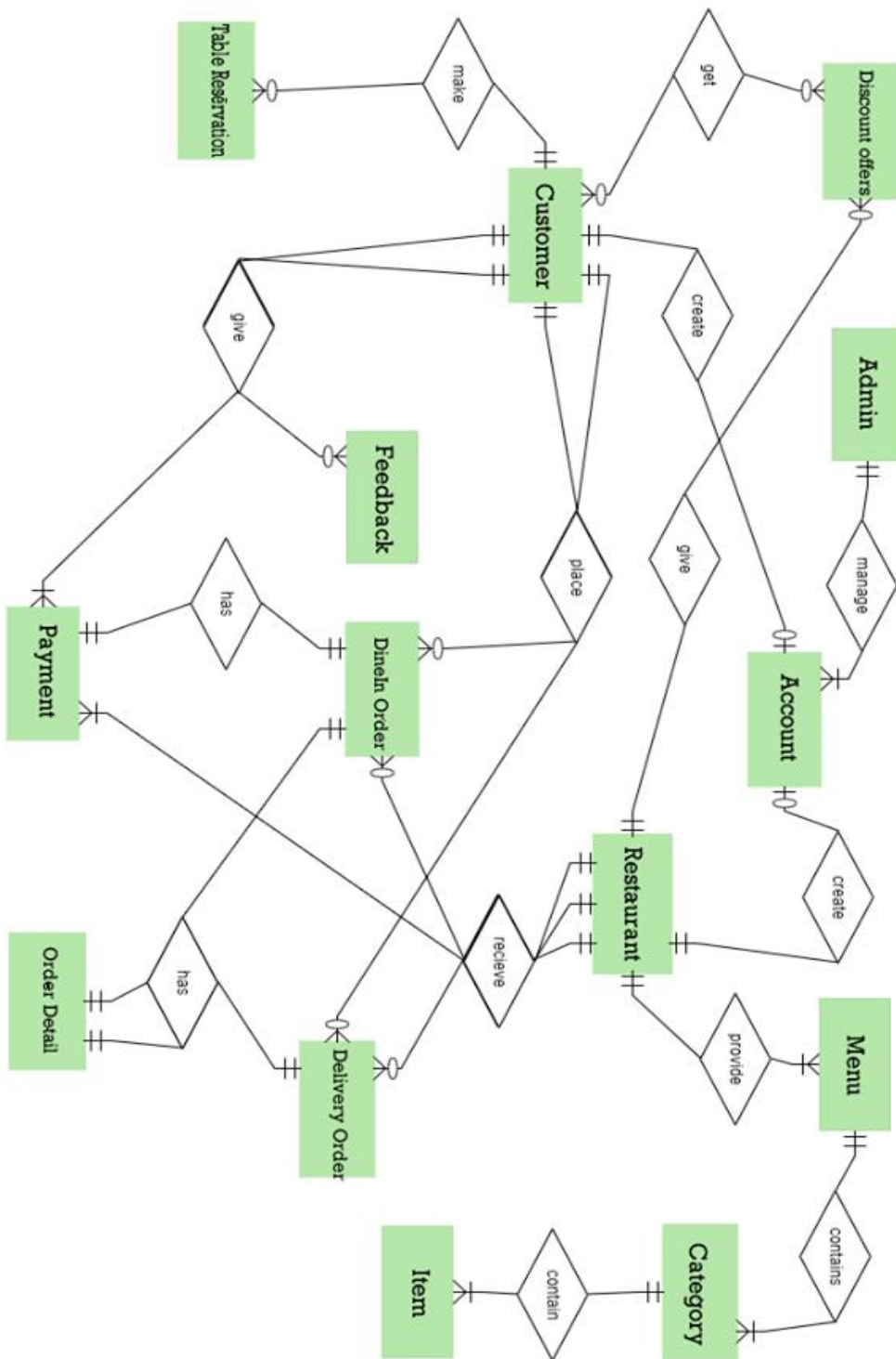


Figure 4.3: Entity Relationship Diagram

### 4.3.1 Attributes:

Table 4.3.1 Attributes

<b>Customer</b>	<ul style="list-style-type: none"><li>• ID</li><li>• Name</li><li>• Email</li><li>• Address</li><li>• Contact number</li></ul>
<b>Restaurant</b>	<ul style="list-style-type: none"><li>• ID</li><li>• Name</li><li>• Location</li><li>• Contact number</li><li>• CNIC number</li><li>• Email</li></ul>
<b>Admin</b>	<ul style="list-style-type: none"><li>• ID</li></ul>
<b>Payment</b>	<ul style="list-style-type: none"><li>• ID</li><li>• Type</li><li>• Cash paid</li><li>• Net price</li><li>• Delivery Fee</li></ul>
<b>Table Reservation</b>	<ul style="list-style-type: none"><li>• Order ID</li><li>• Arrival Time</li><li>• Table ID</li><li>• No of Persons</li><li>• Checkout Time</li></ul>
<b>Delivery Order</b>	<ul style="list-style-type: none"><li>• Order ID</li><li>• Departure Time</li><li>• Arrival Time</li></ul>
<b>Dine In Order</b>	<ul style="list-style-type: none"><li>• Order No</li><li>• Date</li><li>• Arrival Time</li></ul>
<b>Account</b>	<ul style="list-style-type: none"><li>• ID</li><li>• Password</li></ul>

