

**RESTRAW - A MODERN SOLUTION FOR RESTAURANTS**

**BY**

**MD. ABDULLAH AL MAHMUD**

**ID: 191-15-12316**

**MD. FAJLE RABBI ISLAM**

**ID: 191-15-12317**

**NISHAT ANZUM NOVA**

**ID: 191-15-12132**

This Report Presented in Partial Fulfillment of the Requirements for the Degree of  
Bachelor of Science in Computer Science and Engineering

Supervised By

**DR. SHEAK RASHED HAIDER NOORI**

Professor & Associate Head

Department of CSE

Daffodil International University

Co-Supervised By

**MD. JUEAL MIA**

Assistant Professor

Department of CSE

Daffodil International University



**DAFFODIL INTERNATIONAL UNIVERSITY**

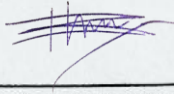
**DHAKA, BANGLADESH**

**JANUARY 2023**

## APPROVAL

This Project titled “**RESTRAW - A MODERN SOLUTION FOR RESTAURANTS**”, submitted by Md. Abdullah Al Mahmud (ID: 191-15-12317), Md. Fajle Rabbi Islam (ID: 191-15-12317) and Nishat Anzum Nova (ID: 191-15-12132) to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 26th January, 2023.

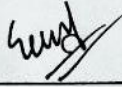
### BOARD OF EXAMINERS



**Dr. Touhid Bhuiyan**  
**Professor and Head**

Department of Computer Science and Engineering  
Faculty of Science & Information Technology Daffodil  
International University

**Chairman**



**Sazzadur Ahmed**  
**Assistant Professor**

Department of Computer Science and Engineering  
Faculty of Science & Information Technology Daffodil  
International University

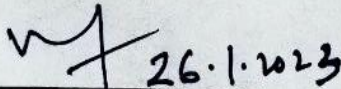
**Internal Examiner**



**Ms. Sharmin Akter**  
**Senior Lecturer**

Department of Computer Science and Engineering  
Faculty of Science & Information Technology Daffodil  
International University

**Internal Examiner**



**Dr. Ahmed Wasif Reza**  
**Associate Professor**

Department of Computer Science and Engineering  
East West University

**External Examiner**

## DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Dr. Sheak Rashed Haider Noori, Professor & Associate Head, Department of CSE** Daffodil International University. We also declare that neither this project nor any portion of this project has been submitted to any institution for the granting of a degree or certificate.

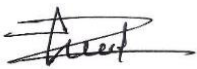
**Supervised by:**



---

**Dr. Sheak Rashed Haider Noori**  
**Professor & Associate Head**  
Department of CSE  
Daffodil International University

**Co-Supervised by:**



---


**Md. Jueal Mia**  
**Assistant Professor**  
Department of CSE  
Daffodil International University

**Submitted by:**



---

**Md. Abdullah Al Mahmud**  
ID: 191-15-12316  
Department of CSE  
Daffodil International University



---

**Md. Fajle Rabbi Islam**  
ID: 191-15-12317  
Department of CSE  
Daffodil International University



---

**Nishat Anzum Nova**  
ID: 192-25-12132  
Department of CSE  
Daffodil International University

## ACKNOWLEDGEMENT

First and foremost, we offer our heartfelt appreciation and gratitude to almighty God for His divine grace, which enabled us to successfully accomplish our final year project.

We really grateful and wish our profound indebtedness to **Dr. Sheak Rashed Haider Noori, Professor & Associate Head, Department of CSE**, Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the “Android Application Development” field to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project.

We would like to express our heartiest gratitude to **Dr. Touhid Bhuiyan, Professor and Head, Department of CSE**, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank every one of our Daffodil International University classmates who participated in this discussion while completing their coursework.

Finally, we must express our gratitude for our parents' unwavering support and patience.

## **ABSTRACT**

Bangladesh is a country with more than 60,000 restaurants currently. But as we still use maximum tradition way of restaurant management, we face some hassle. Like, if we want to go somewhere to eat then through google map, online food ordering app we can know about some of nearest restaurants but we cannot have idea about all restaurants or the best rated restaurants near him or in Bangladesh in just one place. If one person wants to go on a tour or want to visit somewhere, he can't preplan that where he can find best food in best budget for him because there is no application which shows you other cities restaurants in such a best way. Then again, when someone go to a restaurant, sometimes the food menu available in the table is not updated so he has to wait for the waiter for the updated menu to know whether the food he wants is available or not and to give order. Besides, to know about the bestselling or best rated food of the restaurant, the customer is dependent on waiter and their Facebook page if have any. Due to these problems, our time is wasted sometimes. So, after thinking, we plan to make a mobile application which will provide a customer best experience in restaurant fact by giving the solution of all the above problems.

Through this application one can easily find any restaurant in Bangladesh from anywhere and can see their menu and ratings and their offers. Besides, one can go to the restaurant and can scan the QR code to see their updated menu, their best rated food and all info and can order without waiting for the waiter. Moreover, it will create a bridge between customer and restaurant owner and will save time.

## TABLE OF CONTENTS

<b>CONTENTS</b>	<b>PAGE</b>
Board of examiners	i
Declaration	ii
Acknowledgments	iii
Abstract	iv
<b>CHAPTER</b>	
<b>CHAPTER 1: INTRODUCTION</b>	<b>1-3</b>
1.1 Introduction	1
1.2 Motivation	1
1.3 Objective	2
1.4 Expected Outcomes	2
1.5 Project Management & Finance	2
1.6 Report Layout	3
<b>CHAPTER 2: BACKGROUND</b>	<b>4-5</b>
2.1 Introduction	4
2.2 Related Works	4
2.3 Comparative Analysis	4
2.4 Scope of the problems	5
2.5 Challenges	5

<b>CHAPTER 3: REQUIREMENT SPECIFICATION</b>	<b>6-26</b>
3.1 Business Process Model	6
3.2 Requirement collection and Analysis	6
3.3 Use case Model and Description	9
3.4 Logical Data Model	25
<b>CHAPTER 4: DESIGN SPECIFICATION</b>	<b>27-39</b>
4.1 Front-End Design	27
4.2 Back-End Design	34
4.3 Interaction Design	39
4.4 Implementation Requirements	39
<b>CHAPTER 5: IMPLEMENT AND TESTING</b>	<b>40-42</b>
5.1 Implementation of Database	40
5.2 Implementation of Front-End Design	40
5.3 Implementation of Back-End Design	41
5.4 Testing Implementation	41
5.5 Test Result and Reports	42
<b>CHAPTER 6: Impact of Society, Environment and Sustainability</b>	<b>43-44</b>
6.1 Impact on Society	43
6.2 Impact on Environment	43
6.3 Ethical Aspects	44
6.4 Sustainability Plan	44

<b>CHAPTER 7: CONCLUSION AND FUTURE SCOPE</b>	<b>45-45</b>
7.1 Discussion and Conclusion	45
7.2 Scope for further development	45
<b>REFERENCES</b>	<b>46</b>



## LIST OF FIGURES

<b>Figures</b>	<b>Page No</b>
Figure 3.1.1: Business Process Model of Restraw.	6
Figure 3.3.1: Use Case Diagram for Restraw	9
Figure 3.4.1: Logical Data Model of Restraw	26
Figure 4.1.1: Customer Sign-Up Page	27
Figure 4.1.2: Verification Page	27
Figure 4.1.3: Customer Log-In Page	29
Figure 4.1.4: Home Page	29
Figure 4.1.5: QR Page	30
Figure 4.1.6: Restaurant's Home Page	30
Figure 4.1.7: Food details Page	31
Figure 4.1.8: Try list Page	31
Figure 4.1.9: Explore Restaurant Page	32
Figure 4.1.10: User Profile Page	32
Figure 4.1.11: Bar Page	33
Figure 4.2.1: Database of Restraw.	34
Figure 4.2.2: Login page of admin panel.	35
Figure 4.2.3: Dashboard page of admin panel.	36

Figure 4.2.4: Staff account list page of admin panel.	36
Figure 4.2.5: POS terminal page of admin panel.	37
Figure 4.2.6: Expense page of admin panel.	38
Figure 4.2.7: Sales report page of admin panel	38

# CHAPTER 1

## INTRODUCTION

### 1.1 Introduction

Restaurant is place where people go to eat different meals which are cooked by restaurant's chef and served by employees and customers pay for it. When we go to a tour, we usually eat our meals from restaurants there. Besides sometimes we hang out with our friends and spend a good time in a restaurant. So, it's an essential place for us which is available almost every city in our country. But problem is we can't see all restaurants, their menus, ratings on online in just one place. There are some food delivery applications which shows us some nearest restaurants according to our location but they don't show other locations restaurants and don't show us the updated menu so we cannot preplan with it before going to other location's restaurants. So, for that we need to search in different websites and Facebook pages or groups. Then there is some traditional way in restaurants like a physical menu board which are not updated in regular basis, then dependency on waiter to know about house dishes and to order.

So, regarding these issues, we have implemented a smart restaurant management system to come out from the traditional old way and make things smarter. Our proposed system makes easy to search best suitable restaurant in any place in Bangladesh. Through it we can scan QR code of a restaurant to see the updated menu and their best-rated food and can order. It will show the old customer different offers from the restaurant owner so that they became interest to come back again. Ultimately it will be beneficial for not only customers but also to for different restaurant owners.

### 1.2 Motivation

Every country has many restaurants and Bangladesh also has more than 60 thousand restaurants. Whatever, in foreign countries like America, the restaurant ordering and management system is so smart rather than Bangladesh. In Bangladesh, before going somewhere we have to face difficulties to search suitable restaurants and to know about best

rated food and updated menu and ordering have to wait for waiter to come like a traditional way. Besides the restaurant owners cannot reach to their customers again. These issues are hassle and sometimes time consuming.

We are doing this project to make bridge between customer and restaurant owner and to make a smart system which will much helpful and time saver and easy going for customers.

### **1.3 Objectives**

1. To develop a smart system, capable of displaying every restaurant in a given region.
2. To ensure that anybody in the country has access to any restaurant's digital menu.
3. To create an app that allows one customer to see other customers' ratings on the food and restaurant.
4. To ensure that customers may view the restaurant's updated menu by scanning their QR code.
5. To develop a system that will present many unknown establishments and their special offerings to clients.
6. To create an integrated point-of-sale system for all of the restaurants.
7. To create a smart system that allows customers to order on-site to save time.

### **1.4 Expected Outcomes**

1. Users from anywhere can see all of our country's restaurants and their foods, ratings.
2. User can scan the restaurant's QR code to see the updated menu and its website and others information.
3. Restaurant owner can see and manage his restaurants all information including overall growth.

