Chapter 1

Introduction

1.1 Specification:-

Project Title	:	Book My Table
Project Type	:	Web Application
Front–End Language	:	React
Back – End Language	:	Mongo DB
Project Guide	:	Papesh Padhare
Submitted By	:	Prince Beladiya(196120316004)
		Darshak kakdiya(196120316026)
		Rihil sanghani(196120316064)
Submitted To	:	Dr. S. & S.S. GHANDHY COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT

1.1 Specification:-

1.2 Overview of languages:-

❖ React:-

➢ React.js is an open-source JavaScript library that is used for building user interfaces specifically for single-page applications. It's used for handling the view layer for web and mobile apps. React also allows us to create reusable UI components.

➢ React allows developers to create large web applications that can change data, without reloading the page. The main purpose of React is to be fast, scalable, and simple. It works only on user interfaces in the application.

❖ Node.js:-

➢ Node.js is an open-source server environment

➢ Node.js is free

➢ Node.js runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)

➢ Node.js uses JavaScript on the server

➢ Here is how PHP or ASP handles a file request:
1. Sends the task to the computer's file system.
2. Waits while the file system opens and reads the file.
3. Returns the content to the client.
4. Ready to handle the next request.

➢ Here is how Node.js handles a file request:
1. Sends the task to the computer's file system.
2. Ready to handle the next request.
3. When the file system has opened and read the file, the server returns the content to the client.
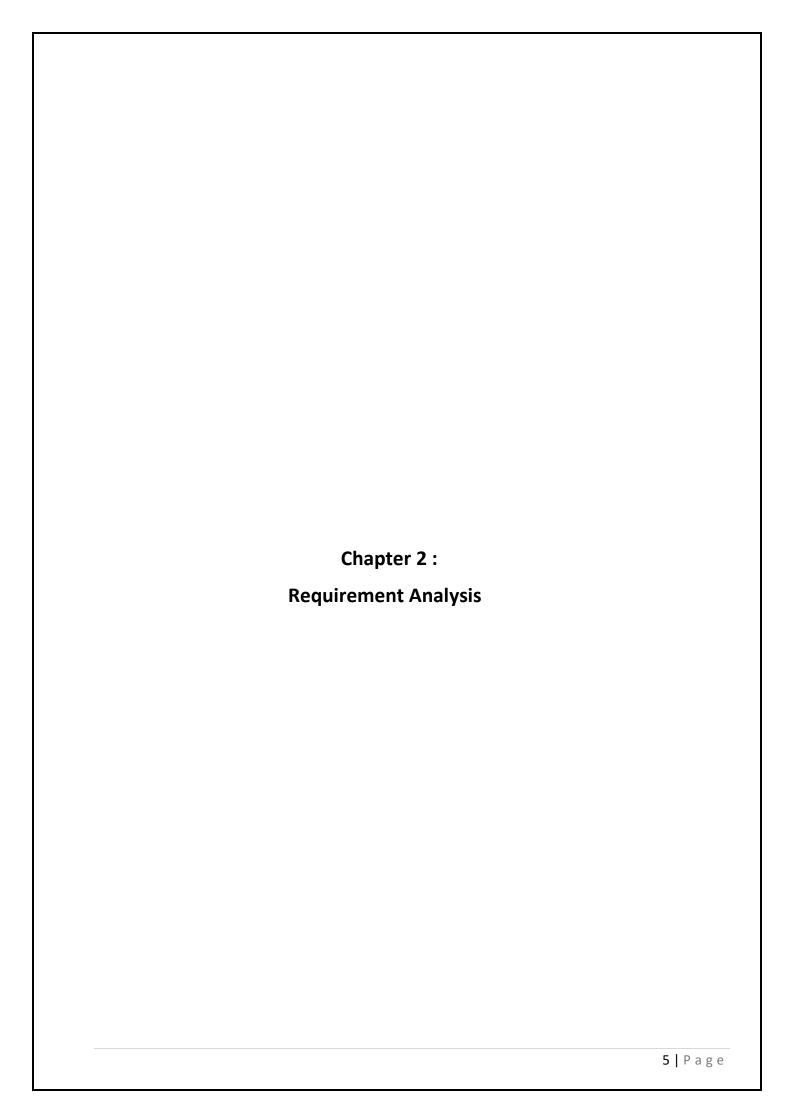4. Node.js eliminates the waiting and simply continues with the next request.

1.3 Scope Of Project:-

➢ We saw in all offline restaurants there we have to wait for our turn and much amounts of time wasted waiting for the meal. When we get our turn we have to wait for the order. But if we use the online booking system than we just need to go to the restaurant and order the food.

➢ So by using this web application , customers don't have to wait for a turn. They can select a timeslot, book a table and order a meal according to their convenient time.

Limitation of Project:-

➢ Time management problem occeurs when customer take more time than time slot allocated or booked.
➢ Online payment integration is not provided.
➢ The user can't use forgot password functionality because OTP is paid.
➢ There are not proper mapping between No of customers and restaurant's table(no of seats)
.

Chapter 2 :

Requirement Analysis

2.1 Requirement Analysis:-

♦ **Users:-**
- ➤ User Registration.
- ➤ Table booking.
- ➤ Update profile.
- ➤ Cancle booked table.

♦ **Vendors:-**
- ➤ Vendor Registration.
- ➤ Restaurant registration by the vendor.
- ➤ Add restaurant.
- ➤ Update profile.
- ➤ All own restaurants details provided to check, update and delete.

♦ **Admin:-**
- ➤ Admin can check all details of restaurants, user and vendors.
- ➤ Admin can activate / Deactivate restaurant.
- ➤ Admin can activate / Deactivate user.
- ➤ Admin can activate / Deactivate vendor.

♦ **Provide a secure environment.**

Software Requirement:-

- ➤ Web Browser.

2.2 Problem-solving techniques:-

- There are two types of problem-solving techniques :
    1. Bottom-up approach
    2. Top-down approach

- We use the Bottom-up approach in our project.

- ❖ **Bottom-up approach:-**
    - **Step – 1:** Problem definition.
        - We saw in all offline restaurants there we have to wait for our turn and much amounts of time wasted waiting for the meal. When we get our turn we have to wait for the order. But if we use the online booking system than we just need to go to the restaurant and order the food.

        Problems :-
        - **Problems are provide at page no. 10 [2.3]**

        - We faced problems in GUI designing (Bootstrap).
        - We faced problems in validation checking. When we hit API that time we get the error because that time validation is not provided.
        - We faced problems in the database when trying to upload images.
        - We faced problems in providing a map facility.
        - We faced problem in providing facility of forget password.

    - **Step – 2:** Cause determination.
        - Then after finding that how can this error or problem in UI is generated.
        - We take all details about the problem and then find creation reasons.

    - **Step – 3:** Ideas generation.
        - Then all team members joined in the google meet or meet somewhere and then discuss the problem.
        - And find all possible solutions together and find some solutions for the generated problems.

+ **<u>Step – 4:</u>** Best solution selection.
  - ➢ We use trade-off analysis and compare all solutions and check which solution is the best.

+ **<u>Step – 5:</u>** Act.
  - ➢ Take this solution as a mini project and then all team members started developing the solution.
  - ➢ This is how we solve the problem.

2.3 Project life cycle model:-

- ❖ **Agile Model:-**
- ➢ The agile SDLC model is a combination of iterative and incremental process models with a focus on process adaptability and customer satisfaction by rapid delivery of working software products. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross-functional teams working simultaneously on various areas like –
  - ✓ Planning
  - ✓ Requirements Analysis
  - ✓ Design
  - ✓ Coding
  - ✓ Unit Testing and
  - ✓ Acceptance Testing.

- ➢ At the end of the iteration, a working product is displayed to the customer and important stakeholders.
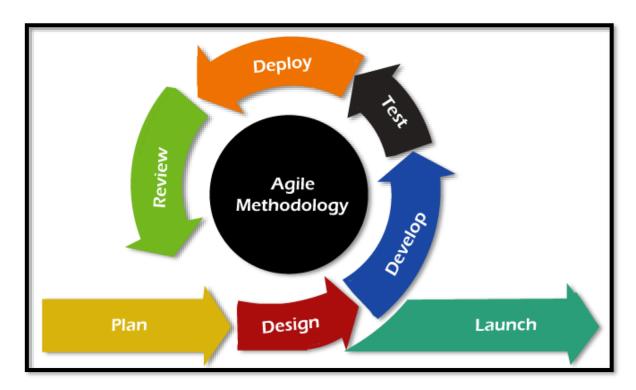- ➢ -Here is a graphical illustration of the Agile Model –

Figure 1

1) <u>Phase – 1: Requirement (Plan):-</u>

➢ In this phase, we collected all the requirements [2.4]. We understood this product and then requirements are gathered us and it is a very important step for the agile model. We collect the functional and non-functional requirements of this product.

❖ <u>Existing System of Table booking:-</u>

1. **Paper-based live booking:** If anyone wants to go to eat at the restaurant then that person has to go to the restaurant and then that person has to say at the reception to book the table and then wait for the turn.
2. **Call-based booking:** In this system, we have to inform the restaurant receptionist to book the table in advance then the receptionist is write our details (Time, number of people, name, etc.).
3. **Advanced booking:** If anyone wants to go to eat at the restaurant then that person has to go to the restaurant and then that person has to say at receptions to book the table for future and pay the deposit for this advanced booking.

❖ <u>Problems with the Existing Table Booking:-</u>

1. **Time Consuming:** People have to go to a restaurant and then wait for a booking to the reception and then wait for their turn. So, this process is very time-consuming.
2. **Communication Problem:** If we want to book the table from home we have to call the restaurant and then book. But here some problems, if we are trying to call the restaurant but the restaurant's mobile is busy or switched off or also because of some errors we do not book by the mobile.
3. **Not Select Best:** Here we cannot check the restaurant's ratings and reviews. So we cannot select the best restaurant.
4. If we book the table at advanced then we have to go to the restaurant and this is time-consuming and we have to go for only booking that is not good.

Requirement analysis is shown below at 2.4.

2) <u>Phase – 2: Design:-</u>

➢ We made design as per the requirements on the paper.

➢ We select the bootstrap and CSS for development.

➢ We take reference from other websites to made design of all components.

3) <u>Phase – 3: Develop:-</u>

➢ All members of the team are taking some Components and then start development on that components.

➢ All related components are developed by team members.

➢ Darshak kakdiya was developed the backend and the all the APIs for add restaurants, book tables, delete user, vendor, admin etc.

➢ Rihil sanghani & prince beladidya made full frontend (all components) such as table booking, home, add restaurant, etc.

4) <u>Phase – 4: Test **[Test cases shown at page no. 54 [2.3]]**</u> :-

➢ In this phase, all team members are checking that if they face any error or UI problems and they find then we solve that problem or error by the bottom-up approach.

5) <u>Phase – 5: Deploy & Review:-</u>

➢ We deploy project on the localhost server then taking review from our guide papesh padhare and friends.

➢ After getting review we consider bottom-up approach for solve problem and try to make website much better.

6) <u>Phase – 7: Launch:-</u>

➢ Launch application for the end-user.

❖ **Proposed System:-**

➢ The project is related table booking System.