### 4.4 Class Diagram:

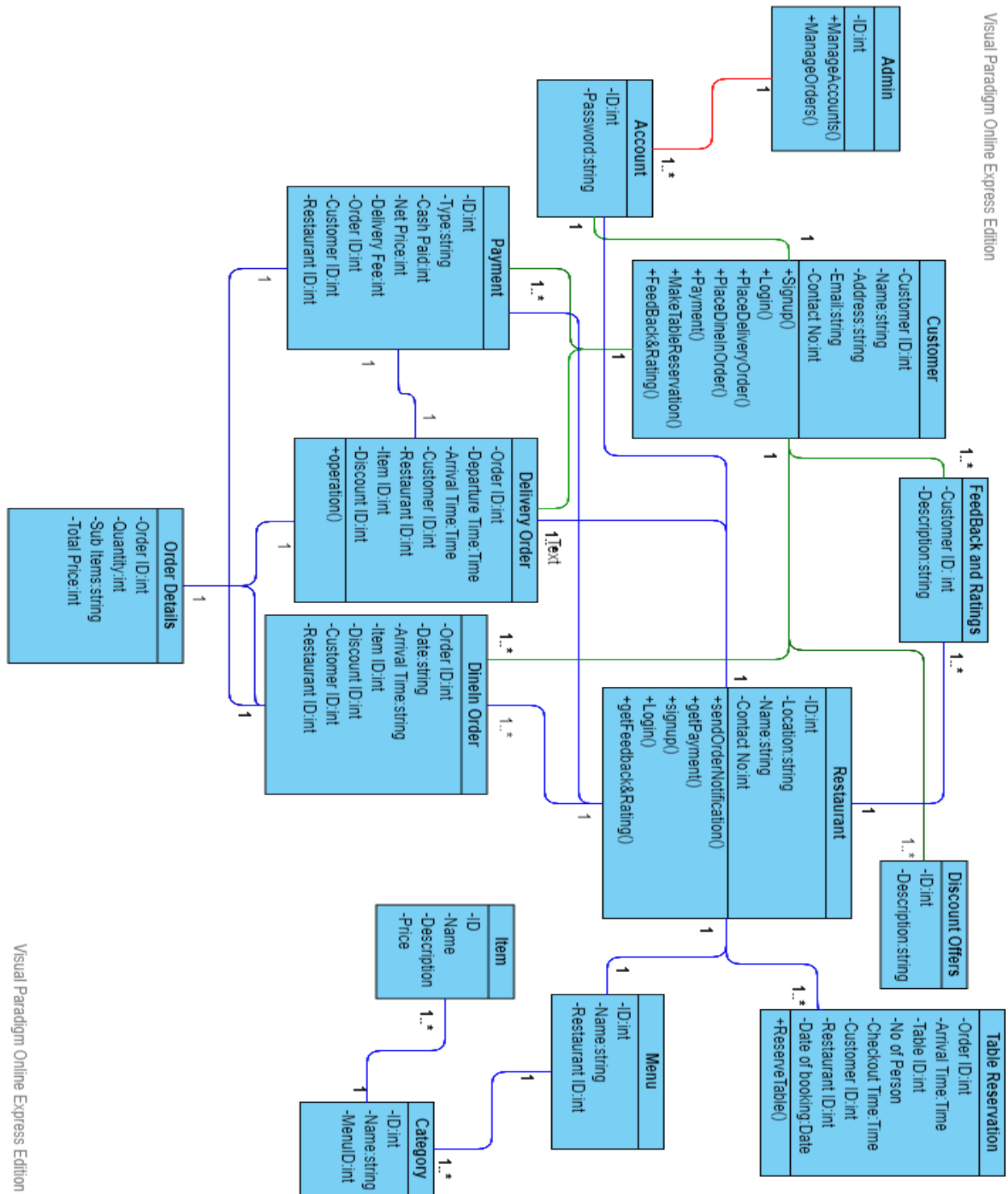


Figure 4.4 Class Diagram



## 4.5 Tables:

### 4.5.1 Customer:

Tables 4.5.1 Customer

Domain	Data Type	Size	Is Null	In unique	Default value	Range
<u>ID</u>	Integer	10	No	Yes	Null	
Name	String	30	No	No	Null	
Address	String	30	No	No	Null	
Email	String	30	No	Yes	Null	
Contact No	Integer	10	No	No	Null	

### 4.5.2 Restaurant:

Table 4.5.2 Restaurant

Domain	Data Type	Size	Is Null	In unique	Default value	Range
<u>ID</u>	Integer	10	No	Yes	Null	
Location	String	30	No	Yes	Null	
Name	String	30	No	No	Null	
Contact No	Integer	10	No	No	Null	
Email	String	30	No	No	Null	

### 4.5.3 Admin:

Table 4.5.3 Admin

Domain	Data Type	Size	Is Null	In unique	Default value	Range
<u>ID</u>	Integer	10	No	Yes	Null	

### 4.5.4 Payment:

Table 4.5.4 Payment

Domain	Data Type	Size	Is Null	In unique	Default value	Range
<u>ID</u>	Integer	10	No	Yes	Null	

Type	String	30	No	Yes	Null	
Cash Paid	String	30	No	No	Null	
Net Price	Integer	10	No	No	Null	
Delivery Fee	String	30	No	No	Null	
Order ID (Fk)	Integer	10	No	Yes	Null	
Customer ID (Fk)	Integer	10	No	No	Null	
Restaurant ID (Fk)	Integer	10	No	No	Null	

#### 4.5.5 Account:

Table 4.5.5 Account

Domain	Data Type	Size	Is Null	In unique	Default value	Range
<u>ID</u>	Integer	10	No	Yes	Null	
Password	String	30	No	Yes	Null	