### **1.5 Project Management and Finance**

There are many food delivery applications in our country which shows only nearest restaurants of current location. Whatever there is no application which will show all restaurants in Bangladesh all together. Again, some restaurants have their own application to see its food menu

and other info and to order but there is no application which has these features and will work like that for all restaurants. So, we discuss it with our respected Supervisor Sir and select some features which we can add.

For completing our project, we collect all data from our Supervisor Sir and followed his instructions. Besides we discuss it with some restaurants like Chillox, Coffee Time, Yummy Yummy, Safus. They give us their opinion and we updated it as their requirement. After we complete the project, we test our application with Yummy Yummy and Safus Restaurant. They will use it and will recommend to some of their customers then they will update us if there is anything we need to fix. Until now we didn't need any money.

## **1.6 Report Layout**

In our project report, we manufactured our contents in chapters such as in

Chapter 1 We have described the Introduction, Motivation, Objectives o and Expected outcomes of this project.

Chapter 2 We spoke about the background of our application, how we plan it, and why we choose this application.

Chapter 3 we talk about our project's required specifications.

Chapter 4 system design and implementation of requirements.

Chapter 5 is based on the implementation and testing of our application for error testing.

Chapter 6 is based on Impact on Society, Environment and Sustainability.

Chapter 7 is our final chapter where we discussed about future scope and conclusions of our application.

## **CHAPTER 2**

### **BACKGROUND**

#### **2.1 Introduction**

In this chapter, we will look at various project works that are relevant to our project idea and try to identify and explain their limits. Then we'll figure out what separates them from us. We'll also explain why our application, 'Restraw,' is the greatest. Finally, we will discuss our application's challenges.

#### **2.2 Related Works**

As our project idea is a little bit common, there are many related works that happened but we want to make a better application with uncommon and smart features than others. Let us point out some related works.

1. Food Panda [1]
2. Pathao
3. Foodfex
4. Smart Ordering System for Restaurants [2]
5. Foody – Smart Restaurant Management System and Ordering [3]

#### **2.3 Comparative Analysis**

We investigated various initiatives connected to our subject and attempted to learn about them. We examined numerous things and discovered many things such as accuracy, restrictions, algorithms, various characteristics, and so on. Now we will compare and contrast our project 'Restraw' with another project.

“Food Panda” [1] and Pathao: the main difference between our and these projects are they are food delivery app that shows us some nearest restaurant according to the user’s current location and they don’t show restaurants if you are not in the region where food panda give service. But our application will show all restaurants in Bangladesh. Besides, there are no Dine-in facilities in

food panda but in our project, there will be these facilities and also user can scan restaurant's QR code to see the updated menu and order.

"Foodfex" is also a food delivery app with a feature to see favorite restaurants and different restaurant information but still it doesn't have our project's features.

"Smart Ordering System for Restaurants "[2] in this project they implemented onsite dine-in ordering system but like our project, it can't show all restaurants in a country.

"Foody – Smart Restaurant Management System and Ordering" this project is to avoid confusion between orders, provide a better view of food and allow the customer to choose the menu according to their taste at a minimum. But it can't show all restaurants and it doesn't have the scanning QR feature. Besides, this also doesn't allow restaurant owner to send different offers to their old customers.

## **2.4 Scope of the Problem**

There is various development project on restaurant management system, various application on play store. No one can't create a application which will show all restaurants in a country and will provide Dine-in facilities for them and along with make a bridge between owner and customer. But our application can create and update it.

## **2.5 Challenges**

- i) This application can be used only in Bangladesh.
- ii) As there are already food delivery/ pick up the app and there are waiters in restaurant so how much the user will accept it that's a big question.
- iii) It needs so many entries to generate it properly.

## CHAPTER 3

### REQUIREMENT SPECIFICATION

#### 3.1 Business Process Model

Business process modelling (BPM) in business management and systems integration is the practice of describing an enterprise's operations so that they may be examined, modified, and automated. The business process model for this project is given below.

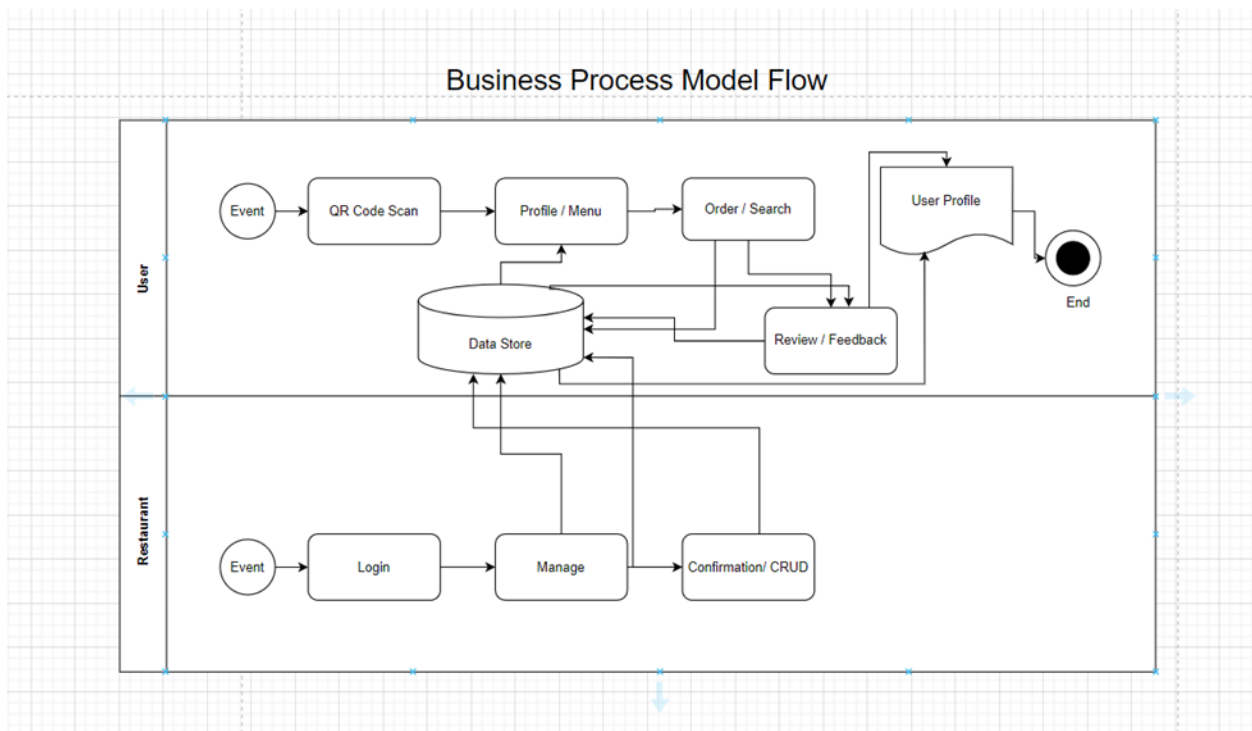


Figure 3.1.1: Business Process Model of Restraw.

#### 3.2 Requirement Collection and Analysis

Requirement collection and analysis is an important part and condition for developing any kind of application or system. Through this we can specify what are the requirements that the system or the application need, what we need to implement and on what requirements we need to focus.



We must specify two significant kind of requirements which is functional requirements and non-functional requirements. There is slightly different between them like functional requirements are the functions that system should have, what the software can do, its behavior. Besides non-functional requirements that specify the personality, quality, limitations of the software or application. For our system, we also have some functional and non-functional requirements.

### **Functional Requirements of the project are here:**

For Customer application: The customer needs to register and log in first to access the application's advantage that's why sign up and authentication log in is needed and these data of the user need to be stored in a database. Our project is mainly for those customers who need to see area base restaurants and see the restaurant's updated menu digitally and easily so the application has explored restaurants and view menu functions for customers. Besides customers can scan the QR code of the restaurant and can view the try list of any restaurant. Customers can create order, view order, give review and see ratings of the restaurant.

For the integrated point of sale system of restaurants: The restaurant's owner or manager needs to manage the information like adding, deleting, and updating the information of the staff, waiter, table, inventory, products, orders, and stocks so the system has these functions. The owner can make the expense report and can also see other reports of the restaurant and can specifically search from the data and can see the overall summary in the dashboard. Besides, the owner can print the invoice for any purchase or order.

The admin can manage the overall system, and its users like customers, and restaurant owners. They can delete or add any user and update the information.

### **The Non-Functional Requirements of the project are here:**

For our project, we need four kinds of non-functional requirements which are usability, maintainability, availability, and security requirements. To use the software easily and interactively, our system needs a user-friendly UI and UX and as there's a big part about managing of the restaurant and customers' different things then the modifying, addition, and maintenance should be easy. Then the customer application and owner's system should be

accessible and work properly all-time. Besides, as there is a system for restaurant owners where many important, confidential information about restaurants are stored and from where an owner can modify many information or data so the system must need to be secured. The username and password will be given only by the admin and only authorized users can get access.

### 3.3 Use Case Modeling and Description:

#### 3.3.1 Use Case Modeling:



Figure 3.3.1 Use Case Diagram for Restraw

### 3.3.2 Use Case Description:

#### 3.3.2.1 Sign Up

Use Case Name	Sign Up
Primary Actor	Customer
Secondary Actor	Admin
Pre-Condition	None
Post Condition	1. Account created successfully. 2. The server must store the sign-up information.
Description	When the sign-up form appears, the customer must give name, mobile no., and password to sign up. Then an authentication code will be sent to the given mobile no. and have to insert the code on the app, then the registration will be complete.

#### 3.3.2.2 Log In

Use Case Name	Log In
Primary Actor	Customer
Secondary Actor	Owner, Admin
Pre-Condition	The user should have an account or have a username and password.

Post Condition	<ol style="list-style-type: none"> <li>1. Successfully logged in to the system.</li> <li>2. Go to the home page of the system</li> </ol>
Description	When the user will give valid username/mobile no and password, he can be logged in to the system successfully, otherwise, it will show an invalid or bad credential message.

### 3.3.2.3 Scan QR

Use Case Name	Scan QR
Primary Actor	Customer
Pre-Condition	<ol style="list-style-type: none"> <li>1. Customer has to log in first.</li> <li>2. Must have a scanner on the system.</li> <li>2. Click on the scanner.</li> </ol>
Post Condition	Display the restaurant's menu and system.
Description	Every restaurant will have a unique QR to scan. If customer wants to see the menu and go to the restaurant's system then he can click on the QR scan button and scan the QR.

### 3.3.2.4 View Menu

Use Case Name	View menu
Primary Actor	Customer
Pre-Condition	Customer has to log in first.
Post Condition	N/A
Description	Customer can view the menu of any restaurant anytime.