➢ The project maintains three levels of user:-

   ✓ Administrator Level

   ✓ Vender Level

   ✓ Client level

- The main facilities available in this project are:-
  - ✓ Maintaining Table-Booking
  - ✓ Providing all information about the restaurant.
  - ✓ Providing cancellation process.
  - ✓ No fraud (Providing a safe and secure environment).

2.4 Software Requirement Specification(SRS):-

❖ **Requirements:**

➢ There are two types of Requirements:-

    1. **Functional Requirements.**

    2. **Non- Functional Requirements.**

❖ **Functional Requirements :**

USER SIDE

User Login

    1. **Login**

    2. **Sign up**
    ➢ Users can sign up with personal details.
    ➢ If the sign-up process is complete then the system gives the message and email to the user.

Reservation Panel

    1. **Select city and area**
    ➢ Select city and area to find the nearer restaurants.

    2. **Select food types**
    ➢ Select food type to find a restaurant with restaurant normal details and average detail.
    ➢ Sort a list of restaurants that have this food type where your selected food type is available for eat.

    3. **Select restaurant**
    ➢ Select a restaurant from a sorted list.

    4. **Show details of the restaurant**
    ➢ Show the restaurant's name, description, photos of the restaurant, location, available food types with price, offers by the restaurant, and reviews.

    5. **Select a book or cancel**
    ➢ Book option for a book this restaurant.
    ➢ Cancel for going back to your sorted list.

6. **Number of person entry**
➤ Enter no. of person for checking restaurant can feed entered persons or not.
➤ if the capacity Is over the restaurant's capacity then go to a sorted list with a simple message alert.

7. **Select time slot**
➤ If the capacity of a person is not over to the restaurant's capacity then the user has to select a time slot for eating food.
➤ Otherwise, the user can click on cancel (go to sorted list).

8. **Select the date and enter other details**
➤ You can select the date for the current booking and also for the advanced booking.
➤ You have to enter some details to reserve a table in the restaurant.
➤ After all, detail-filled you can book the table in the restaurant but if that table is not full then and then you can book otherwise you can't book that table.

Your Reservation

1. **See bookings**
➤ Here, the user can see the user's all bookings that are completed or which are ongoing.
➤ Also if the user wants to cancel the ongoing table then he/she can but if it is before one hour otherwise the user can't cancel that booking.

Offers

1. **Showing live offers**
➤ The system show user offers from the admin side.

User Account

1. **Update profile**
➤ Users can update their profile's all details.

2. **Help**
➤ Users get help from the admin

3. **Contact Us**
➤ Users can contact the admin for any queries.

4. **Log in/Log out**
➤ If the user is logged in user can do log out.
➤ If the user is logged out then the user can do log in.

Vendor Login

1. **Login**

2. **Sign up**
   ➢ The restaurant owner can sign up with personal details and some restaurant details.
   ➢ If the sign-up process is complete.
   ➢ Then these sign-up details go to the admin to check these details the restaurant is a fraud or not and then the system gives the message and email to the user from the admin your registration is passed or not.

Main menu

1. **Your Restaurant**
   ➢ The restaurant owner can show restaurant details and also can update that detail.
   ➢ Upload food types details
      ✓ The restaurant owner is upload food types which is available in that restaurant.
      ✓ Restaurants can also add prices of that food types with images.
      ✓ Also, upload the name of that food.
   ➢ Enter description
      ✓ Restaurants can also add a description for users.
   ➢ Add offers
      ✓ Add offers for users.

2. **(add, delete – update) Holiday**
   ➢ The owner also can add, update, delete holiday detail.
   ➢ The owner adds which day for the holiday that day in the week that restaurant is not able to select by user.

3. **Time managemnt**
   ➢ The owner adds, updates, or deletes the timing of all days of the week.
   ➢ So the user can see that restaurant for this entered time by the owner of the restaurant.
   ➢ Also, you can add, update, delete the full day timing slot in your comfort.
   ➢ The timing slot Is shown to the user for booking which the owner set for the owner's restaurant.

4. **Booking**
➢ The owner of the restaurant can see his full-day booking, also see details of that user to confirm user booking or cancel the booking.
➢ The owner can see slots here which is booked and which are empty and also booked slot as offline booking so the user has not booked that slot.

5. **Help, Contact Us**
➢ Admin provides help to the restaurant owner.
➢ Also, the restaurant owner can contact the admin about queries.

ADMIN SIDE

➢ Admin can see detail about users(ID, name, email, mobile number, date).
➢ Admin can see details about restaurants(name, address, contact number, facility, status, table, opening time, closing time, holiday, food type, food category).
➢ Admin can unblock the blocked users.
➢ Admin can manage all activities of the restaurants. Admin verifies the restaurant (fraud or not).
➢ Admin can de-activate user or vendor accounts. Help to provide to users and also provide to the restaurants.
➢ Backend support.

❖ **Non - Functional Requirements:-**

1. **Database:-**
   ➢ The Database Mongoose should store all the data and activity of the user system.
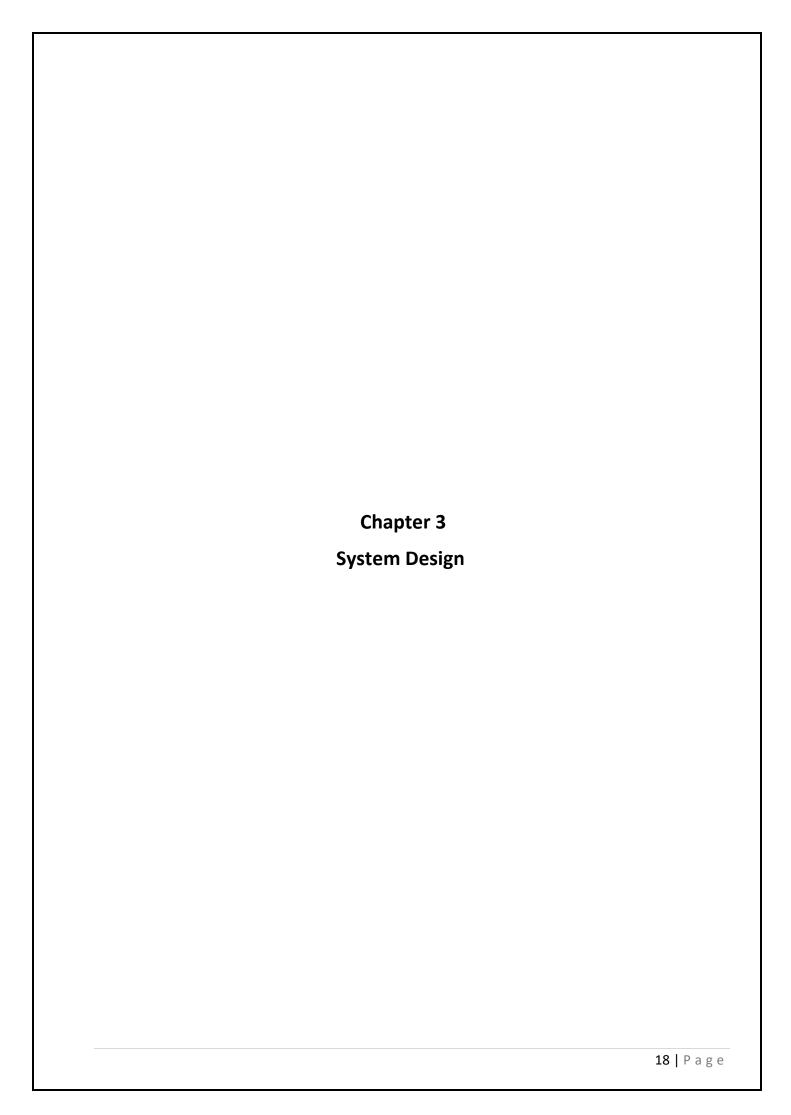
2. **Platform:-**
   ➢ The Web Application should run on a Web Browser.

3. **Security:-**
   ➢ The user details and the restaurant details are stored in the database and also it is secure because they cannot change or see by any user.
   ➢ Net banking is the most secure.