#### 4.5.6 Delivery Order:

Table 4.5.6 Delivery Order

Domain	Data Type	Size	Is Null	In unique	Default value	Range
<u>Order ID</u>	Integer	10	No	Yes	Null	
Departure Time	Time	30	No	No	Null	
Arrival Time	Time	30	No	No	Null	
Customer ID (Fk)	Integer	10	No	No	Null	
Restaurant ID (Fk)	Integer	30	No	No	Null	
Item ID (Fk)	Integer	10	No	No	Null	
Discount ID (Fk)	Integer	10	No	No	Null	

## 4.5.7 Dine-In Order:

Table 4.5.7 Dine-in Order

Domain	Data Type	Size	Is Null	In unique	Default value	Range
<b>Order ID</b>	Integer	10	No	Yes	Null	
Date	String	30	No	Yes	Null	
Arrival Time	String	30	No	No	Null	
Item ID (Fk)	Integer	10	No	No	Null	
Discount ID (Fk)	String	30	No	No	Null	
Customer ID (Fk)	Integer	10	No	Yes	Null	
Restaurant ID (Fk)	Integer	10	No	No	Null	

## 4.5.8 Order Details:

Table 4.5.8 Order Details

Domain	Data Type	Size	Is Null	In unique	Default value	Range
<b>OrderID (Fk)</b>	Integer	10	No	Yes	Null	
Quantity	Integer	10	No	No	Null	
Sub Items	String	30	No	No	Null	
Total Price	Integer	10	No	No	Null	

## 4.5.9 Feedbacks

Table 4.5.9 Feedback and Ratings

Domain	Data Type	Size	Is Null	In unique	Default value	Range
CustomerID (Fk)	Integer	10	No	No	Null	
Description	String	30	No	No	Null	

#### 4.5.10 Table Reservation:

Table 4.5.10 Table Reservation

Domain	Data Type	Size	Is Null	In unique	Default value	Range
<b><u>OrderID</u></b>	Integer	10	No	Yes	Null	
Arrival Time	Time	10	No	No	Null	
Table ID	Integer	10	No	No	Null	
No of Persons	Integer	10	No	No	Null	
Checkout Time	Time	10	No	No	Null	
Date	Date	10	No	No	Null	
Customer ID (Fk)	Integer	10	No	No	Null	
Restaurant ID (Fk)	Integer	10	No	No	Null	

#### 4.5.11 Discount Offers:

Table 4.5.11 Discount Offers

Domain	Data Type	Size	Is Null	In unique	Default value	Range
<b><u>ID</u></b>	Integer	10	No	Yes	Null	
Description	String	30	No	No	Null	

#### 4.5.12 Menu:

Table 4.5.12 Menu

Domain	Data Type	Size	Is Null	In unique	Default value	Range
<b><u>ID</u></b>	Integer	10	No	Yes	Null	
Name	String	30	No	No	Null	
Restaurant ID (Fk)	Integer	10	No	No	Null	

### 4.5.13 Category:

Table 4.5.13 Category

Domain	Data Type	Size	Is Null	In unique	Default value	Range
<u>ID</u>	Integer	10	No	Yes	Null	
Name	String	30	No	Yes	Null	
Menu ID	Integer	10	No	No	Null	

### 4.5.14 Item:

Table 4.5.14 Item

Domain	Data Type	Size	Is Null	In unique	Default value	Range
<u>ID</u>	Integer	10	No	Yes	Null	
Name	String	30	No	No	Null	
Price	Integer	10	No	No	Null	
Description	Integer	10	No	No	Null	
Category ID (Fk)	Integer	10	No	No	Null	

# Chapter 5:

## System Implementation

### 5.1 System Architecture:

Our System comprises of two main components:

- Website
- Database

### 5.2 Tools Used:

Different tools and strategies have been used for the completion of this project. Some of these tools have been mentioned below:

#### 5.2.1 VS Code Text Editor:

VS Code is the text editor that has been used for writing the initial HTML code during the web design process.

#### 5.2.2 Microsoft Visual Studio:

Microsoft Visual Studio has been used to convert the basic HTML web design into Asp.Net MVC web forms and connected to the database later.

##### 5.2.2.1 ASP.NET:

ASP.NET framework is used for the production of dynamic web-pages that are interactive and data driven over the internet.

#### 5.2.3 Microsoft SQL Server Management Studio:

Microsoft SQL Server Management Studio has been used for the creating the database for Customer, Restaurant and Admin module.

## 5.3 Languages Used:

### 5.3.1 JavaScript:

JavaScript is the scripting language that is used on the client side which means that the source code is processed on the Web Browsers. Google Map JS API is used for the retrieval of maps, directions and markers etc.

### 5.3.2 Structured Query Language (SQL):

An open source Relation Database Management System (RDBMS) is used for the maintenance and execution of data. RDBMS is used for queries like SELECT, DELETE and MODIFY etc.

## 5.4 Methodology:

Mainly, there are three types of interactions with our application.

1. Customer
2. Restaurant
3. Admin

### 5.4.1 Customer Services:

- There will be a login or signup form. Customers will make an account before making a request.
- Provided his / her location, customer can select any restaurant of his / her choice.
- Customer can graphically view the restaurants table status at their desired time. They can book the table by selecting the no. of persons then the system will book the table/tables according to the persons and tables available at that time.
- Customer will be able to see the menu of restaurant according to its categories.
- The customer will order the dishes according to their choice and while placing order, based on their choice; the application will provide recommendations on food items previously reviewed by other customers.
- Customers will get notified via the application when their order is prepared and ready for serving.
- Customers will be billed according to their order.