### 3.3.2.5 Create Order

Use Case Name	Create Order
Primary Actor	Customer
Pre-Condition	<ol style="list-style-type: none"><li>1. Customer has to log in first.</li><li>2. Must have to scan the QR of the restaurant.</li><li>3. Must display the restaurant's site and menu</li></ol>
Post Condition	<ol style="list-style-type: none"><li>1. The order list will be generated.</li><li>2. The order will go to the restaurant's owner site.</li></ol>
Description	A customer can make on-site order after going to a restaurant, when he will go to the restaurant, he will go to the restaurant's site through scanning QR and can make order from the menu.

### 3.3.2.6 View Order

Use Case Name	View Order
Primary Actor	Customer
Secondary Actor	Owner
Pre-Condition	<ol style="list-style-type: none"><li>1. Customer has to log in first.</li><li>2. Must have to scan the QR of the restaurant.</li><li>3. Must display the restaurant's site and menu</li><li>4. Have to make order</li></ol>
Post Condition	N/A
Description	Customer can view the order he makes and the owner also can view the order customer makes to process it.

### 3.3.2.7 Explore Restaurants

Use Case Name	Explore Restaurants
Primary Actor	Customer
Pre-Condition	<ol style="list-style-type: none"><li>1. Customer has to log in first.</li><li>2. System must have maps.</li></ol>
Post Condition	Display restaurants in the area
Description	If a customer wants to find restaurants in his area he can find them on the map on the system and can explore restaurants.

### 3.3.2.8 View Try list

Use Case Name	View Try list
Primary Actor	Customer
Pre-Condition	1. Customer has to log in first. 2. Search and explore a restaurant
Post Condition	N/A
Description	A customer can see the popular or try food list of a restaurant.

### 3.3.2.9 Give Review

Use Case Name	Give Review
Primary Actor	Customer
Pre-Condition	1. The customer has to log in first. 2. Must have to scan the QR of the restaurant. 3. Must display the restaurant's site and menu 4. Have to make an order
Post Condition	Review will be stored on the server and will show on the system.
Description	After on-site ordering food and eating it, customer can give a review for the food and restaurant.



### 3.3.2.10 See Ratings

Use Case Name	See Ratings
Primary Actor	Customer
Pre-Condition	<ol style="list-style-type: none"><li>1. The customer has to log in first.</li><li>2. Select a restaurant and food item from the menu.</li><li>3. System must have the ratings for each food item.</li></ol>
Post Condition	N/A
Description	Customer can explore any restaurant and see the food menu and the ratings of each food item and can decide he will try it or not.

### 3.3.2.11 Check Profile Details

Use Case Name	Check Profile details
Primary Actor	Customer
Secondary Actor	Owner
Pre-Condition	<ol style="list-style-type: none"><li>1. The user must have an account</li><li>2. The customer has to log in first.</li></ol>
Post Condition	N/A
Description	Customer can see their profile and owner can see the restaurant's admin profile.

### 3.3.2.12 Manage Staff info

Use Case Name	Manage Staff Info
Secondary Actor	Owner
Pre-Condition	Must have to logged in to the system.
Post Condition	Updated information should be saved on the server successfully.
Description	Owner can add staff information to the system and also can update the information at later anytime.

### 3.3.2.13 Manage Waiter Info

Use Case Name	Manage Waiter info
Secondary Actor	Owner
Pre-Condition	Must have to logged in to the system.
Post Condition	Updated information should be saved on the server successfully.
Description	Owner can add waiter information like the waiter's code, name, mobile no., salary, and branch to the system and also can update the information later anytime.

### 3.3.2.14 Manage Table info

Use Case Name	Manage Table info
Secondary Actor	Owner
Pre-Condition	Must have to logged in to the system.
Post Condition	Updated information should be saved on the server successfully.
Description	Owner can add table information like table code, table number, and picture of the table to the system and also can update the information later anytime if needed.

### 3.3.2.15 Manage Payment Method

Use Case Name	Manage Payment method
Secondary Actor	Owner
Pre-Condition	<ol style="list-style-type: none"><li>1. Must have to logged in to the system.</li><li>2. The restaurant should have these payment methods.</li></ol>
Post Condition	Information should be saved on the server successfully.
Description	Owner can add, delete, and update the payment method for the restaurant.

### 3.3.2.16 Manage Currency

Use Case Name	Manage Currency
Secondary Actor	Owner
Pre-Condition	1. Must have to logged in to the system. 2. The restaurant must have to able to accept the currency
Post Condition	Information should be saved on the server successfully.
Description	Owner can add, delete, and update the currency for the restaurant.

### 3.3.2.17 Manage Inventory

Use Case Name	Manage Inventory
Secondary Actor	Owner
Pre-Condition	Must have to logged in to the system.
Post Condition	Information should be saved on the server successfully.
Description	Owner can manage all type of inventory things through the system like he can add any item and item's category, unit, material information. Besides also can update the stock, minimum stock quantity to alert, and sales rate of the item. Then if any item needs then he can purchase the item by selecting supplier and payment and can insert and update this information in the system.

### 3.3.2.18 Manage Product

Use Case Name	Manage Products
Secondary Actor	Owner
Pre-Condition	<ol style="list-style-type: none"><li>1. Must have to logged in to the system.</li><li>2. The item list, category, product list should be available first.</li></ol>
Post Condition	Information should be saved on the server successfully.
Description	Owner can add, update every food item's recipe. Then he also can assign any staff to make any food and can add and update that information.

### 3.3.2.19 Receive Orders

Use Case Name	Receive Orders
Secondary Actor	Owner
Pre-Condition	<ol style="list-style-type: none"><li>1. Must have to logged in to the system.</li><li>2. The customer must have to order.</li><li>3. The order information from the customer should come to the owner's site.</li></ol>
Post Condition	Stored the info on the server.
Description	When a customer will order any food then the order list and information will transfer to the owner's site and the owner can receive the order.

### 3.3.2.20 Manage Orders

Use Case Name	Manage Orders
Secondary Actor	Owner
Pre-Condition	<ol style="list-style-type: none"><li>1. Must have to logged in to the system.</li><li>2. The customer must have to order.</li><li>3. The order information from the customer should come to the owner's site.</li></ol>
Post Condition	Should be stored all the information
Description	The owner can manage the order by adding, deleting, updating as per the customer's request or necessity.

### 3.3.2.21 Update Expenses

Use Case Name	Update Expenses
Secondary Actor	Owner
Pre-Condition	<ol style="list-style-type: none"><li>1. Must have to logged in to the system.</li><li>2. The staff list and payment methods should be available in the system.</li></ol>
Post Condition	Information should be saved on the server successfully and generate a list.
Description	Owner can add expense information of the restaurant and also can update and delete the information. Besides he can print the money receipt of the expense.

### 3.3.2.22 Stock Adjust

Use Case Name	Stock Adjust
Secondary Actor	Owner
Pre-Condition	1. Must have to logged in to the system. 2. The staff list and item list should be available in the system.
Post Condition	Should stored the information and generate a list.
Description	The owner can adjust the stock of the items of the restaurant and can add and update the adjustment information.

### 3.3.2.23 View all reports

Use Case Name	View all reports
Secondary Actor	Owner
Pre-Condition	Must have to logged in to the system.
Post Condition	N/A
Description	Owner can see all the reports of the restaurant like Sales report, Sale summary report, Purchase report, Income report, and Expense report. Besides, owner can search from the report and can see reports in specific term.

### 3.3.2.24 Printing Invoice

Use Case Name	Printing Invoice
Secondary Actor	Owner
Pre-Condition	1. Must have to logged in to the system 2. Should store the payment information. 3. Printer should be available.
Post Condition	Should print the invoice and store the invoice information in the server
Description	Owner can make invoice from the customer's order and from the expenses of the restaurant and also can print the invoice

### 3.3.2.25 Search

Use Case Name	Search
Secondary Actor	Owner
Pre-Condition	Must have to logged in to the system.
Post Condition	Search item will show
Description	The owner can do search in some cases like can search branch, can search items, can search from any list or in any reports of the system.



### 3.3.2.26 Change password

Use Case Name	Change password
Secondary Actor	Owner
Pre-Condition	Must have to logged in to the system.
Post Condition	New password will be saved in the server.
Description	The owner can change the restaurant's site's password anytime by filling up the form's current password, new password, and confirming password field.

### 3.3.2.27 View Restaurant's Summary Statistics

Use Case Name	View restaurant's summary statistics
Secondary Actor	Owner
Pre-Condition	Must have to logged in to the system.
Post Condition	N/A
Description	The owner can see the restaurant's overall summary like the best customer, best waiter, top selling items, today's sale, collection, total users, total items, Statistics of monthly wise sales and transactions, etc.

### 3.3.2.28 Manage Users

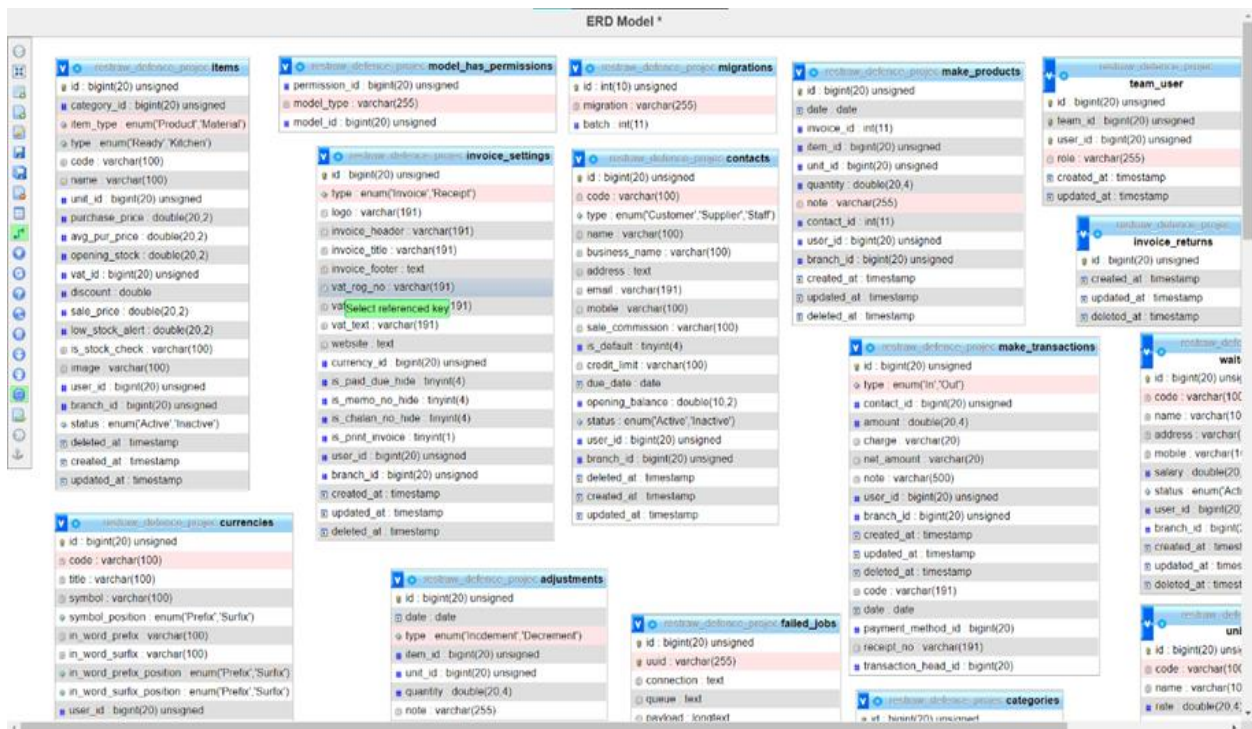
Use Case Name	Manage Users
Secondary Actor	Admin
Pre-Condition	Must have to logged in to the system.
Post Condition	Updated information should be saved on the server successfully and update the user's access accordingly
Description	Admin can manage user and user access like he can remove or block any user.