Chapter 3

System Design

3.1 E R Diagram:-
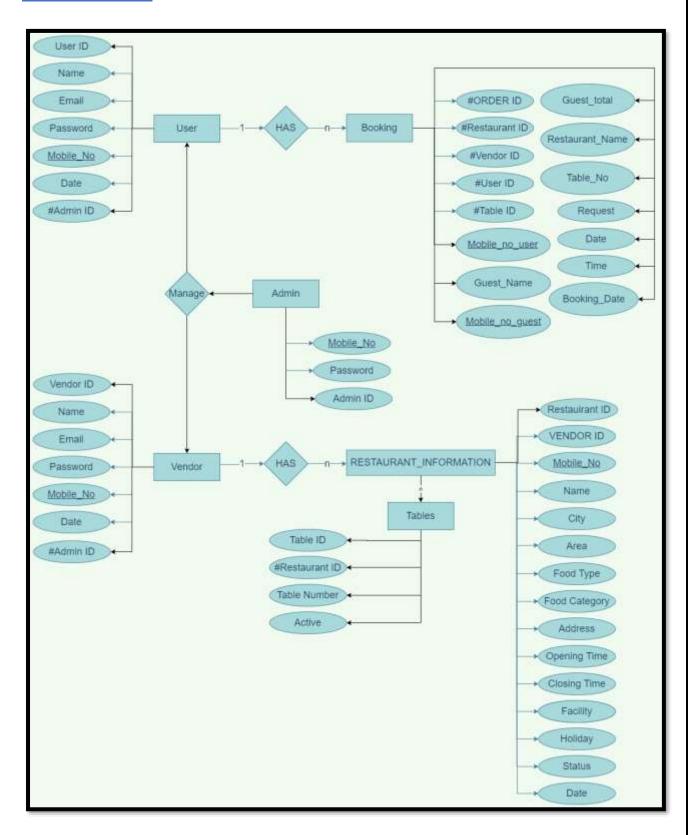
Figure 2

3.2 Data Flow Diagram [DFD] :-

❖ Zero level DFD :-
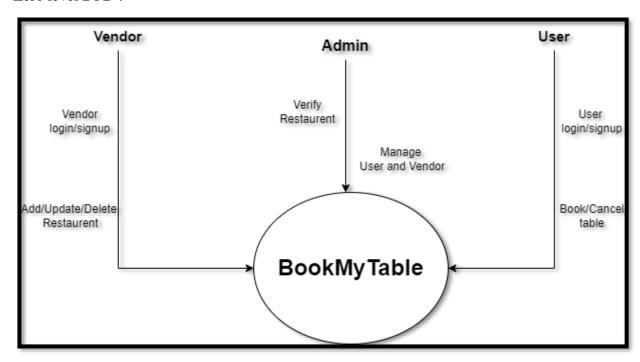
Figure 3

❖ 1st level DFD (Admin):-
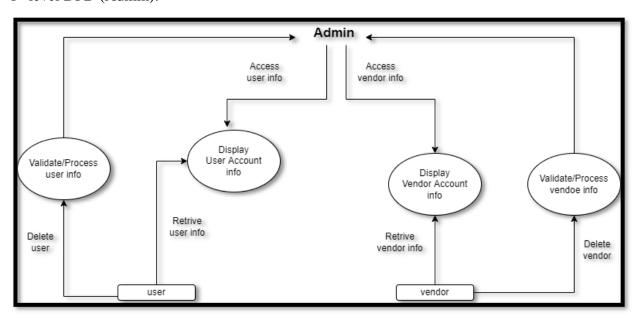
Figure 4

❖ 1st level DFD (User):-
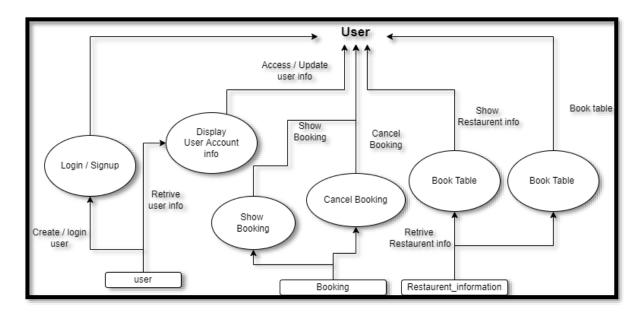
Figure 5

❖ 1st level DFD (Vendor):-
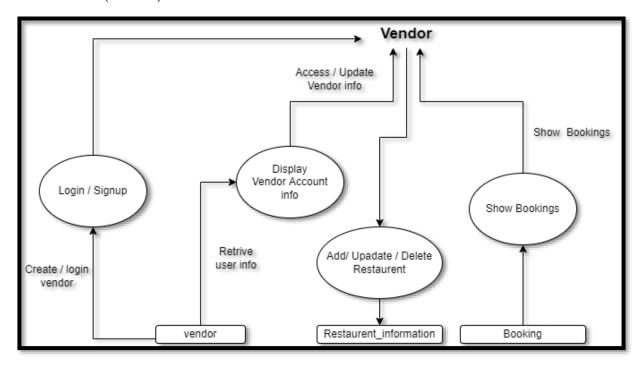
Figure 6

3.3 Data Dictionary:-

❖ User Information:-

⌂ users	
_id ObjectId	**Name** String
1 61ebb6e2a3b326e03ed08998	"Prince Beladiya"
2 61f11ca3521463634b6953f2	"Prince007"
3 6213ae282923bff678b00f80	"Beladiya"

Email String	**Password** String
"princebeladiya007@gmail.com"	"$2a$10$Spyb7GmU6vrXeYDYH3dnKOv
"prince@gmail.com"	"$2a$10$gyijmsIgm.4kbDsUhB.8j.m
"beladiya@gmail.com"	"$2a$10$3HfG6MTG1evgaFfhnRdoJ.h

Mobile_no Mixed	**Date** Date
9276805977	2022-01-22T07:48:50.435+00:00
9999999999	2022-01-26T10:04:19.479+00:00
1452036122	2022-02-21T15:22:16.645+00:00

❖ Vendor Information:-

⌂ vendors	
_id ObjectId	**Name** String
1 61ebc747e52182b39086a05c	"Prince Beladiya"
2 622efb4d629284bc0509d632	"Darshak kakadiya"

Email String	**Password** String
"princebeladiya007@gmail.com"	"$2a$10$jWecpmIJf40LUCidpuKhtO6
"darshak@gmail.com"	"$2a$10$advUOfHNEzzi0vGzsNiQGuv

Mobile_no String	Date Date
"9276805977"	2022-01-22T08:58:47.275+00:00
"1236547890"	2022-03-14T08:22:37.000+00:00

❖ Admin Information:-

admins

	_id ObjectId	Mobile_no String	Password String
1	62107badbddfb4bccc943b3a	"0000780000"	"$2a$10$aTPkdqj2SgjowDCdGeKKWe7

❖ Restaurants Information:-

restaurant_informations

	_id ObjectId	Vendor ObjectId
1	61fe362a5b2f8b5bcc490a80	61ebc747e52182b39086a05c
2	6201049db132bec687ede77f	61ebc747e52182b39086a05c
3	620109feb132bec687ede7a1	61ebc747e52182b39086a05c

Name String	City String
"Prince Beladiya"	"Surat"
"Beladiya"	"Surat"
"Sin Shen"	"Surat"

Area String	FoodType String
"Varachha Road"	"Both"
"Varachha Road"	"Both"
"Varachha Road"	"Both"

	FoodCategory String	Address String
1	"Chanise"	"varachcha Road"
2	"Punjabi"	"varachcha Road"
3	"Gujarati, Punjabi, Italian, Ch	"varachcha Road"

TimeOpen String	TimeClose String
"10 AM"	"10 PM"
"9 AM"	"12 PM"
"8 AM"	"12 PM"

Contact String	Facility String
"1452368097"	"Lodge, Waiting Area, Toilets,
"1459632870"	"Entrance Lobby, Queue, Serving
"1203405678"	"Lodge, Waiting Area, Toilets,

Holiday String	Active Boolean
"Sunday"	true
"Sunday"	true
"No-Holiday"	true

Table_require Int32	Date Date
8	2022-02-14T14:26:58.378+00:00
6	2022-02-16T10:40:29.094+00:00
8	2022-02-19T08:36:31.826+00:00

❖ Tables Information:-

	_id ObjectId	Restaurant ObjectId	Table_No Int32	Active Boolean
1	622ecd1c83fadd17d4211d2d	622ecd1c83fadd17d4211d2a	1	true
2	622ecd1c83fadd17d4211d2f	622ecd1c83fadd17d4211d2a	2	true
3	622ecd1c83fadd17d4211d31	622ecd1c83fadd17d4211d2a	3	true
4	622ecd1c83fadd17d4211d33	622ecd1c83fadd17d4211d2a	4	true
5	622ecd1c83fadd17d4211d35	622ecd1c83fadd17d4211d2a	5	true
6	622ecd1c83fadd17d4211d37	622ecd1c83fadd17d4211d2a	6	true
7	622ecd1c83fadd17d4211d39	622ecd1c83fadd17d4211d2a	7	true
8	622ecd1c83fadd17d4211d3b	622ecd1c83fadd17d4211d2a	8	true
9	623db2f06ef3c4d2ab345a5e	623db2f06ef3c4d2ab345a5b	1	true
10	623db2f06ef3c4d2ab345a60	623db2f06ef3c4d2ab345a5b	2	true

❖ Booking Information:-

bookings

	_id ObjectId	Restaurant ObjectId
1	623c47ed8f444b84b6e7d563	622ecd1c83fadd17d4211d2a
2	623c47ed8f444b84b6e7d569	622ecd1c83fadd17d4211d2a
3	623c47ed8f444b84b6e7d56f	622ecd1c83fadd17d4211d2a

Vendor ObjectId	User ObjectId
61ebc747e52182b39086a05c	61ebb6e2a3b326e03ed08998
61ebc747e52182b39086a05c	61ebb6e2a3b326e03ed08998
61ebc747e52182b39086a05c	61ebb6e2a3b326e03ed08998

Table ObjectId	Mobile_no_user String
622ecd1c83fadd17d4211d33	"9276805977"
622ecd1c83fadd17d4211d35	"9276805977"
622ecd1c83fadd17d4211d37	"9276805977"

	Guest_Name String	Mobile_no_guest String
1	"Prince Beladiya"	"9824673114"
2	"Prince Beladiya"	"9824673114"
3	"Prince Beladiya"	"9824673114"

Guest_Total String	Restaurant_Name String
"5"	"Crazy Cafe"
"5"	"Crazy Cafe"
"5"	"Crazy Cafe"

Table_No Int32	Request String
4	""
5	""
6	""
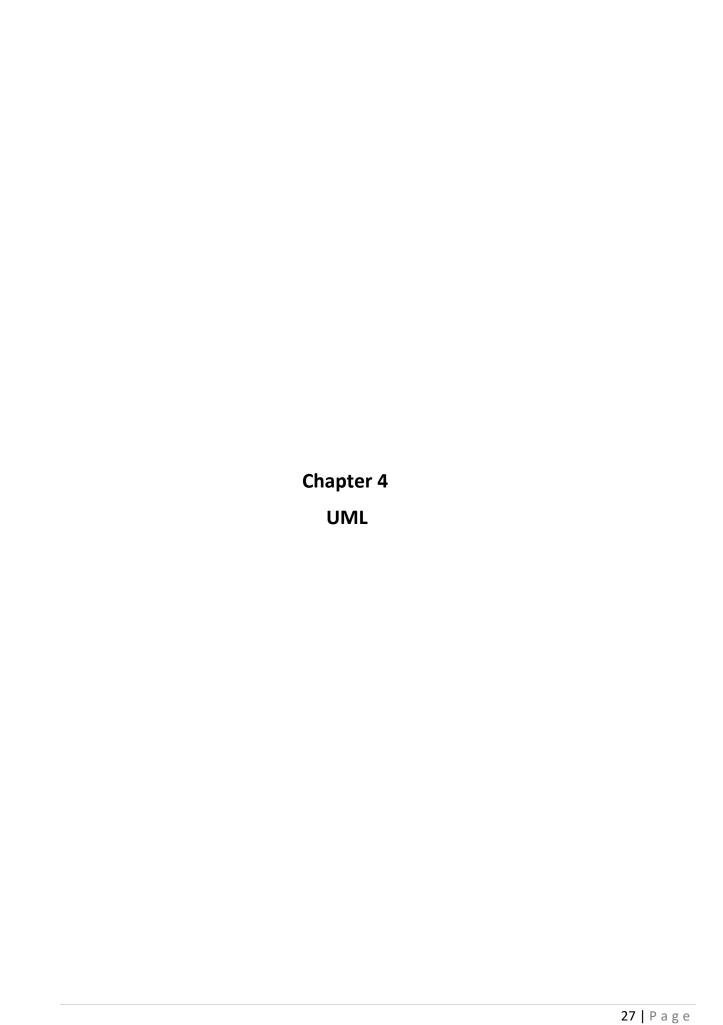
Date String	Time String
"25/3/2022"	"8"
"25/3/2022"	"8"
"25/3/2022"	"8"

Boking_Date Date
2022-03-24T10:29:01.391+00:00
2022-03-24T10:29:01.758+00:00
2022-03-24T10:29:01.941+00:00