### 5.4.2 Restaurant Services:

- Restaurant should be able to register to the site giving location and menu of his restaurant.

- After logging in the restaurant will be able to view the table and order bookings placed by customers according to their arrival time.
- Restaurant will be able to see the profit earned at any day and manage the sales and marketing.
- Firstly, the management member has to login and according to the designation the privileges are set. If the staff member is a cook, then he/she can only view the order placed by customers.
- Management will be notified when an order is placed, and they will be able to notify the customer, when the order is ready for serving.
- The management can edit the menu information such as price, items available currently and check if the order is served or not.
- They will also manage the customer feedback.

### **5.4.3 Admin Services:**

The admin module will be like a CMS that will include:

- Admin will be able to manage the restaurants and customers details, who are engaging through this application.
- Admin will be able to manage the details of the table reservations and by the customers and will be responsible to keep the communication between the customers and the restaurants management up and running all the time.
- Admin will have the access to block any restaurant and customer for having bad conduct or bad faith.
- Admin will be working on the feedback provided by the customers and make improvements in the application for the customers to have a better experience.



# Chapter 6:

## System Testing and Evaluation

System testing guarantees that every module of the framework is working appropriately and giving the outcomes as indicated. Every module of the framework is tested independently and after that the application is tested. The main objective behind testing is to analyze whether the developed system meets the required quality guidelines or not. Here is the list of testing techniques:

### 6.1 Interface Testing:

The Graphical User Interface (GUI) relies heavily on the interaction of human beings. GUI comprises of design, colors, user friendliness and all the other aspects that collaborate in improved interaction of computer and the human beings.

### 6.2 Usability Testing:

Usability testing is used to see how much the system is user friendly and interactive. We have tested the different functionalities of our system which were specified in the requirement document.

### 6.3 Integration Testing:

Our system has three different modules which are to provide different services to the respective users hence an integration testing is performed. We initiated the integration testing by generating various functionalities from customer end and verified them on the restaurant and admin end.

### 6.4 Exception Handling:

We have used different cases for testing the exception handling of our system e.g. an @ must be present in an email address was one of the exceptions.

## 6.5 Test Cases:

### 6.5.1.1 Login:

Table 6.5.1 Login Screen Test

<b>Test Case Id</b>	User login	DK-1
<b>Use Case Id</b>	Login Process	DK-BU02
<b>Description</b>	Test the login system in which customer and restaurant have registered themselves and they are now logging in.	
<b>Requirement</b>	Internet connection.	
<b>Initial Condition</b>	Application is running, and login page is available, User knows userID and password.	
<b>Step</b>	<b>Procedural Steps</b>	
1	Go to Login Page	
2	Enter ID and Password	
3	Press Login Button	
<b>Step</b>	<b>Task and Expected Result</b>	<b>Status</b>
1	Validate Password and user ID fields take values correctly.	Pass
2	Validate every field of login screen take values correctly.	Pass
3	If user enter the wrong input show error	Pass
4	UserID input takes int values	Pass
5	Password is not visible	Pass

### 6.5.1.2 Sign Up:

Table 6.5.2 Signup Screen Test

<b>Test Case Id</b>	User signup	DK-2
<b>Use Case Id</b>	Registration process	DK-BU01
<b>Description</b>	Testing the sign-up details of restaurant and customer	
<b>Requirement</b>	Internet Connection.	

<b>Initial Condition</b>	The customer/restaurant provides with the required information which includes his name, address, phone number, location etc.	
<b>Step</b>	<b>Procedural Steps:</b>	
1	Go to customer/restaurant sign up page	
2	Validate all needed fields for registration are present.	Pass
3	Validate all buttons are working.	Pass
4	Validate all required fields are filled up.	Pass
5	Validate all the fields take values correctly.	Pass

### 6.5.1.3 Home Page

Table 6.5.3 Home Screen Test

<b>Test Case Id</b>	Home Page Restaurant/Customer	DK-3
<b>Description</b>	Testing the home page of restaurant and customer	
<b>Requirement</b>	Internet Connection.	
<b>Initial Condition</b>	The customer/restaurant has logged in.	
<b>Step</b>	<b>Procedural Steps:</b>	
1	Go to customer/restaurant home page.	
2	Validate the home page is displayed properly.	Pass
3	Validate every option of home page is working.	Pass
4	Validate the navigation of all available options on the screen are working.	Pass
5	Validate all the fields take values correctly.	Pass

### 6.5.2.1 Login Restaurant/Customer:

Table 6.5.2.1 Login Test

<b>Test Case Id</b>	Customer/Restaurant login	DK-4
<b>Use Case Id</b>	Login Process	DK-BU02
<b>Description</b>	Test the login system in which customer and restaurant have registered themselves and they are now logging in.	
<b>Requirement</b>	Internet connection.	
<b>Initial Condition</b>	Application is running and login page is available, User knows userID and password.	
<b>Step</b>	<b>Procedural Steps</b>	
1	Go to Login Page	

2	Enter ID and Password	
3	Press Login Button	
<b>Step</b>	<b>Task and Expected Result</b>	<b>Status</b>
1	If password and user ID matches login successful	Pass
2	If password and user ID not match Error Show	Pass