### 3.3.2.29 Manage Owner

Use Case Name	Manage Owner
Secondary Actor	Admin
Pre-Condition	Must have to logged in to the system.
Post Condition	Updated information should be saved on the server successfully and update the owner's access accordingly.
Description	Admin can manage the owner, owner's information, and owner's access like he can block any owner restaurant's access, he can add a new restaurant owner and can provide a username and password for the restaurant's owner site, can update their information.

### 3.4 Logical data model (ERD)

An entity-relationship diagram (ERD) depicts the relationships between entity sets in a database. In this sense, an entity is an object, a data component. An entity set is a group of entities that are comparable. These entities can have properties defined via attributes. The Entity-Relationship Diagram (ER diagram) is shown in the logical data model. A data entity is an item or notion about which data is maintained. The partial logical data model of this project is given below.



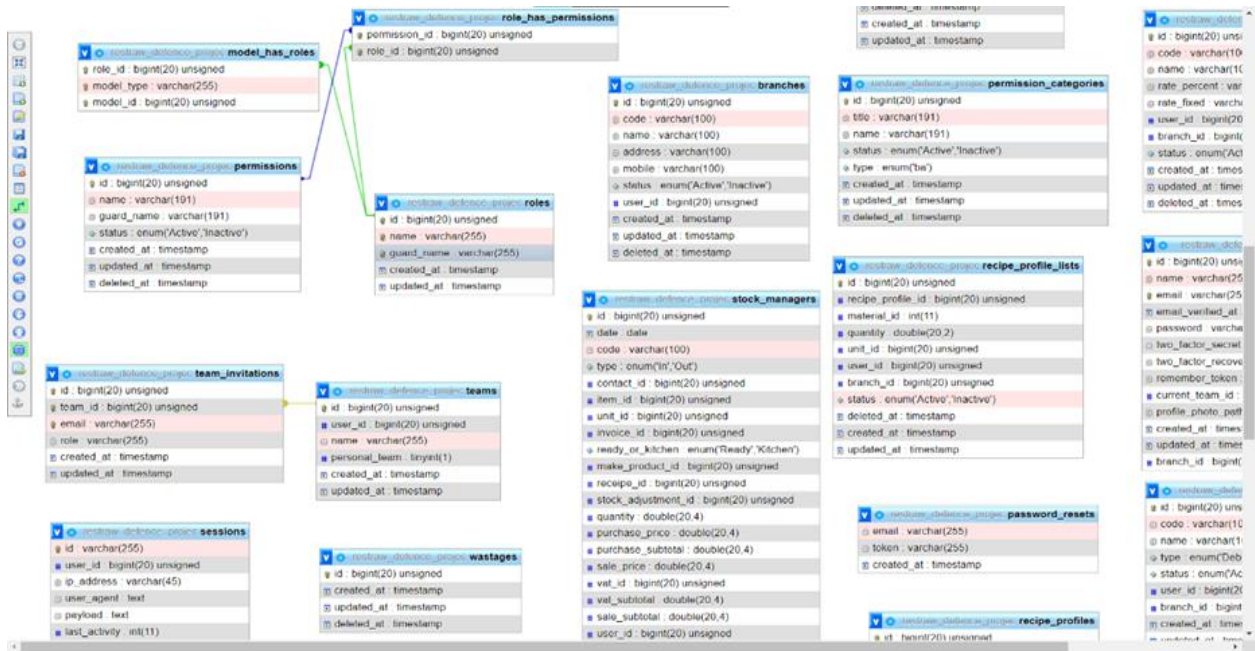


Figure 3.4.1: Logical Data Model of Restraw

# CHAPTER 4

## DESIGN SPECIFICATION

### 4.1 Front-end Design

Front-end design is the design of client sites which will be visible to the clients of our project. In front-end design, we need to think about both user-friendly user interfaces and logically correct operations. In our project, API do the main work and transfers the data from the backend to the frontend. Through XML files we have designed this overall easy-to-use and simple and nice user interface with the addition of several essential elements like buttons, icons, and text area.

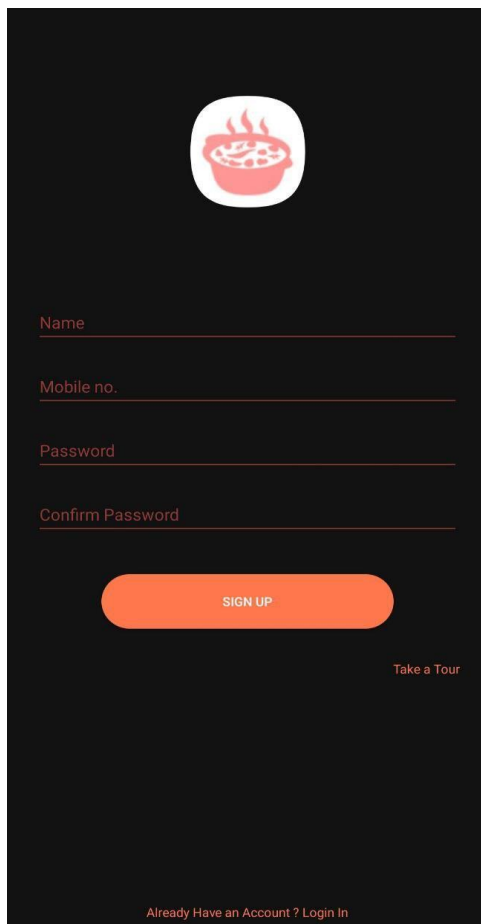


Figure 4.1.1: Customer Sign-Up Page

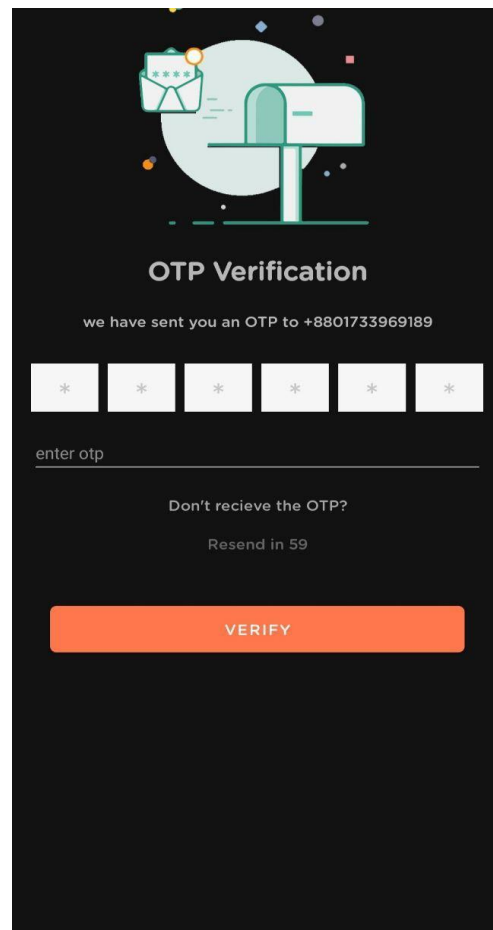


Figure 4.1.2: Verification Page

Figure 4.1.1 shows the customer sign up page. When a new user customer will install and open the application this page will come. If the customer doesn't have any account before than he has to sign up first. For sign up, he has to fill up the form with name, mobile no., password, and confirm password. Then after pressing the 'sign up' button the figure 4.1.2 page will show up. Besides, before sign up they can take a tour of the application and see the features if he wants by pressing the 'Take a tour' option. If he has an account before than he has to press the 'Log in' option which is in the below of the page and it will go to figure 4.1.3 log-in page.

Figure 4.1.2 shows the new customer's OTP verification page. When a new user press sign up button then this page will appear and a verification code will be sent to the given phone number which he need to submit correctly on the 'enter otp' section. If there's any need to resend the code than he has to press on resend option after the 60-second countdown. After submitting the code he needs to press the 'verify' button than if the code is correct then the account will be verified and created. However, the customer doesn't get any code or faces a problem in getting the code than he has to press on the option 'Don't receive the OTP?' option.