Chapter 4

UML

4.1 Use case Diagram:-

Admin:-
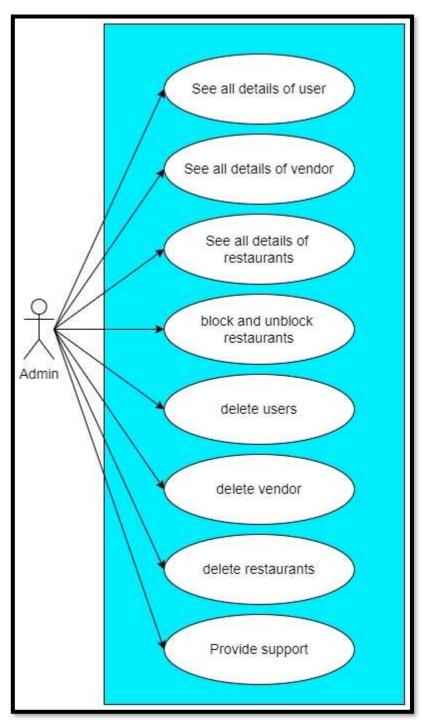
Figure 7

User:-
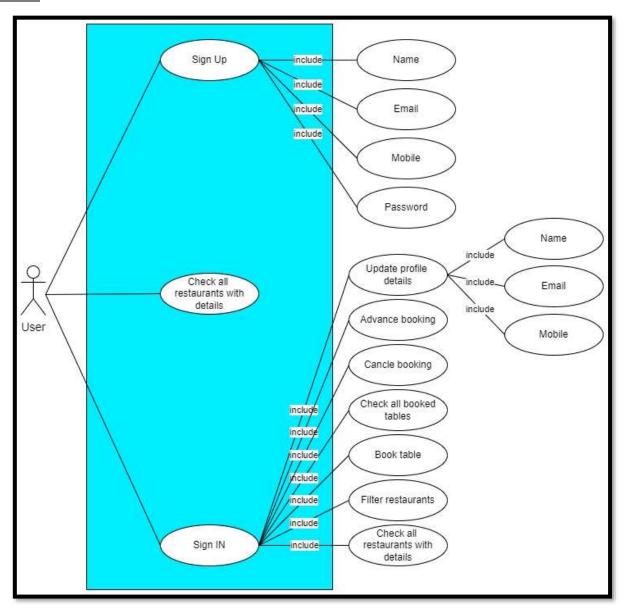
Figure 8

Vendor:-
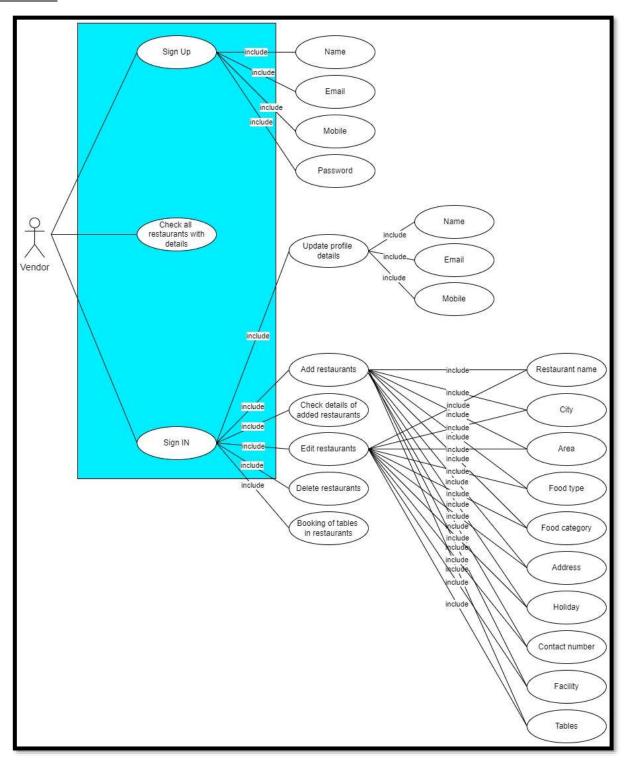
Figure 9

4.2 Sequence Diagram:-

Admin:-
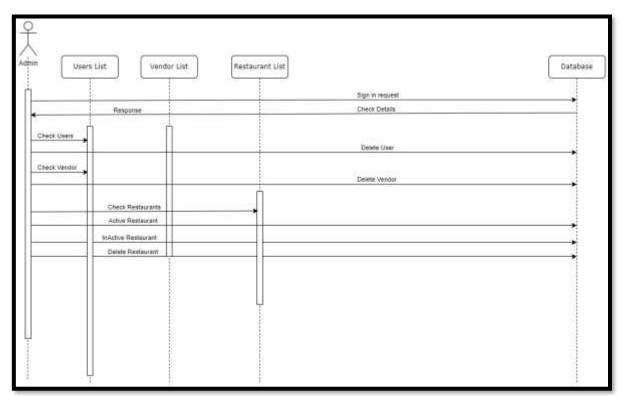
Figure 10

User:-
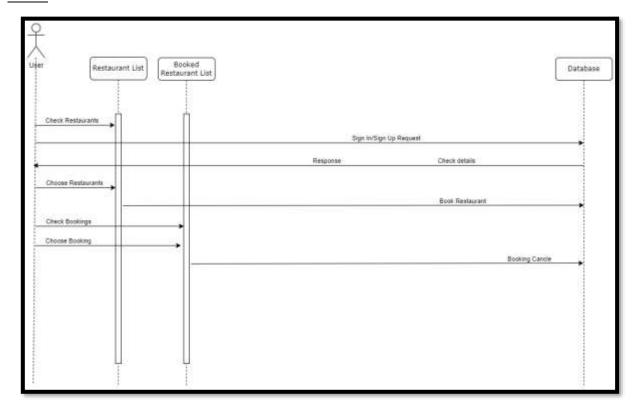
Figure 11

Vendor:-
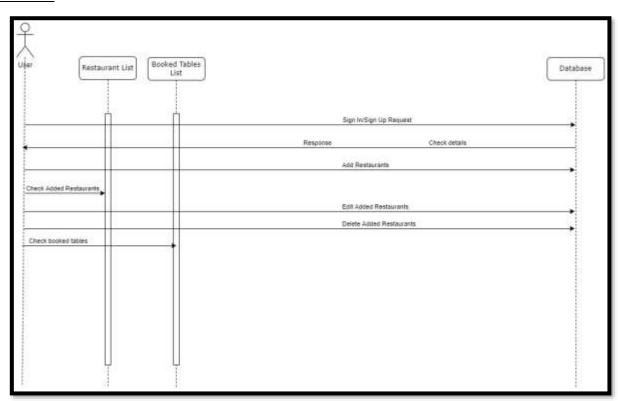
Figure 12

4.3 Activity Diagram:-

Admin:-
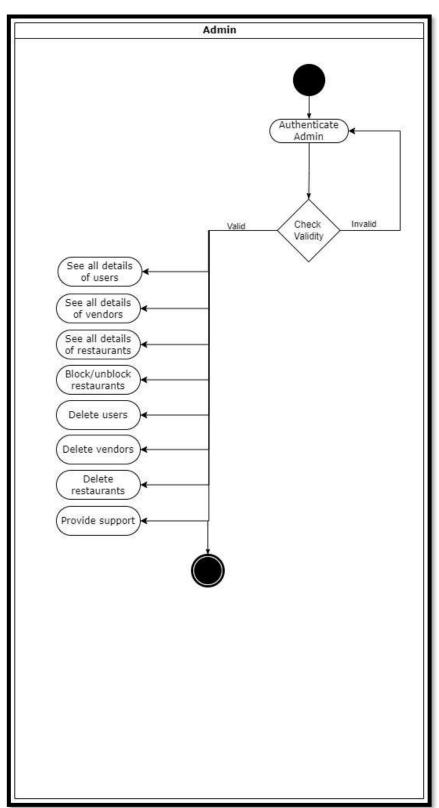
Figure 13

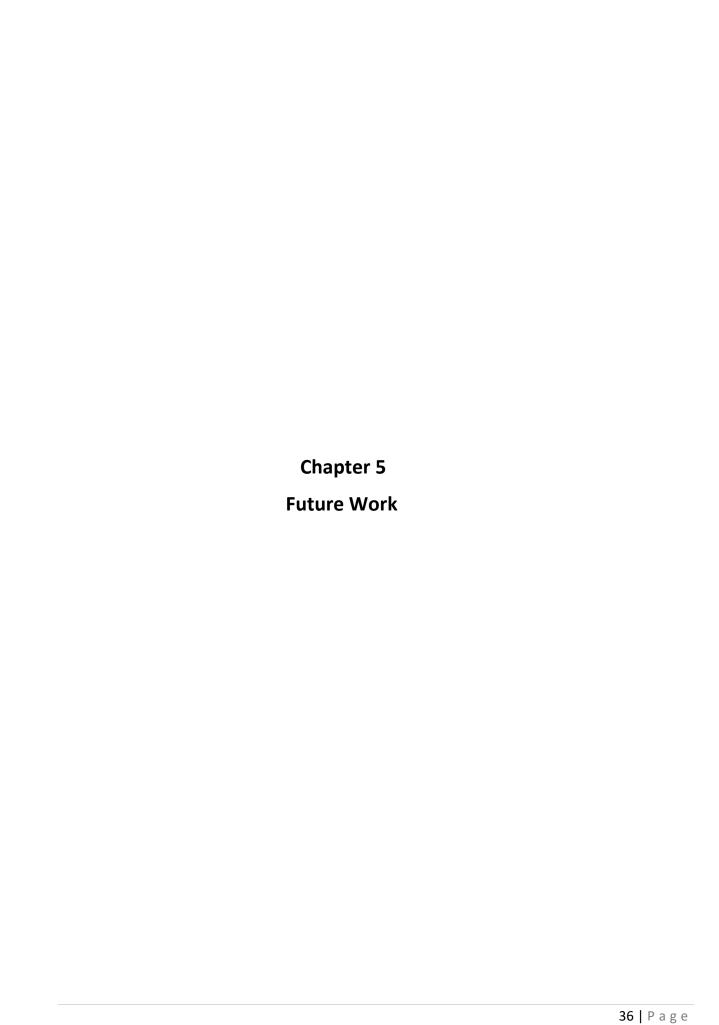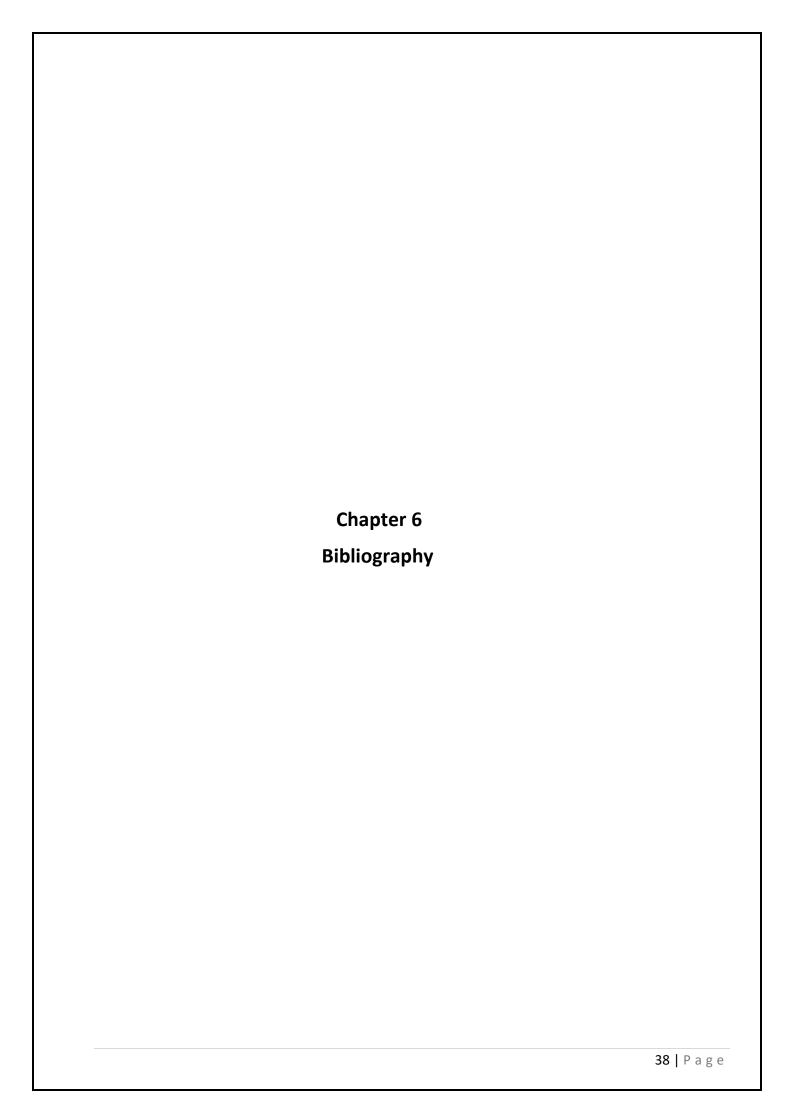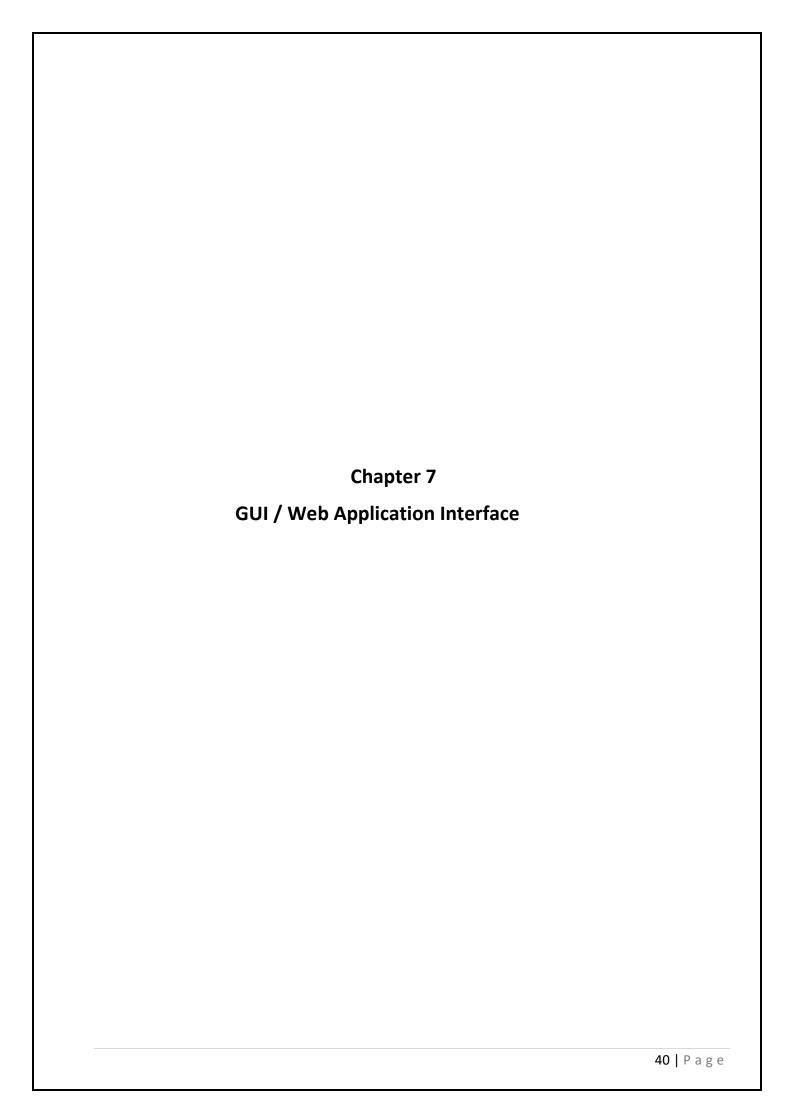User:-
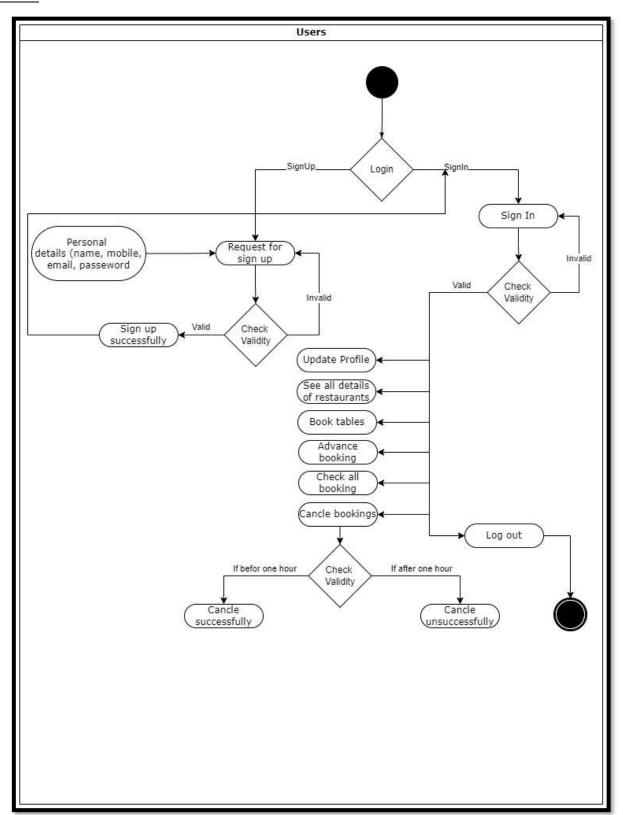
Figure 14

Vendor:-

Figure 15

Chapter 5

Future Work
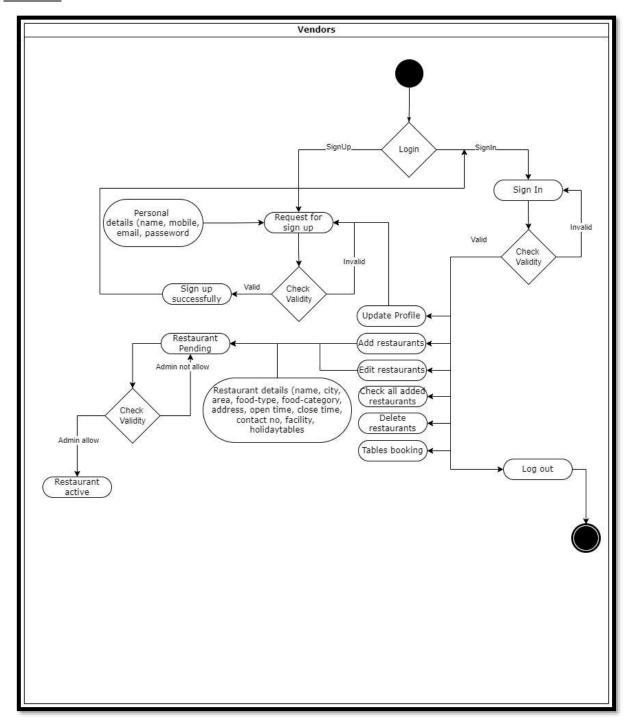
The following section discusses the work that will be implemented with the future release of software:-

1) In future work, we will add payment integration for booking tables in the restaurant because we have to purchase payment API for this functionality.

2) Facility for offers and coupons.

3) Time management issue at real time after booking at restaurant.