### 6.5.2.2 Sign Up Restaurant/Customer:

Table 6.5.2.2 Signup Test

<b>Test Case Id</b>	Restaurant/Customer sign up	DK-5
<b>Description</b>	Testing the sign-up details of restaurant and customer	DK-BU01
<b>Requirement</b>	Internet Connection.	
<b>Initial Condition</b>	The customer/restaurant provides with the necessary information which includes his name, address, phone number, location etc.	
<b>Step</b>	<b>Procedural Steps:</b>	
1	Go to customer/contractor sign up page	
2	Enter the required information.	
3	Press Signup Button	
<b>Step</b>	<b>Task and Expected Result</b>	<b>Status</b>
1	Enter the values in all the required fields.	Pass
3	More than 7-character Password.	Pass
4	Only number enter in ID input.	Pass
5	Password is <b>not</b> visible	Pass
6	In case of wrong input show error	Pass
7	Notification is received for successful /unsuccessful registration	Pass

### 6.5.2.3 Dine IN Food Order Customer/Restaurant:

Table 6.5.2.3 Dine-IN Order Test

<b>Test Case Id</b>	Food Order placed by Customer and received by restaurant.	DK-6
<b>Use Case Id</b>	Dine-In Food Order Process	DK-BU04
<b>Description</b>	The use case explains the dine-in food order placed by customers. The customer will search the desired restaurant, navigate the menu and then place the order by selecting the item and entering the order details with the arrival time. The application will notify the order details to the restaurant.	
<b>Requirement</b>	Network Connection	

<b>Initial Condition</b>	Customer searched for the desired restaurant and select the food item by navigating the menu.	
<b>Step</b>	<b>Procedural Steps</b>	
1	GPS is turn on and allow to locate restaurant.	
2	The customer will provide the name of restaurant or cuisine name.	
3	The customer will select the item along with quantity and amount and also the different options for order details like flavor etc. if there are any.	
4	Customer clicks on the Dine-In order option and will enter the date and arrival time.	
5	Restaurant will receive the order details. The system will notify the customer when the food is ready.	
<b>Step</b>	<b>Task and Expected Result</b>	<b>Status</b>
1	GPS turn on restaurant located	Pass
2	GPS turn off user restaurant located	Pass
3	Validate select the desired item along with different options available.	Pass
4	Customer order placed successfully.	Pass
5	Order received by restaurant successfully.	Pass
6	Notification sent and received successfully.	Pass

### 6.5.2.4 Home Delivery Order:

Table 6.5.2.4 Home delivery Order Test

<b>Test Case Id</b>	Home delivery order placed by customer and received by restaurant.	DK-7
<b>Use Case Id</b>	Home Delivery Order Process	DK-BU05
<b>Description</b>	The use case explains the home delivery order placed by customers and received by restaurant. The customer will search the desired restaurant, navigate the menu and then place the order by selecting the item and entering the order details. The system will notify the order details to the restaurant and arrival time to the customer	
<b>Requirement</b>	Internet Connection	
<b>Initial Condition</b>	Customer searched for the desired restaurant and select the food item by navigating the menu.	
<b>Step</b>	<b>Procedural Steps</b>	
1	GPS is turn on and allow to locate restaurant.	
2	The customer will provide the name of restaurant or cuisine name.	
3	The customer will select the item along with quantity and amount and also the different options	

	for order details like flavor etc. if there are any.	
4	Customer clicks on the home delivery order option.	
5	Restaurant will receive the order. The system will notify the arrival time to customer.	
<b>Step</b>	<b>Task and Expected Result</b>	<b>Status</b>
1	GPS turn on restaurant located	Pass
2	GPS turn off user restaurant located	Pass
3	Validate select the desired item along with different options available.	Pass
4	Customer order placed successfully.	Pass
5	Order received by restaurant successfully.	Pass
6	Notification sent and received successfully.	Pass

### 6.5.2.5 Table Reservation

Table 6.5.2.5 Table Reservation Order Test

<b>Test Case Id</b>	Table reservation order placed by customer and received by restaurant.	DK-8
<b>Use Case Id</b>	Table Reservation Process	DK-BU03
<b>Description</b>	The use case explains the table reservations made by customers. The customer will enter the number of persons and some personal information. The system will reserve the empty table according to the number of persons.	
<b>Requirement</b>	Internet Connection	
<b>Initial Condition</b>	Customer searched for the desired restaurant and select the table reservation option.	
<b>Step</b>	<b>Procedural Steps</b>	
1	GPS is turn on and allow to locate restaurant.	
2	The customer enters the number of persons, date, time of arrival and the checkout time	
3	The system will look for the table or tables according to the number of persons and notify the table no to the customer.	
4	The system will notify the customer if there is no table available at that time.	
5	The restaurant will receive all the reservations with details.	
<b>Step</b>	<b>Task and Expected Result</b>	<b>Status</b>
1	GPS turn on restaurant located	Pass
2	GPS turn off user restaurant located	Pass
3	All the fields validate information entered by the	Pass

	customer.	
4	System assign the no of table or tables correctly.	Pass
5	Order received by restaurant successfully.	Pass
6	Notification sent and received successfully.	Pass

### 6.5.2.6 Admin Panel:

Table 6.5.2.6 Admin Panel Test

<b>Test Case Id</b>	System admin	DK-9
<b>Use Case Id</b>	Manage Restaurants and Customers Process	DK-BU08
<b>Description</b>	Admin login and check the restaurant and customer order and feedback details.	
<b>Requirement</b>	Internet Connection	
<b>Initial Condition</b>	Login successful and the main page opens.	
<b>Step</b>	<b>Procedural Steps</b>	
1	For detail of specific orders click on the respective option	
2	Orders will be shown on the page.	
3	Admin check the table reservation by selecting the option in side bar.	
4	Admin check the feedbacks from the customers.	
<b>Step</b>	<b>Task and Expected Result</b>	<b>Status</b>
1	All orders and reservations are shown correctly.	Pass
2	Feedbacks are shown correctly.	Pass