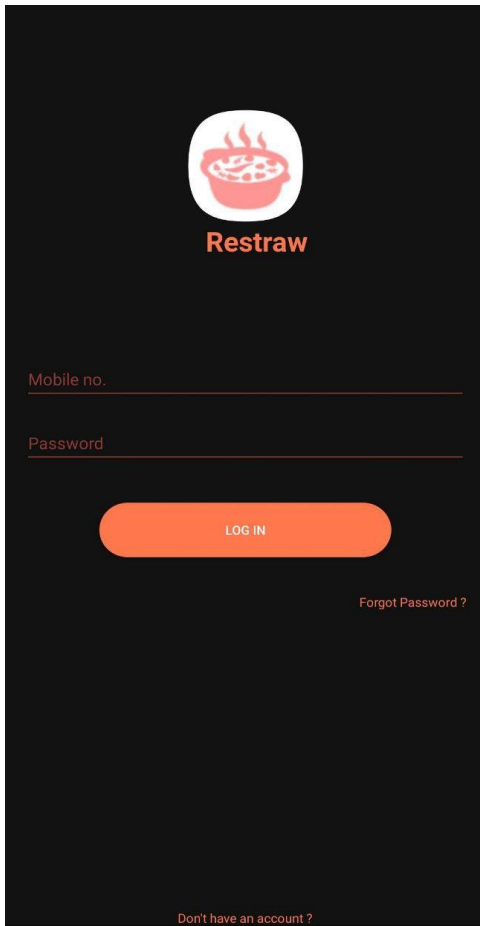


Figure 4.1.3: Customer Log-In Page

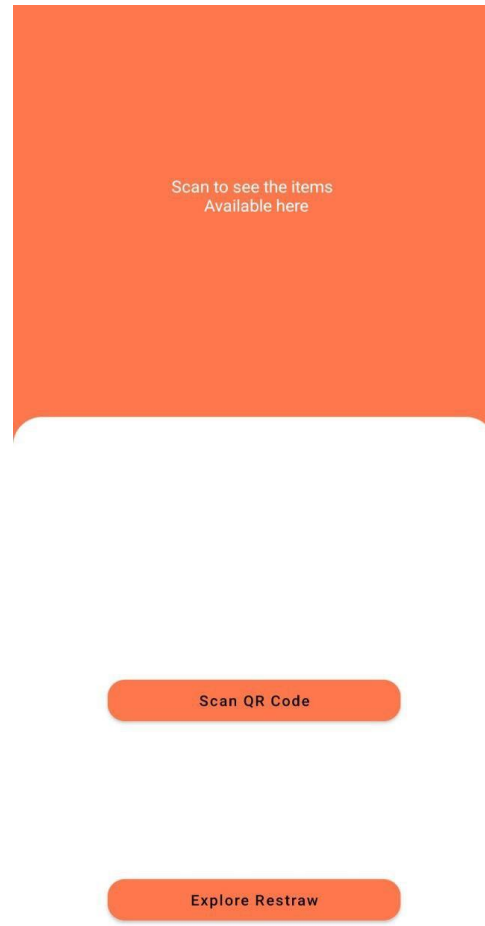


Figure 4.1.4: Home Page

Figure 4.1.3 shows the customer log in page. If the user already has an account and presses on log-in option then this page will show up. Here, the user needs to give the mobile no. and password which he has given at registration time and press on ‘Log In’ button. If he forgets the password then he has to press on the ‘Forgot password?’ option and if he doesn’t have an account then he has to press on the ‘Don’t have an account?’ option.

Figure 4.1.4 shows the home page of the customer application. When the user will login to the application, this home page will appear where the ‘Explore Restraw’ and ‘Scan QR Code’ options will be there. If someone wants to randomly explore restaurants in his area then he will press on the ‘Explore Restraw’ button and if someone goes to a restaurant and wants to scan the restaurant’s menu’s QR then he will press on the ‘Scan QR Code’ button.



Figure 4.1.5: QR Page

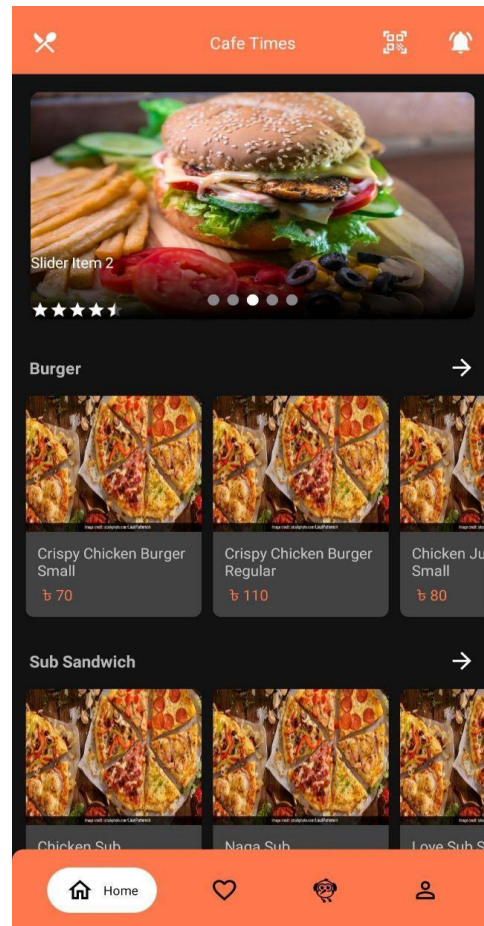


Figure 4.1.6: Restaurant's Home Page

Figure 4.1.5 shows the QR scanner page from where the customer can scan the restaurant's QR after going to a restaurant and can access the restaurant's menu and other functions.

Figure 4.1.6 shows the restaurant's home page where the restaurant's menu will be displayed. There will be shown many options to operate like try list, explore, profile, qr scan, notification, and menu bar options. In the beginning of the restaurant's home page, there will be shown a slide display where five popular items will be displayed sequentially. Besides customer can see the rating of the restaurant in this page.



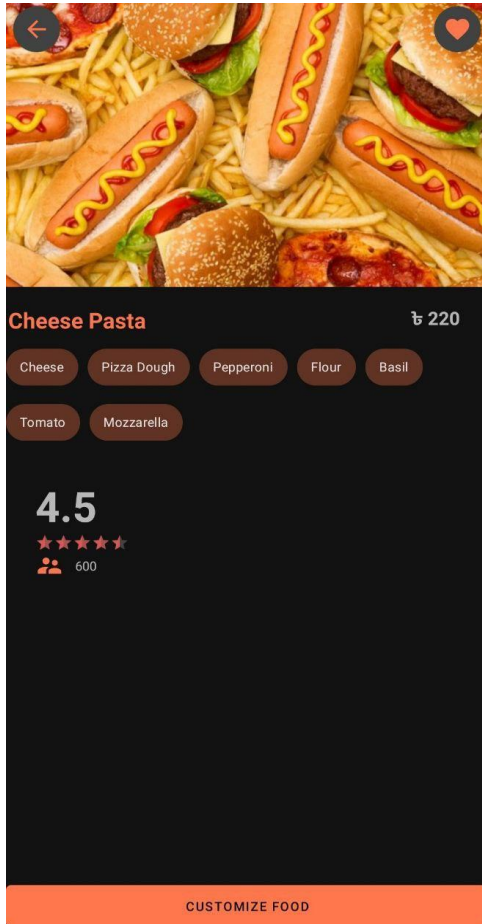


Figure 4.1.7: Food details Page

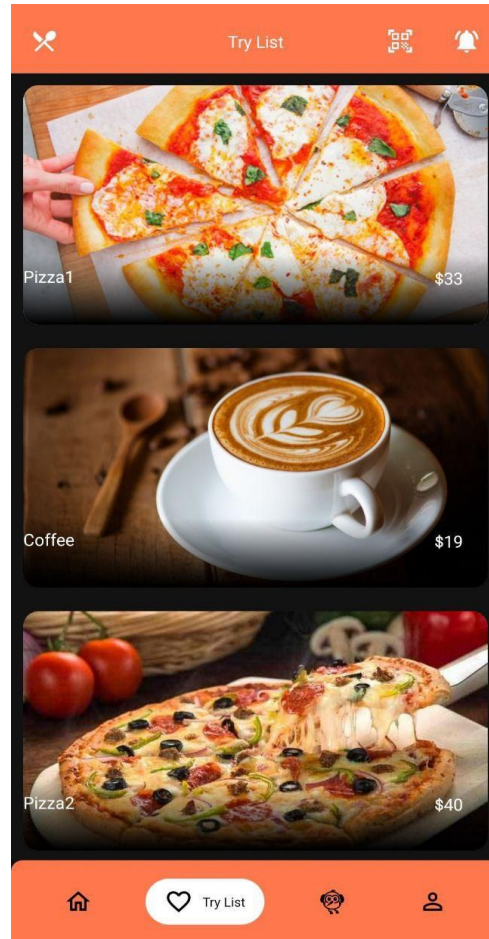


Figure 4.1.8: Try list Page

Figure 4.1.7 shows the any specific food details page. When a customer will click on any food from the restaurant's menu this page will show up and, in this page, customer can see the ratings of the food and how many people rate on it, the price of the food, picture of the food, details of the food.

Figure 4.1.8 shows the try list page of the restaurant. In a restaurant, which foods are best-selling or best rated or popular those food will be shown up in this page so that a customer can easily decide and choose the food.



Figure 4.1.9: Explore Restaurant Page

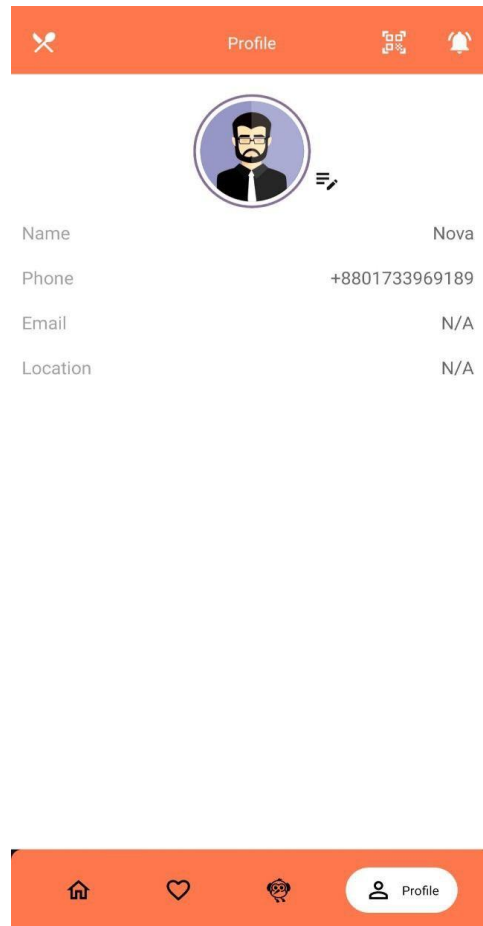


Figure 4.1.10: User Profile Page

Figure 4.1.9 shows the map page from where a user can find restaurants in his area, even in other areas. When someone press the ‘Explore Restraw’ or ‘Tour’ button this map page will show up.

Figure 4.1.10 shows the customer’s user profile page. User name, phone no, email, location, and profile picture can be seen from here. Besides user can edit the information if he wants.

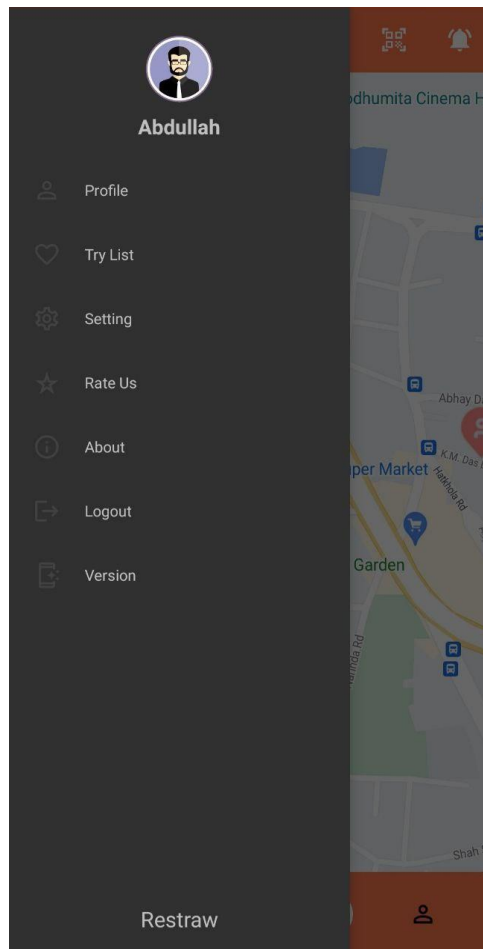


Figure 4.1.11: Bar Page

Figure 4.1.11 shows up the menu bar page of the application. In the top left corner of the application, there is a knife-fork sign which indicates the menu bar of the application. After clicking on the sign, the user can see different options of the application as a menu like profile, try list, settings, rate the application, about the application, log out, and version of the application. One can choose any option as he wants to do like if someone wants to log out from the application then he can choose and press the log out option.