4) Provide a feature for forgetting password functionality.

5) Table and no of mapping.

Chapter 6

Bibliography

Reference:-

1) We take reference for the react from:-
- https://reactjs.org/docs/getting-started.html
- https://www.youtube.com/playlist?list=PLu0W_9lII9agx66oZnT6IyhcMIbUMNMdt

2) We take reference for the Node.js from:-
- https://nodejs.org/en/docs/
- https://www.youtube.com/playlist?list=PLwGdqUZWnOp00IbeN0OtL9dmnasipZ9x8

3) We take reference for the mongo Db from:-
- https://www.mongodb.com/docs/
- https://www.youtube.com/playlist?list=PLwGdqUZWnOp00IbeN0OtL9dmnasipZ9x8

4) We take reference for the express & npm from:-
- https://www.npmjs.com/package/express
- https://www.youtube.com/playlist?list=PLwGdqUZWnOp00IbeN0OtL9dmnasipZ9x8

5) We take reference for the designing (Bootstrap) from:-
- https://getbootstrap.com/docs/5.1/getting-started/introduction/

6) We take reference for the error solving(Stack overflow) from:-
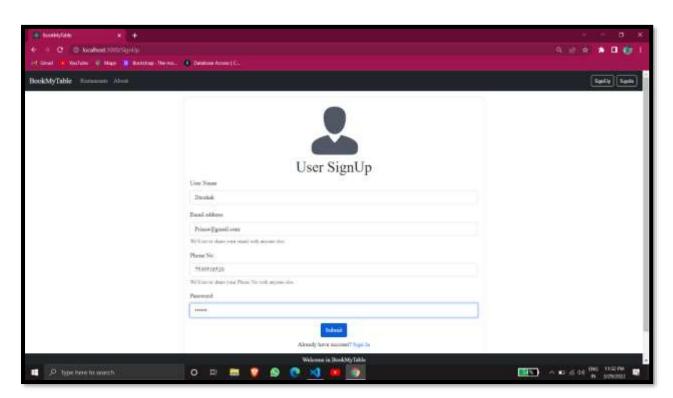- https://stackoverflow.com/

Chapter 7

GUI / Web Application Interface
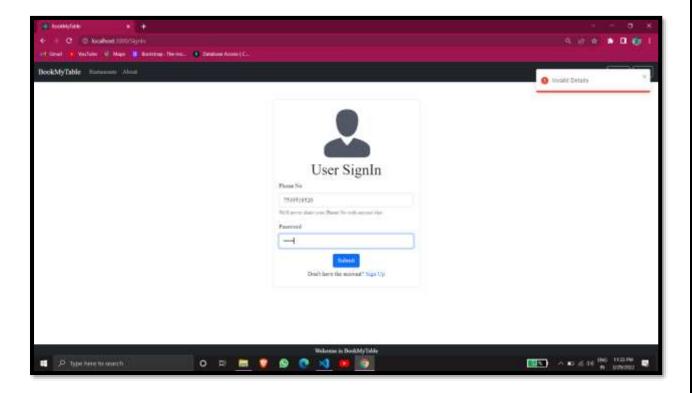
5.1 GUI design of web application:-