### 6.5.2.7 Software Performance Test Case:

Table 6.5.2.7 Performance Test

<b>Test Case Id</b>	System Performance Testing	DK-10
<b>Description</b>	Testing the response time of restaurant and customer	
<b>Requirement</b>	Internet Connection.	
<b>Initial Condition</b>	Stable Internet connection	
<b>Step</b>	<b>Procedural Steps:</b>	
1	Validate response time of the whole system	Pass
2	Login to the system.	Pass
3	Validate restaurant search time	Pass
4	Redirect time to the page of any option selected.	Pass
5	Validate Order placement time.	Pass
6	Validate Order receiving time.	Pass
7	Add, delete and update menu.	Pass

8	Notifications receive time.	Pass
---	-----------------------------	------

## 6.6 Compatibility Testing:

Compatibility testing is evaluating the compatibility of the system to check whether the product is equipped for running on various platforms. The important purpose of compatibility testing is to ensure that the system works according to the specified requirements and to check whether the application can be operated in the customer's environment.

Table 6.6 Compatibility Test

Task	Expected Outcome	Actual Result
Windows7 IE Explorer 11	Pass/Fail	Pass
Windows10 IE Explorer 11	Pass/Fail	Pass
Windows7 Firefox 63.0	Pass/Fail	Pass
Windows10 Firefox 63.0	Pass/Fail	Pass
Windows7 Opera 56.0	Pass/Fail	Pass
Windows10 Opera 56.0	Pass/Fail	Pass
Windows7 google chrome 70.0.3538.77	Pass/Fail	Pass
Windows10 google chrome 70.0.3538.77	Pass/Fail	Pass



# Chapter 7:

## Conclusion

The main goal of Dastarkhwan is to provide a web application to the people which they will use to make table reservations at their favorite restaurants, in their desired timings. People can place an order for food as well through the application and will get a time duration for the order to be ready and will get a notification from the restaurant management that their order is prepared, and they can Dine IN any time in their allocated time. People can also order food for home delivery from the same web application and get it delivered at their homes.

### **7.1 Future Work:**

Many improvements can be made in future version when we will market the application. Due to time constraints we were not able fully develop the application professionally. Much improvement will be accomplished in future some of these are listed below.

#### **7.1.1 Google Maps:**

For development purpose, Google API's can be used which come with limited feature moreover to improve map performance, paid API's can be used which provide more features in the mapkit and the main and most efficient feature of real time location tracking.

#### **7.1.2 Membership Payment Procedure:**

As now, we only use cash payment process but in future customers can register themselves with the restaurants and credit money into their accounts which they can use to make table reservations, order food from the web application.