## 4.2 Back-End Design

Restraw is an Android app, hence we need a robust backend for operations. The database is the most important component since all tasks are conducted in accordance with the database. For the Restraw, we utilized both SQL and NoSQL databases. MySQL was chosen because of its speed, dependability, availability, scalability, and security. We used a MySQL database for the Android application as well as a web-based admin panel. For the authentication of the Android application, we also employed a NoSQL database (Firebase).

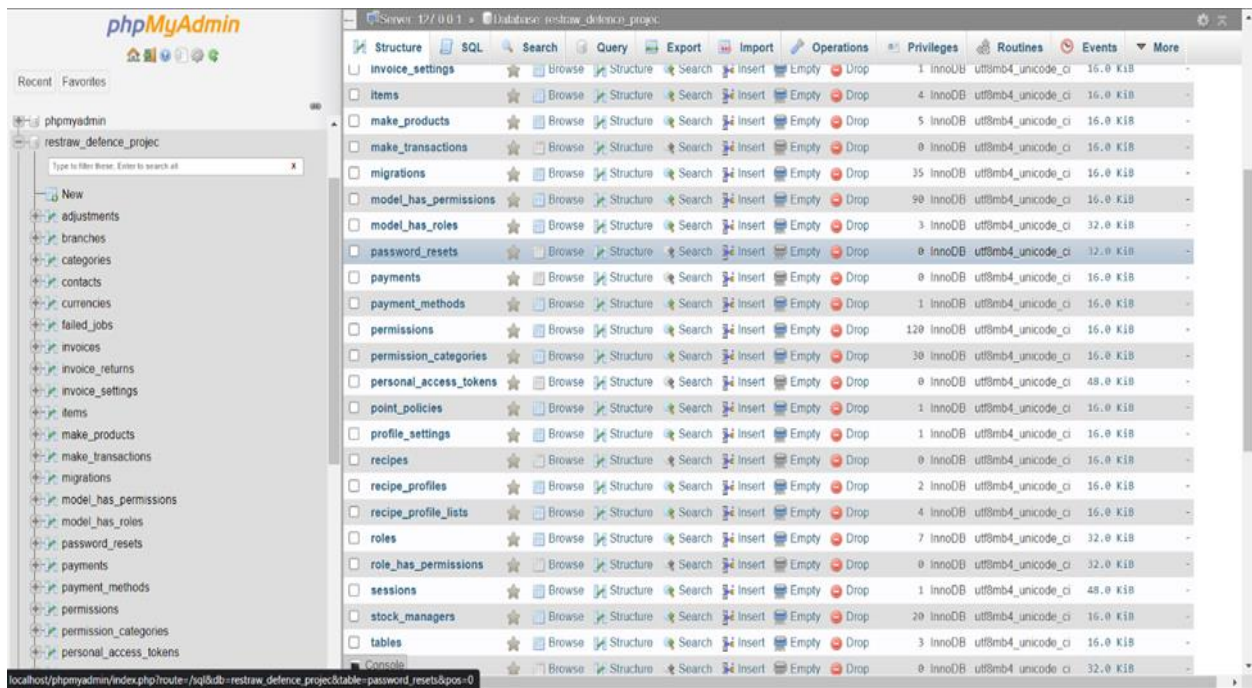


Figure 4.2.1: Database of Restraw.

Figure 4.2.1 depicts a portion of the MySQL database that we utilized. The database table list may be seen in that image. The database has 40 tables for various activities and information. These tables contain and retain information for many features and tasks.

We utilized a web-based admin panel that required the development of many computer languages. HTML5, CSS3, JavaScript, and PHP [4] were utilized. We also used the Laravel framework. An admin may manage all of the restaurant's tasks from the admin interface. Adding a menu, pricing, and food availability. Keep track of each item's sales on each day. An administrator may track all of his spending and income using the admin panel.

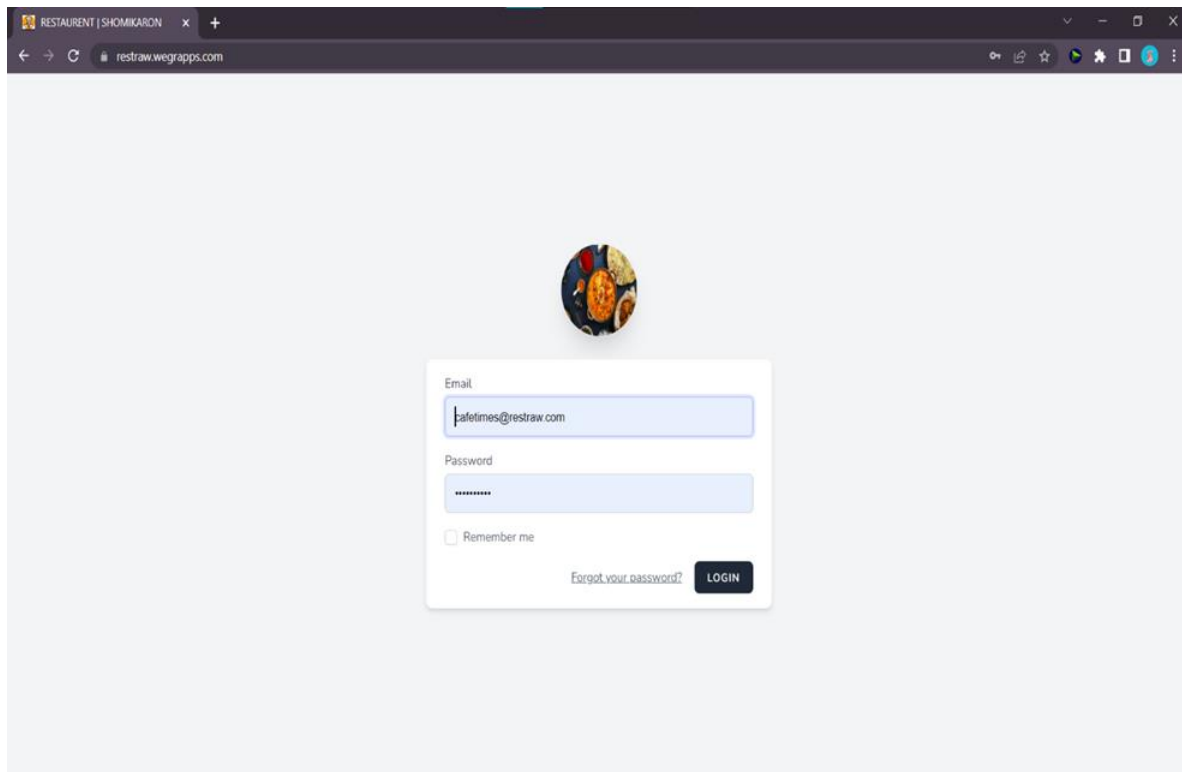


Figure 4.2.2: Login page of admin panel.

Figure 4.2.2 depicts the admin panel's login page. The restaurant owner can access the admin panel via this page by entering appropriate credentials. Nobody can access the admin panel unless they have a legitimate login and password.

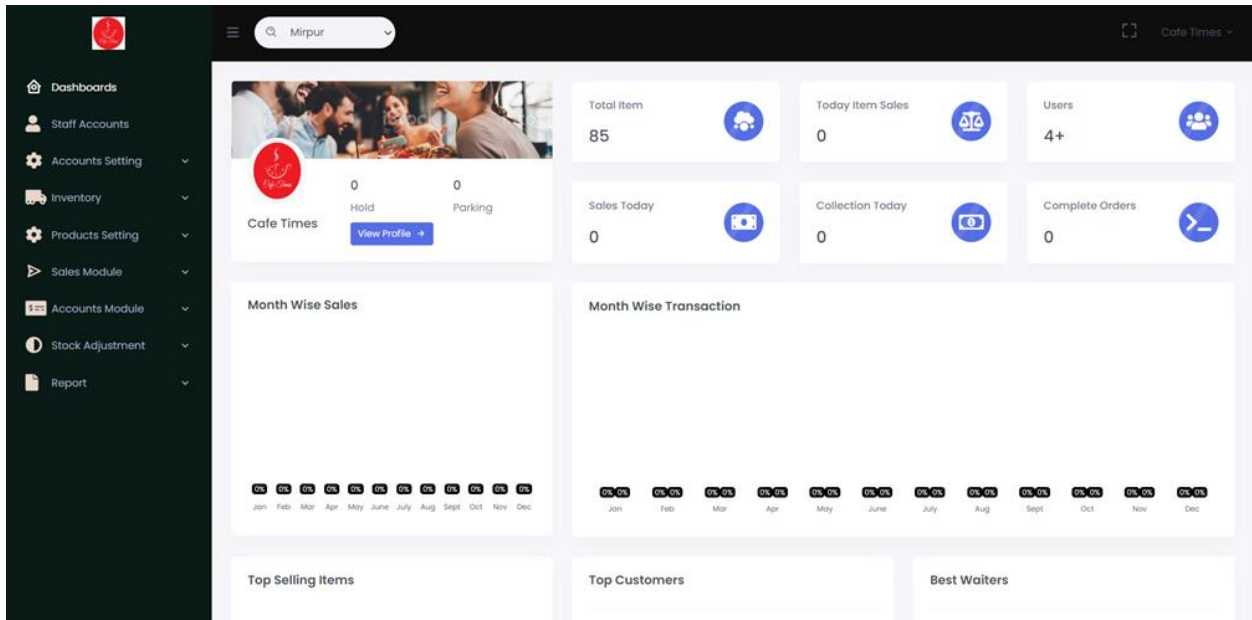


Figure 4.2.3: Dashboard page of admin panel.

Figure 4.2.3 depicts the admin panel's dashboard. The administrator may keep track of total products, daily sales items, users, sales, completed orders, and so on.