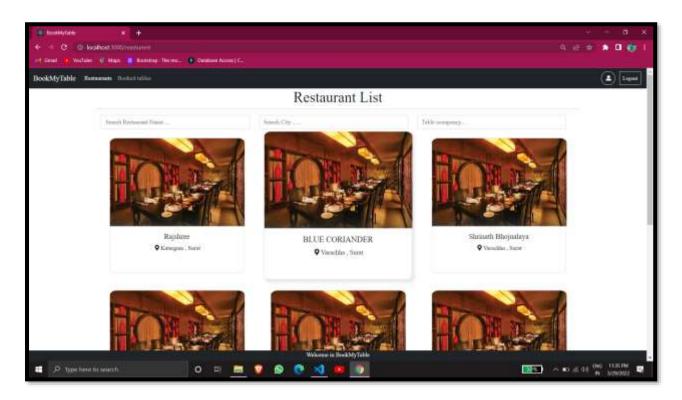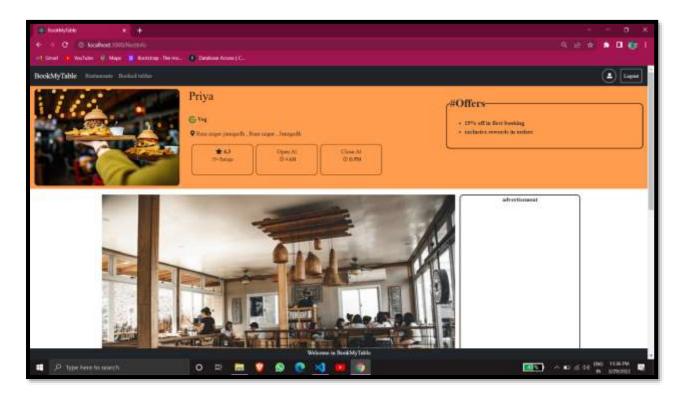➤ This is the home page of the website:-
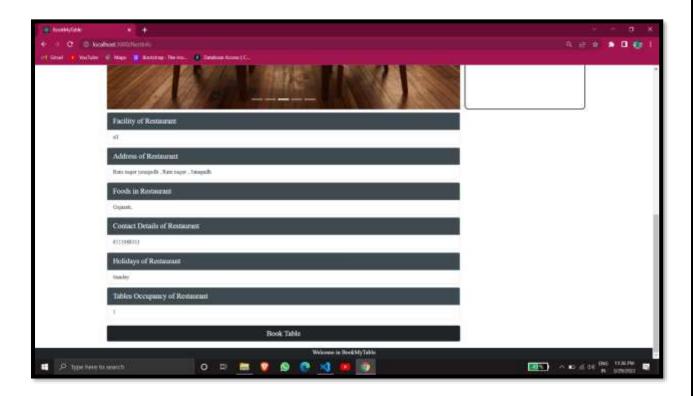

➤ User Sign Up:-

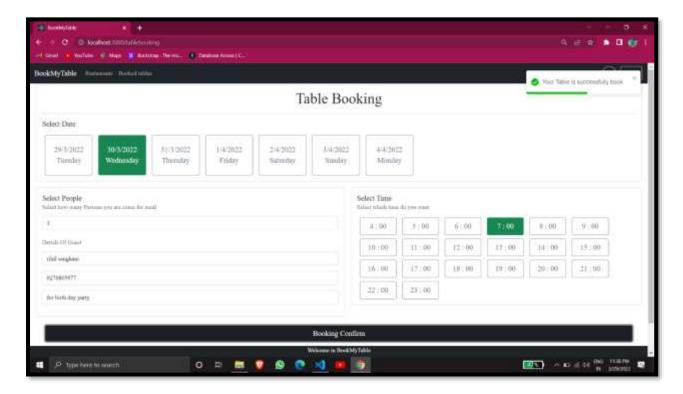➢ <u>User Sign In:-</u>

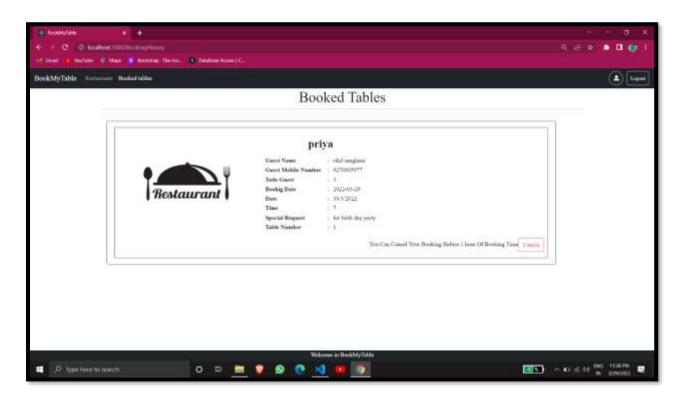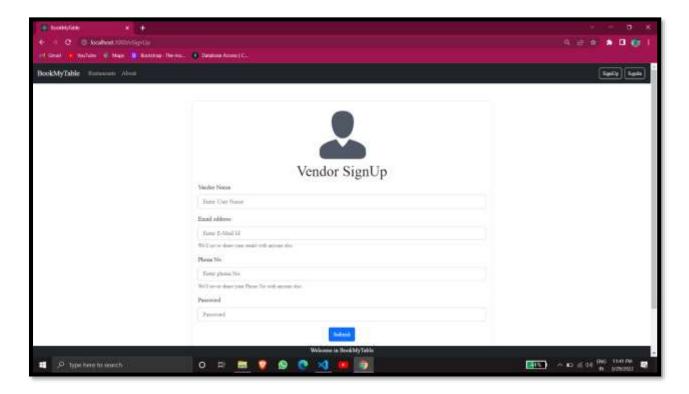➢ <u>Restaurant List:-</u>

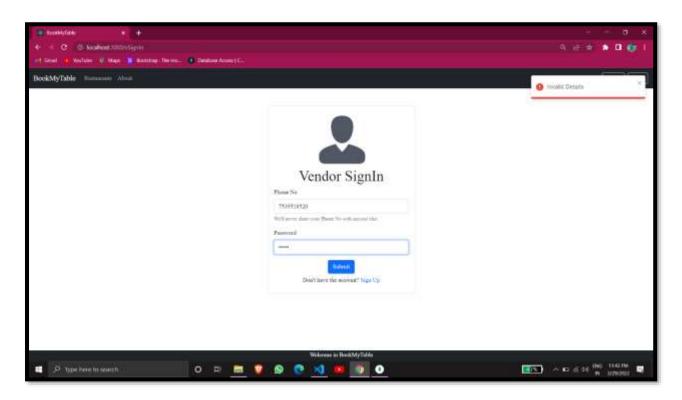## ➢ Restaurant Info:-

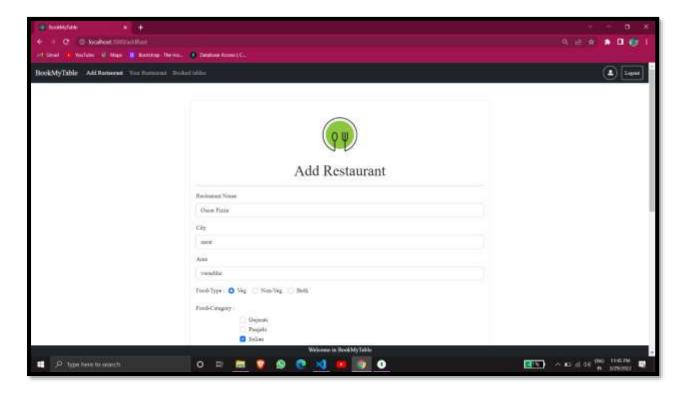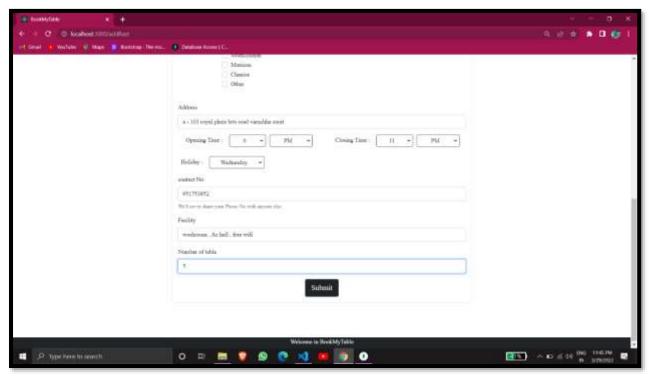➢ Restaurant's Table Booking:-

➢ User Table Booking History:-
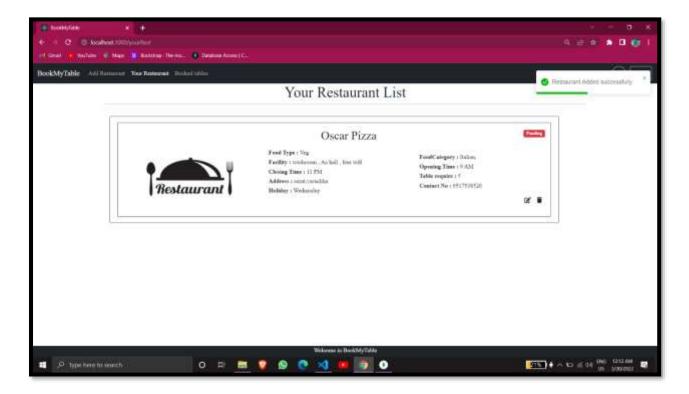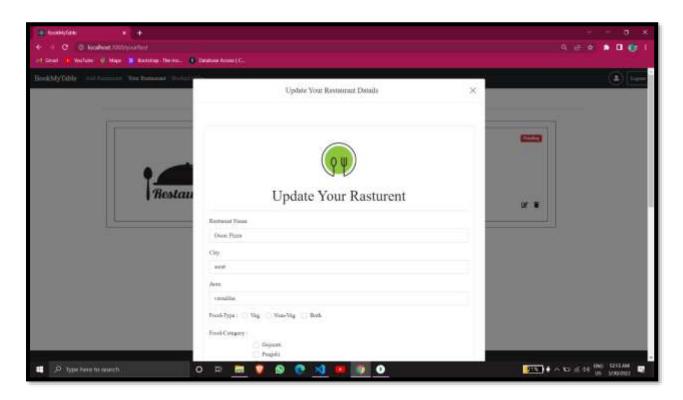
➢ Vendor Sign Up:-