# APPENDIX A

## A.1 Customer Module Homepage:

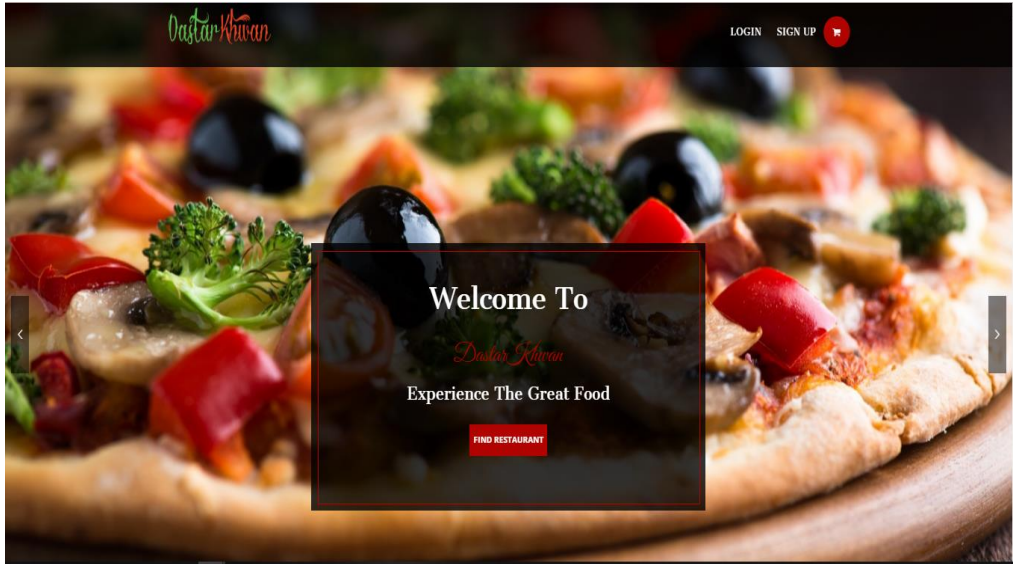


Figure A.1 Customer Homepage

## A.2 Restaurant Search:

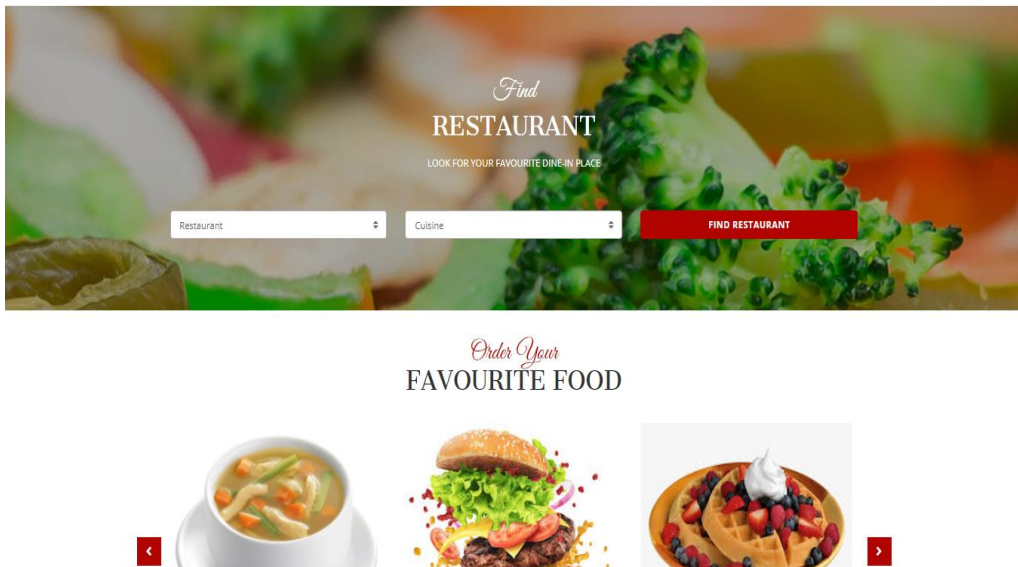


Figure A.2 Find Restaurant

### A.3 Book Your Table:

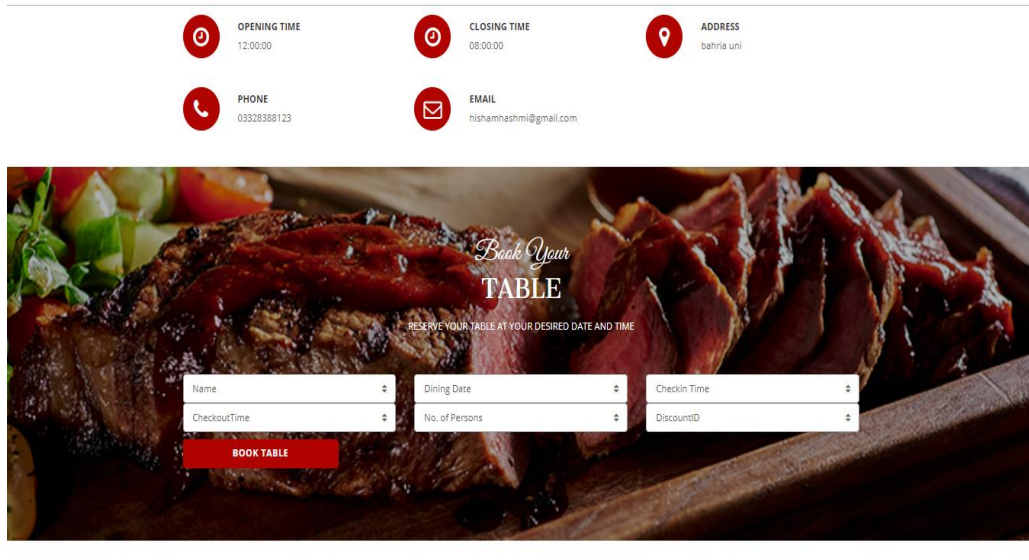


Figure A.3 Book Your Table

### A.4 Menus:

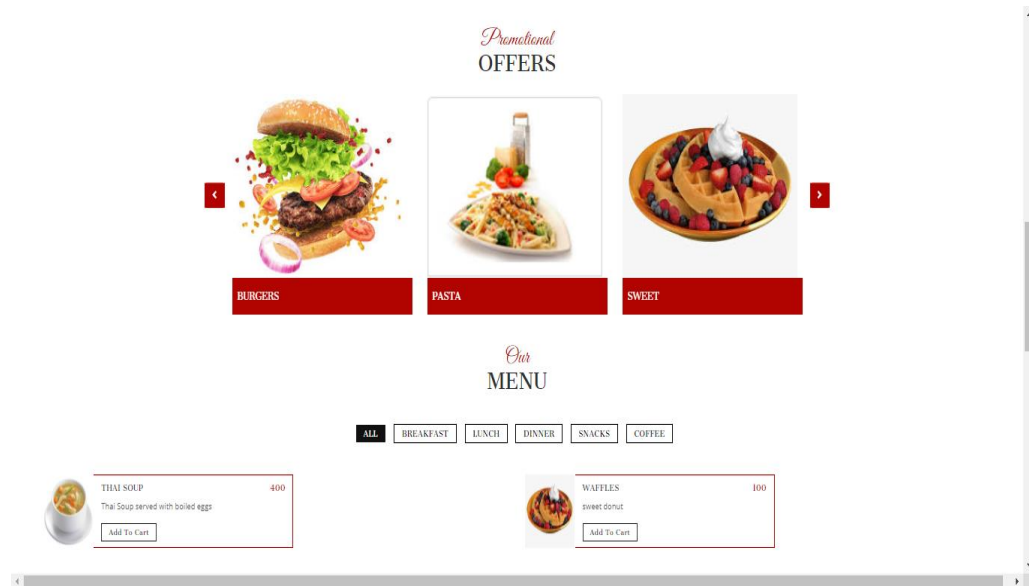


Figure A.4 Menus

## A.5 Restaurant Module Login

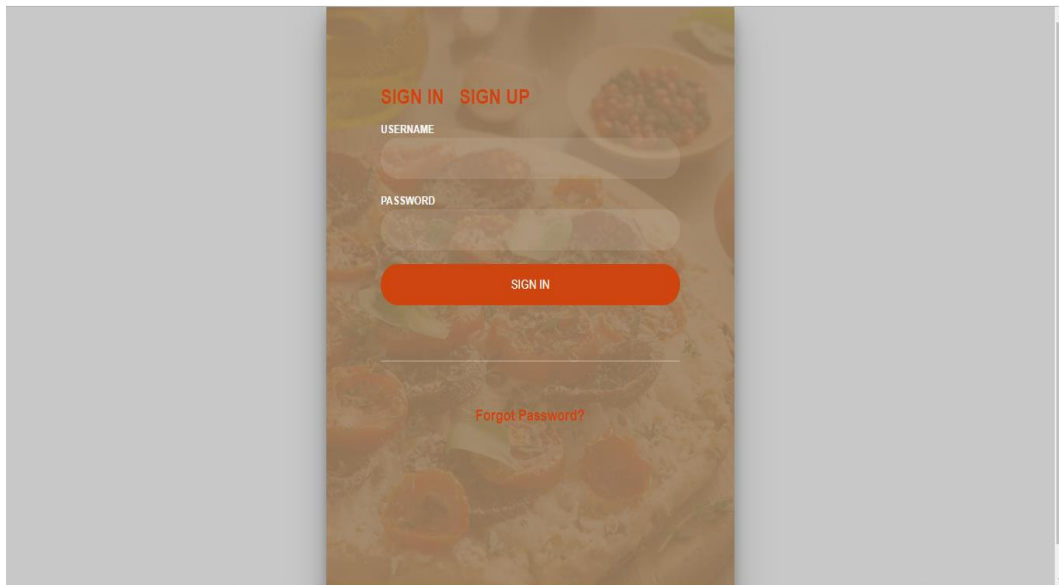


Figure A.5 Restaurant Login

## A.6 Restaurant Homepage

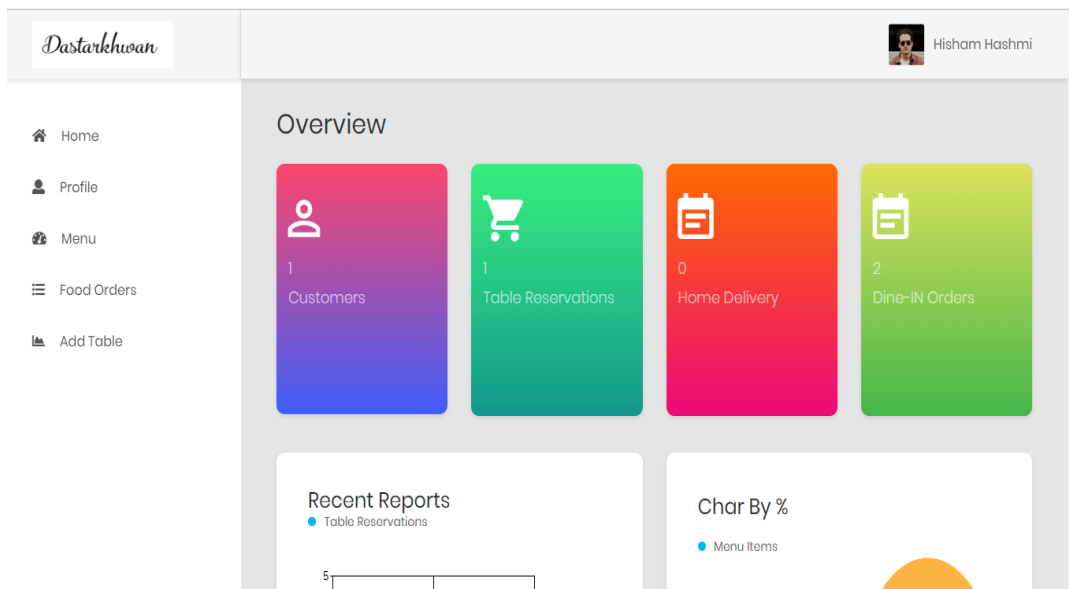


Figure A.6 Restaurant Homepage

# References

- [1] Ajaxtoolkit.net. (2019). *ASP.NET AJAX Control Toolkit*. [online] Available at: <http://www.ajaxtoolkit.net/> [Accessed 9 May 2019].
- [2] Ajaxtoolkit.net. (2019). *ASP.NET AJAX Control Toolkit*. [online] Available at: <http://www.ajaxtoolkit.net/> [Accessed 9 May 2019].
- [3] Foodora.com. (2019). *foodora delivery service | We deliver, you enjoy!*. [online] Available at: <https://www.foodora.com/> [Accessed 9 May 2019].
- [4] <https://www.gloriafood.com/> [Accessed 9 May 2019].
- [5] CanvasJS. (2019). *ASP.NET MVC Charts & Graphs from JSON Data Using AJAX | CanvasJS*. [online] Available at: <https://canvasjs.com/asp-net-mvc-charts/json-data-api-ajax-chart/> [Accessed 9 May 2019].
- [6] Ajaxtoolkit.net. (2019). *ASP.NET AJAX Control Toolkit*. [online] Available at: <http://www.ajaxtoolkit.net/> [Accessed 9 May 2019].
- [7] <https://www.scribd.com/doc/157468790/Online-Hotel-Reservation>