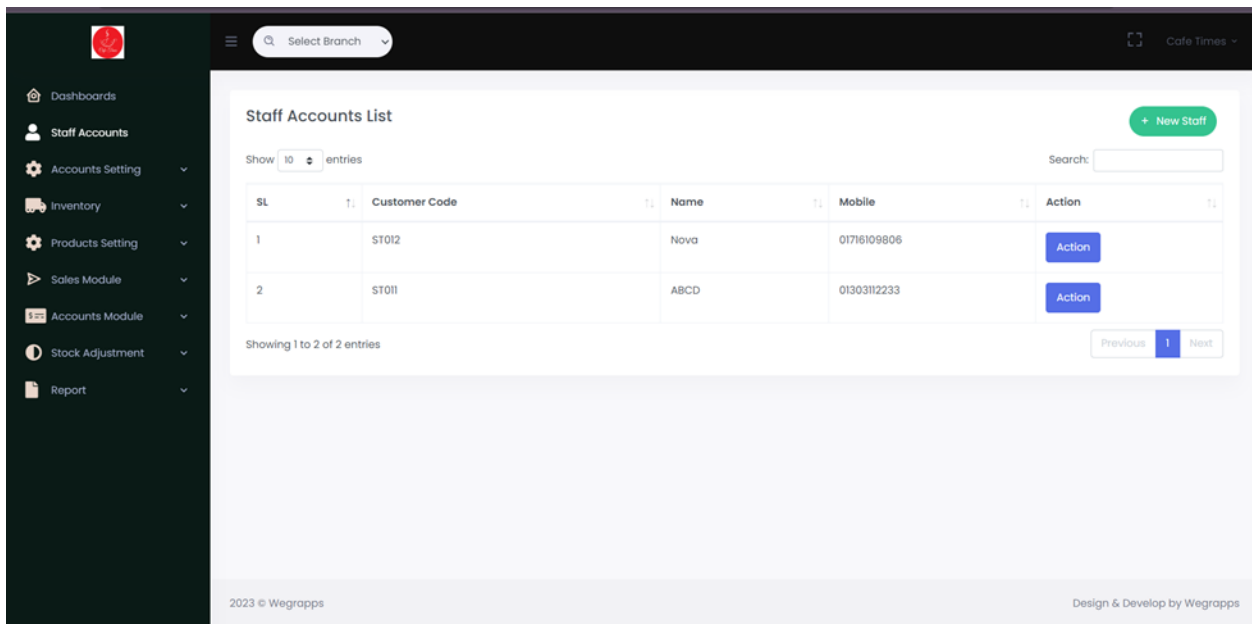


Figure 4.2.4: Staff account list page of admin panel.

Figure 4.2.4 depicts the restaurant's employee list.

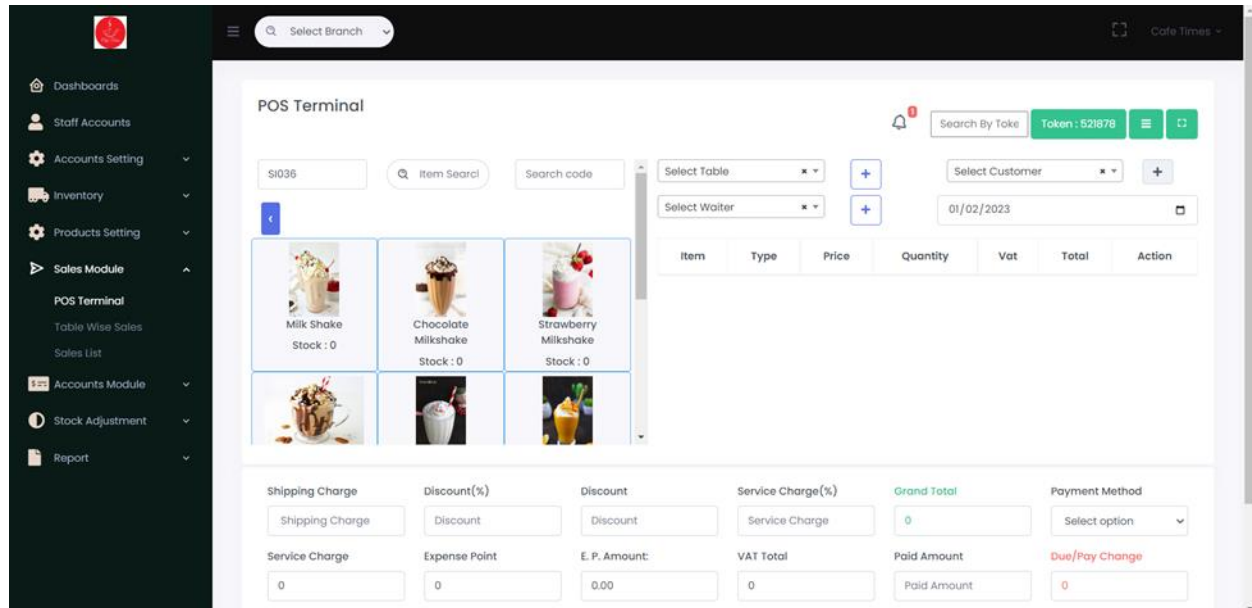


Figure 4.2.5: POS terminal page of admin panel.

Figure 4.2.5 depicts the admin panel's POS terminal page. The administrator will use this POS terminal to sell meals, accept payments, and print invoices. The sales module includes a POS terminal. Table-wise sales and a sales list are also included in the sales Module.

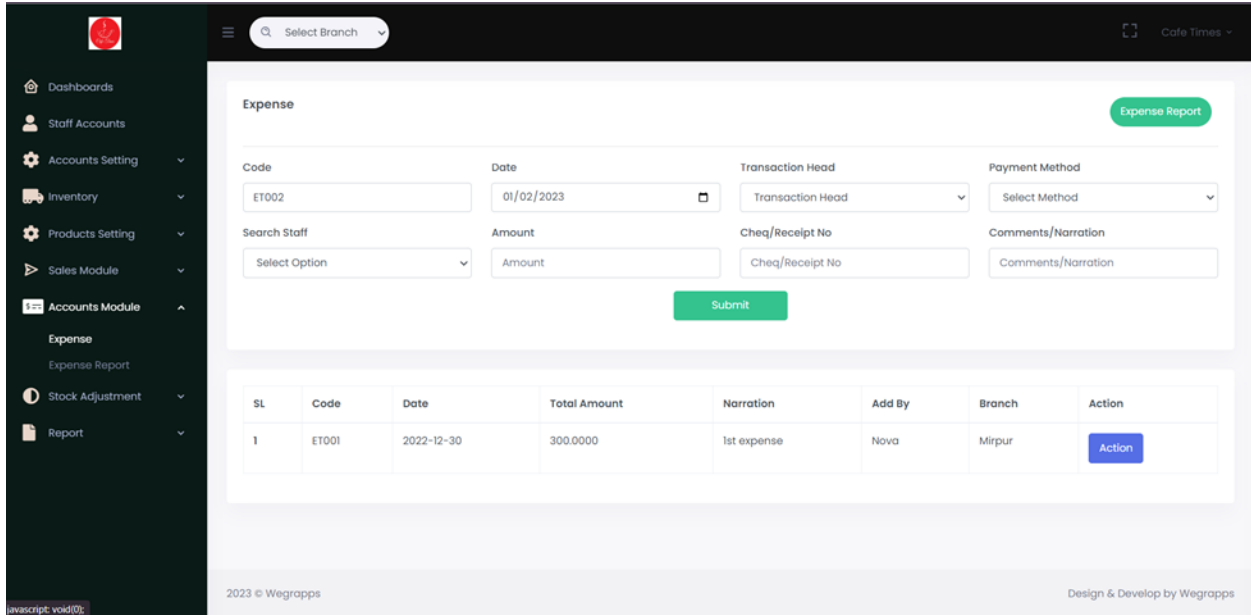


Figure 4.2.6: Expense page of admin panel.

Figure 4.2.6 depicts the date-wise expenses of the restaurant.

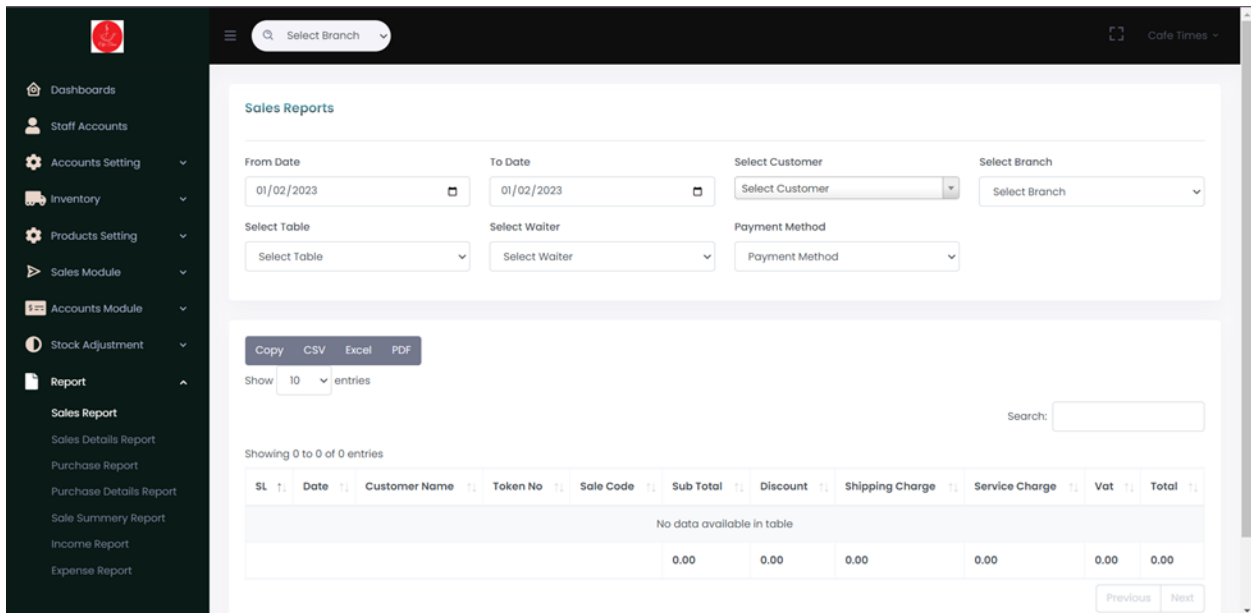


Figure 4.2.7: Sales report page of admin panel



Figure 4.2.7 depicts the admin panel's sales report. An administrator may quickly search any report by date. An administrator can access sales reports, spending reports, revenue reports, purchase reports, and so on under the report area. He may look for and download any report he wants.

### **4.3 Interaction Design**

Nowadays, interaction design is a must-have for any Android phone or tablet. The interface design improves the app's usability and appeals to users. The purpose of interaction design is to build products that help users achieve their goals in the most effective way possible. The right technique to monitor the interaction between a system and its user is through interaction design. User experience, on the other hand, focuses on the total interaction design between a user and an application.