➢ Vendor Sign In:-

➤ **Add New Restaurant By Vendor:-**
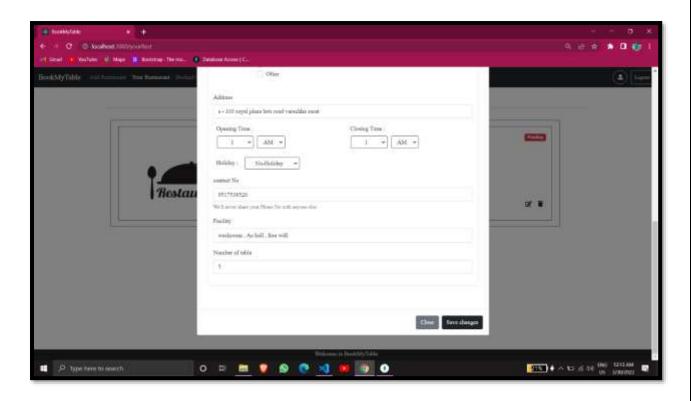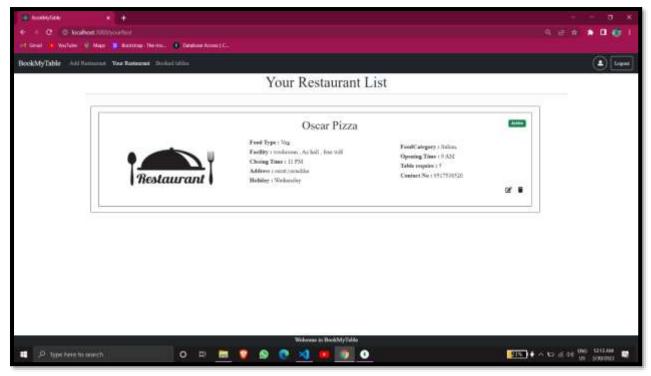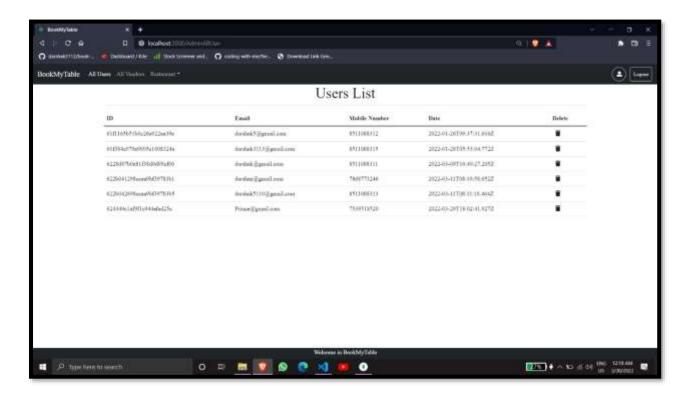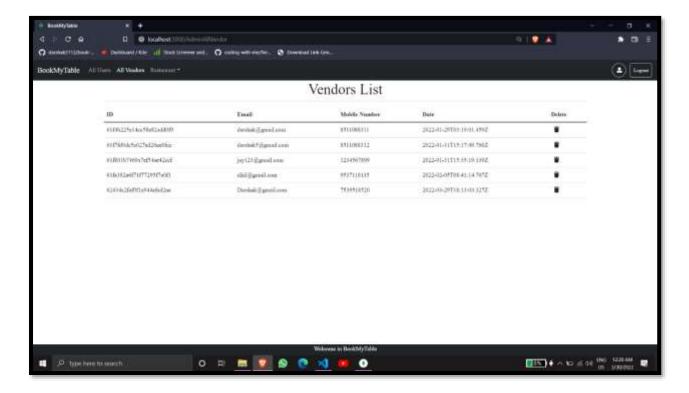
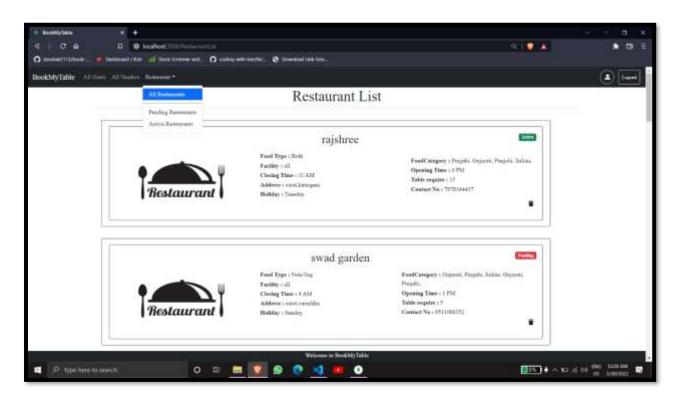➤ Vendor Restaurants list:-

➤ Update Restaurant:

❖ Admin site:-

➢ User List:-
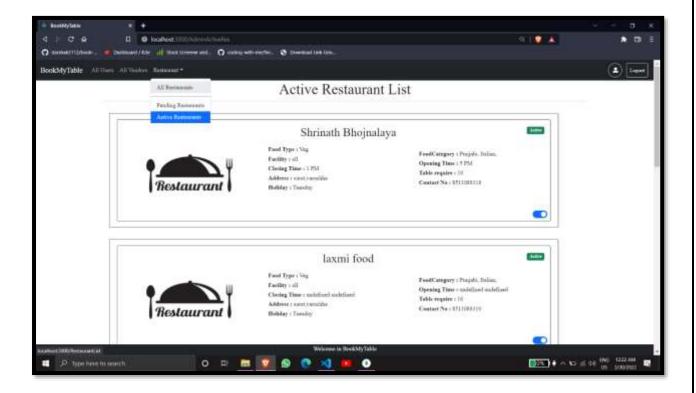
➤ Vendor List:-
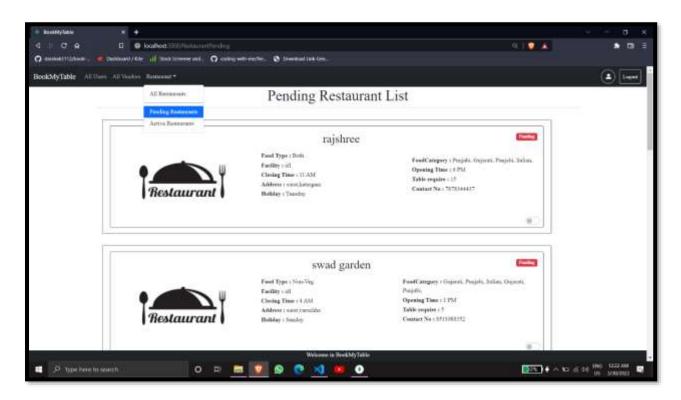
➤ All Restaurant:-
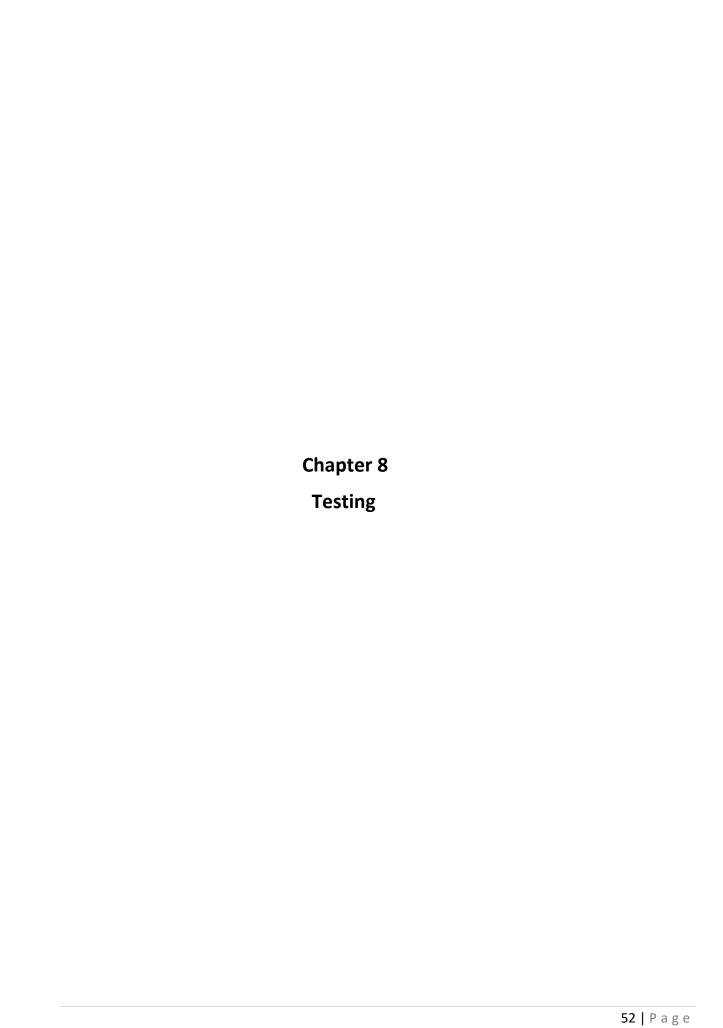
➢ Active Restaurant:-

➢ All Pending Restaurant:-

Chapter 8

Testing

8.1 Testing:-

Functionality Testing:-

➢ The main goal of functional testing is to make sure that all the functions within a web app are working smoothly without any technical glitches. In a web application, functional Testing could cover different things like whether all the links are working properly or not, testing forms in all the pages, validating HTML or CSS, testing database for the security, and so on.

➢ One should also ensure that test cases cover all the boundary conditions that need to be tested. Also, invalid inputs should give an appropriate error message.

Usability Testing:-

➢ When it comes to making the application user-friendly and effective, its user interface should comply with the standards. It is important that you follow all the global conventions and web standards while developing a web application. Usability testing is suitable for applications that are intended to streamline the manual process.

➢ However, one should also keep in mind certain important aspects like proper navigation, site map, avoiding using over-crowded content, and more while approaching usability testing.

Web UI Testing:-

➢ One of the most important interfaces within a web application is the web server and application server interface and database server interface. Web UI Testing will ensure that all the individual components within a web application are connected appropriately. One should check whether the interaction between these servers is executed properly or not with the help of this testing method.

Compatibility Testing:-

➢ The compatibility of your web application is one of the most crucial things you should consider while testing the application. Compatibility testing will check your website or web application for browser compatibility, operating system compatibility, and mobile browsing.

Performance Testing:-

➢ Performance testing will help you determine the performance of your web application under various scenarios. Performance testing usually involves stress testing, scalability testing, and load testing. In this testing method, the website is usually tested for its functionality on the different operating systems, hardware platforms, and more.

Security Testing:-

➢ This testing method is one of the most important ones for your web application as if data leaks or modifications are tolerable or not. It usually involves various things like testing the CAPTCHA for automating scripts logins, testing SSL for security measures, whether it is possible to access web directories or files directly or not, and so on.

Admin Test Cases:-

Sr. No:	Description	Test Data	Expected Output	Actual Output	Type of Testing
1.	Admin Login	By clicking on the Admin sign-in link it shall redirect to the admin sign-in page.	Redirect to admin sign-in page.	Redirect to admin sign-in page.	Web UI testing
2.	Admin Login Success	The first admin has to fill in the details then click on submit. If details are not correct it will see the error message as a toast.	If all details are correct admin will redirect to the admin panel.	If all details are correct admin will redirect to the admin panel.	Functionality testing, Usability testing, Security testing, Web UI testing
3.	Users List	If the Admin click on the 'All User' in the navbar admin will redirect to All user's information Page	Redirect to the All Users Information page.	Redirect to the All Users Information page.	Functionality testing, Usability testing, Web UI testing, Compatibility testing, Performance testing
4.	Remove User	If the admin wants to remove the user from the website. So the admin will click on the delete icon then the user will remove it.	The user removes from all users list.	The user removes from all users list.	Functionality testing, Web UI testing, Compatibility testing, Performance testing

5.	Vendors List	If the Admin click on the 'All Vendors' into the navbar admin will redirect to All Vendor's information Page	Redirect to the All Vendors Information page.	Redirect to the All Vendors Information page.	Functionality testing, Usability testing, Web UI testing, Compatibility testing, Performance testing
6.	Remove Vendor	If the admin wants to remove the vendor from the website. So the admin will click on the delete icon then the vendor will remove it.	The user removes from all vendors list.	The user removes from all vendors list.	Functionality testing, Web UI testing, Compatibility testing, Performance testing
7.	All Restaurants	If Admin clicks on the 'All Restaurants' into the navbar admin will redirect to All Restaurants information Page	Redirect to the All Restaurants Information page.	Redirect to the All Restaurants Information page.	Functionality testing, Usability testing, Web UI testing, Compatibility testing, Performance testing
8.	Active Restaurants	If the Admin click on the 'Active Restaurants' into the navbar admin will redirect to Active Restaurants information Page	Redirect to the Active Restaurants Information page.	Redirect to the Active Restaurants Information page.	Functionality testing, Usability testing, Web UI testing, Compatibility testing, Performance testing
9.	Change Restaurant's status Active to Pending	If Admin wants to change the restaurant's status from Active to Pending. So just click on the switch	Remove restaurant from active restaurant's list and add in pending restaurant's list and also gives the message	Remove restaurant from active restaurant's list and add in pending restaurant's list and also gives the message	Functionality testing, Usability testing, Web UI testing, Compatibility testing, Performance testing

| 10.	Pending Restaurants	If the Admin click on the 'Pending Restaurants' into the navbar admin will redirect to Active Restaurants information Page	Redirect to the Pending Restaurants Information page.	Redirect to the Pending Restaurants Information page.	Functionality testing, Usability testing, Web UI testing, Compatibility testing, Performance testing
11.	Change Restaurant's status Pending to Active	If Admin Wonts to change restaurant's status Pending to Active. So just click on the switch	Remove the restaurant from the Pending restaurant list and add it to the Active restaurant's list and also gives the message	Remove the restaurant from the Pending restaurant list and add it to the Active restaurant's list and also gives the message	Functionality testing, Usability testing, Web UI testing, Compatibility testing, Performance testing
12.	Logout	Click on the logout button which Is on to the navbar	It will give you a message and redirect you to the home page of the website	It will give you a message and redirect you to the home page of the website	Functionality testing, Usability testing, Web UI testing, Performance testing
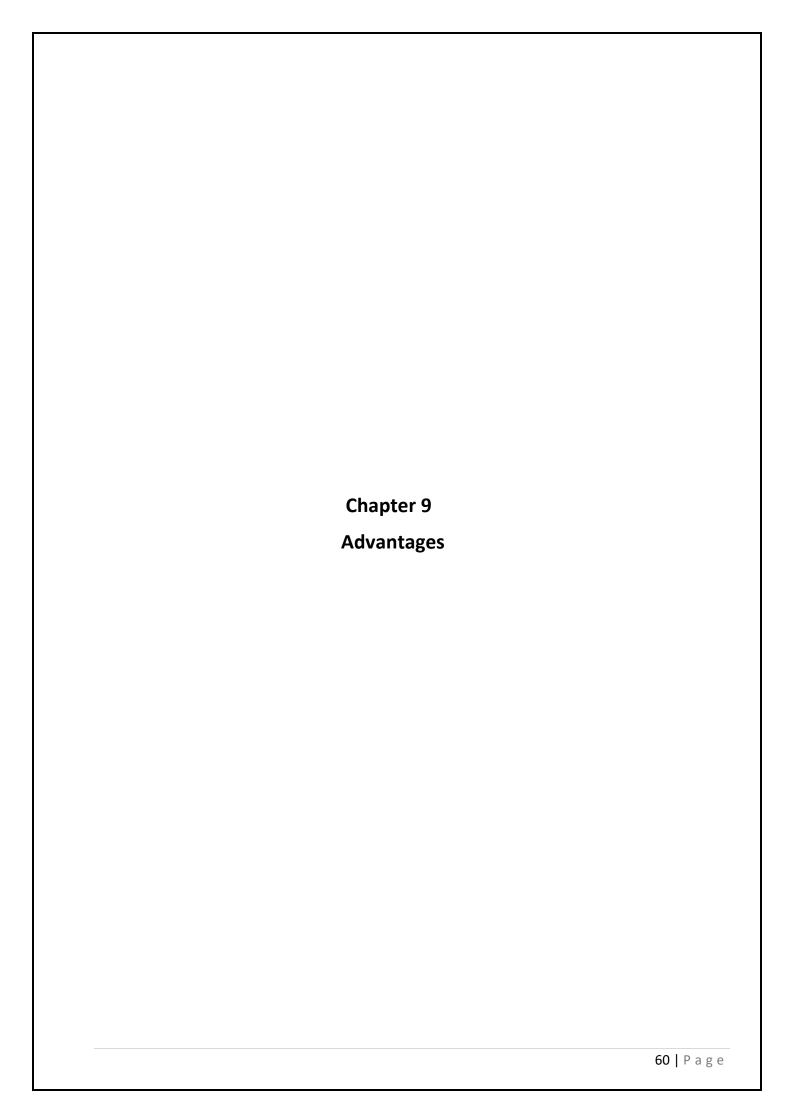
User Test Cases:-

Sr. No:	Description	Test Data	Expected Output	Actual Output	Type of Testing
1.	User Sign Up	Fill all required details then submit	It will give the message of sign up and redirect to the sign-in page.	It will give the message of sign up and redirect to the sign in	Functionality testing, Usability testing, Web UI testing, Performance testing
2.	User Sign In	Fill in all required details then submit	It will give the message of sign-in and redirect to the home page.	It will give the message of sign-in and redirect to the home page.	Functionality testing, Usability testing, Security testing, Web UI testing
3.	Restaurant List	Click on the Restaurants which is on the navbar	It will redirect to the Restaurants list page	It will redirect to the Restaurants list page	Functionality testing, Usability testing, Web UI testing, Compatibility testing, Performance testing
4.	Restaurant Info	Click on any restaurant	It will redirect to the restaurant's all information	It will redirect to the restaurant's all information	Functionality testing, Usability testing, Web UI testing, Performance testing
5.	Table Booking	Click on the Booking Button	It will redirect to the booking slots page then fill in all required details then click on the book it will give a message booking successful	It will redirect to the booking slots page then fill in all required details then click on the book it will give a message booking successful	Functionality testing, Usability testing, Web UI testing, Compatibility testing, Performance testing

| 6.	Booking History	Click on the Booked Table which is on the navbar	It will redirect to the booked table. There are users who can see all booking table	It will redirect to the booked table. There are users who can see all booking table	Functionality testing, Usability testing, Web UI testing, Compatibility testing, Performance testing
7.	Cancel Booking	Click on the cancel button	It will remove booking information from Booked table list while the user can cancel the booking before 1 hour of booking time and give a message	It will remove booking information from Booked table list while the user can cancel the booking before 1 hour of booking time and give the message	Functionality testing, Usability testing, Web UI testing, Performance testing
8.	Logout	Click on the logout button which Is on to the navbar	It will give you a message and redirect you to the home page of the website	It will give you a message and redirect you to the home page of the website	Functionality testing, Usability testing, Web UI testing,
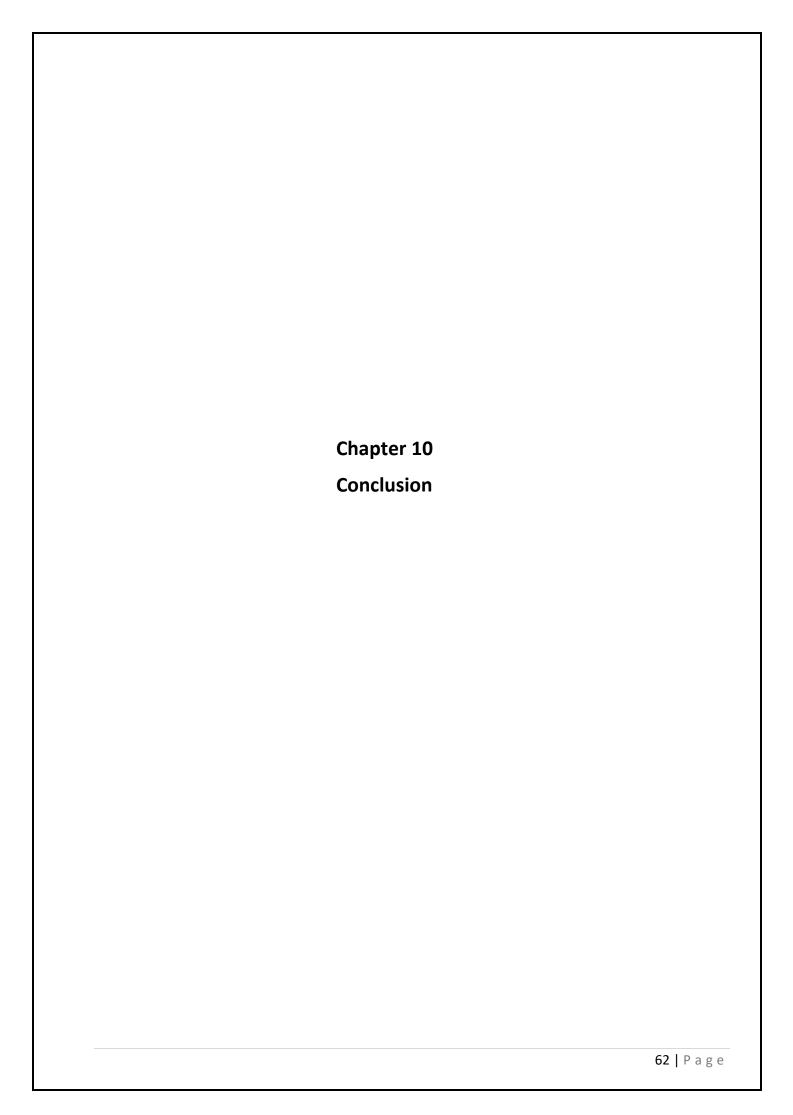
Vendor Test Cases:-

Sr. No:	Description	Test Data	Expected Output	Actual Output	Type of Testing
1.	Vendor Sign Up	Fill all required details then submit	It will give the message of sign up and redirect to the sign-in page.	It will give the message of sign up and redirect to the sign in	Functionality testing, Usability testing, Web UI testing, Performance testing
2.	Vendor Sign In	Fill in all required details then submit	It will give the message of sign-in and redirect to the home page.	It will give the message of sign-in and redirect to the home page.	Functionality testing, Usability testing, Security testing, Web UI testing
3.	Add Restaurant	Click on Add Restaurant which is on the navbar then fill in all required details and submit	It will give the message and redirect to the Your Restaurant	It will give the message and redirect to the Your Restaurant	Functionality testing, Usability testing, Performance testing, Web UI testing
4.	Your Restaurants	Click on the Your Restaurants which is on to the navbar	It will redirect to the Your Restaurant page	It will redirect to the Your Restaurant page	Functionality testing, Usability testing, Web UI testing, Compatibility testing, Performance testing
5.	Update Restaurant details	Go into the Your restaurant page there is an edit icon click on it then the model will lunch and the vendor can see restaurant information then change the information which is the vendor has to change then click on save.	It will give the message of updated restaurant information.	It will give the message of updated restaurant information.	Functionality testing, Usability testing, Performance testing, Web UI testing

Chapter 9

Advantages

Advantages:-

➢ Faster booking and as well as easy to find restaurants which you want.

➢ Easy to add restaurants for the vendor.

➢ Better management and time-saving.

➢ Better quality of services.

➢ Offers, coupons, and many more.....

Advantages:-

Chapter 10

Conclusion

Conclusion:-

➢ The Book My Table application when launched can change the life of people. The main goal of the web application is to provide table booking for the user. vendor can add their restaurant with simple tricks.

➢ Book My Table is to provide to book a table in a very small amount of time

➢ The user interface(UI) of the web application is so simple as the user and vendor can easily manage the system at their fingertips.