### **4.4 Implementation Requirements**

We required Android Studio to design and build the entire project in order to implement it. On the other hand, we required a browser and a Google account for firebase. For the backend and admin panel, we chose the Laravel framework, which requires the PhpStorm IDE. We also established APIs to access the database and retrieve data for the android application. For the Android applications, we utilized Java, and for the backend and admin panel, we used HTML, CSS, JavaScript, and PHP. For database management, we used MySQL. In the code, we use the MVC pattern. JDK, SDK, and NDK are required for testing. In order to launch and test the software on the PC, we also need a virtual device. Adobe Photoshop and Illustrator are also used to create various kinds of icons and backdrops. We also required a robust internet connection to update the Android studio's Gradle file.

## **CHAPTER 5**

### **IMPLEMENTATION AND TESTING**

#### **5.1 Implementation of Database**

A dependable and secure database is required for each project. In this project, we employed both SQL and NoSQL databases, which allowed us to efficiently execute numerous complicated tasks while also allowing us to boost the security and scalability of the program. Cloud FireStore was a firebase database that we discovered. This is both safe and functional. We set up a Firebase account and a project for this app. Then we link our project to the database. Following that, we created code to generate a table and documents. The firebase database handled authentication. On the other hand, we employ a SQL database to store all of the restaurant's information. Our primary database is MySQL. We retrieve all of the data from the MySQL database to the Android application using the API and code. Using Firebase to generate a unique user id. The firebase is used for authentication and verification. On the other side, we utilized MySQL authentication to provide a unique id for the role-based separation of all users. We used java classes, numerous libraries, and dependencies to connect all of the database's features to our application. To run our program with a database connection, we utilized a variety of techniques to upload data, retrieve data, and register users using MySQL Authentication.

#### **5.2 Implementation of Front-End Design**

Front-End design is very important for any application. There can be multiple techniques to do it. Here in our project, we have used HTML, CSS, JavaScript, PHP and Laravel framework for design. Through the XML file with required icons, buttons, text edit section, navigation bar, different shape of image view like in circular, square, different background, layout, we have designed our user interface. As per user's request data comes from back-end to front-end through API. Besides, our front-end is connected with two type of databases, one is MySQL and another is NoSQL firebase with which the OTP authentication is done.

### **5.3 Implementation of Back-End Design**

We utilized the PHP language's Laravel framework for the backed design. We developed our backend architecture utilizing the framework. We employ HTML, CSS, JavaScript, and PHP to build the backend admin panel. These languages do an excellent job of making the UI more user-friendly and helpful. We design the entire backend system by developing and implementing API and using the MySQL database.

### **5.4 Testing Implementation**

There are several forms of testing in Android. Local unit tests, instrumented tests, unit tests, integration tests, and system tests are all available. The local units are linked to a local computer running Android Studio and linked to a database. They utilize Java on PCs, thus it works incredibly quickly because no emulator is required. We all know that system tests examine the entire system. These are the tests used to validate the code. Unit testing is critical because software developers frequently try to save time by performing a minimum number of unit tests, and the cost is high as a result of this fabled reason inappropriate unit test error fixing during system testing, integration testing, and even beta testing the application is built. Initial unit testing at an early stage saves time and money in the long run. Android unit testing is used for further testing by integrating test frameworks. We evaluated each feature in the Android applications and admin panel individually using various criteria, techniques, and parameters. We also successfully ran the entire app. We also put the QR code to the test, which works well. We also computed all of the return values. We go over each report one by one, as well as double-check and reaffirm the invoice printing success. Except for us, the entire system was tested by a variety of people. We have already released the application and software to the market. Our clients are also utilizing this product in the pitch. This project's testing goes through several phases and yields a successful outcome.

## 5.5 Test Results and Reports

The system is primarily checked for application accuracy. A test is an essential component of every program. Finally, the program communicates with other software and hardware. The system is a collection of system testing results.

- Registration System
- Login System
- Searching System
- QR code scanning
- Reviewing the food
- Profile section
- Try List system
- Utilizing the MAP
- Profit Loss tracking
- Sales System
- Account Settings
- Inventory
- Product Setting
- Sales Modules
- Accounts Module
- Reports
- Explore Restaurants

## **CHAPTER 6**

### **IMPACT ON SOCIETY, ENVIRONMENT AND SUSTAINABILITY**

#### **6.1 Impact on Society**

“Restraw” is a mobile application that can be an essential system for the customer and restaurant owner in Bangladesh. We are doing this project to save time and minimize the customer’s pressure. Through this project a person can easily find any restaurant then after going there his time will not be wasted while waiting for the waiter to come and give the updated menu and take orders. He can just go and scan the QR and can easily see the menu and do on-site order. In this way, customers can save their time and mental pressure and restaurants also can reduce their work pressure. Besides through the owner’s site, the restaurant owner can manage many important things easily, so it also reduces the mental pressure and makes the work easy. So, by saving time and reducing mental and work pressure, this project will have a good positive impact on society.

#### **6.2 Impact on Environment**

The ‘Restraw’ project is for saving time and making the restaurant experience modern and less pressurizing and easy, so there is no bad impact on life and the environment. Even though this project is good for the environment as its a software and its don’t have any toxic or bad outputs. Besides, because of this project, the restaurant doesn’t need to print or write the traditional physical menu every time it's updated or changed even a little bit, the menu could be updated digitally anytime and the customer can see the menu digitally and easily too, so for that, the paper using will be reduced a slightly and this is also good for the environment.

### **6.3 Ethical Aspects**

There are some moral qualities of our 'Restraw' application which is the ethical aspects like example, the application is not made to harm anyone or will not compromise any user's security, it's a risk-free system that will just help to make a better experience in the restaurant. If our 'Restraw' application works properly and maintains the security of the system and its users and, fulfills user requirements then we can call the 'Restraw' trustworthiness and responsible which are some ethical aspects.

### **6.4 Sustainability Plan**

Normally in the project, sustainability means the efficiency and quality of the system which will continue to be good into the future. Our project will make a good change in restaurants and will be helpful for both restaurants and customers. In our digital world, peoples are now dependent on smartphones and they like to do everything with their smartphones and want everything in it. So, gradually, in future people will be more modern and restaurants will move from the traditional way and all restaurants will step into a modern system and will be wished for everything digitally with modern technology. So, there is a high chance of Restraw to be used in the future efficiently. In the future, we will implement more useful functions and services and according to user needs, changing perspective we will update the system from time to time and these will have a good positive impact on ensuring the quality of service.

## **CHAPTER 7**

### **CONCLUSION AND FUTURE SCOPE**

#### **7.1 Discussion and Conclusion**

As nowadays most of the people are taking their smartphones everywhere with them, absorbing the advantages and can use it very easily anywhere, so we think why not we make people's restaurant experience better than before through a mobile application! Normally, our traditional restaurant system is time-consuming and not technologically much advanced. After going to a restaurant, we have to wait for the waiter to know the updated menu for today, the best-rated food of the restaurant than to take order and all. So, we thought of a modern solution for restaurants and made a system. The customer will have a mobile application on his phone through which he can easily access the updated menu of the restaurant, get to see the best-rated food, and ratings by scanning the QR on the table and also can make on-site orders. Besides, before going to a restaurant a person can make a plan and decide to which restaurant, he should go by seeing the menu and ratings of the restaurant in the application. So, through this application customers can save time and make their restaurant experience better.

#### **7.2 Scopes for Further Development**

As we want to make the best solution for restaurants and make customer and owner's experience much better ever than before, so there are many scopes for further development. Our application is not capable to transfer the customer's order direct to the restaurant's chef but soon we will make this happen. We can integrate different online payment methods like paying with Nagad QR, Credit/Debit cards. We can also make this application a bridge between customer and restaurant. The restaurant can do advertisements and send offers to its regular customer through the application. We can also improve our implemented functionality and make it more user-friendly. We can ensure that our system will be capable to adapt the environment and could be upgraded. Last but not the least we can make the security system and database backup system much stronger for our owner site and customer application.

## References

- [1] Hossain, Md. Kaviul. (2019). *Smart & Digital Ordering System for Restaurants*.  
10.13140/RG.2.2.28668.67201.
- [2] Ahmed, Jashim & Ahmed, Asma. (2018). *Foodpanda: Changing the Way Bangladeshi Eat Meals*.  
10.4135/9781526444561.
- [3] V. Liyanage, A. Ekanayake, H. Premasiri, P. Munasinghe and S. Theiljagoda, "Foody - Smart Restaurant Management and Ordering System," 2018 IEEE Region 10 Humanitarian Technology Conference (R10-HTC), 2018, pp. 1-6, doi: 10.1109/R10-HTC.2018.8629835
- [4] "HTML5" Available-at: <https://en.wikipedia.org/wiki/HTML5> [Last accessed: 2022-11-15 5:00 AM]
- [5] "CSS3" Available-at: <https://en.wikipedia.org/wiki/CSS> [Last accessed: 2022-11-15 5:00 AM]
- [6] "JavaScript" Available-at: <https://en.wikipedia.org/wiki/JavaScript> [Last accessed: 2022-11-15 5:00 AM]
- [7] "PHP" Available-at: <https://www.php.net/> [Last accessed: 2022-11-15 5:00 AM]
- [8] "Java" Available-at: <https://www.java.com/en/> [Last accessed: 2022-11-15 5:00 AM]



OBJECT\_REPORT\_ON\_RESTRAW\_-  
\_A\_MODERN\_SOLUTION\_FOR\_RESTAURANTS.docx

ORIGINALITY REPORT

<b>21</b> %	<b>19</b> %	<b>0</b> %	<b>16</b> %
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

<b>1</b>	<a href="https://dspace.daffodilvarsity.edu.bd:8080">dspace.daffodilvarsity.edu.bd:8080</a> Internet Source	<b>13</b> %
<b>2</b>	Submitted to Daffodil International University Student Paper	<b>5</b> %
<b>3</b>	Submitted to Sri Lanka Institute of Information Technology Student Paper	<b>1</b> %
<b>4</b>	Submitted to Cambridge International University Student Paper	<b>&lt;1</b> %
<b>5</b>	Submitted to University of Greenwich Student Paper	<b>&lt;1</b> %
<b>6</b>	Vindya Liyanage, Achini Ekanayake, Hiranthi Premasiri, Prabhashi Munasinghe, Samantha Thelijjagoda. "Foody - Smart Restaurant Management and Ordering System", 2018 IEEE Region 10 Humanitarian Technology Conference (R10-HTC), 2018 Publication	<b>&lt;1</b> %

7

eprints.utar.edu.my

Internet Source

<1%

---

8

Submitted to Higher Education Commission  
Pakistan

Student Paper

<1%

---

Exclude quotes  On

Exclude matches  < 10 words

Exclude bibliography  On

