Food Swipe Canteen Food Ordering System

H. F. Mohamed Insaaf

2021

Food Swipe Canteen Food Ordering System

A dissertation submitted for the Degree of Master of Information Technology

H. F. Mohamed Insaaf
University of Colombo School of Computing
2021

Declaration

The thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.

To the best of my knowledge it does not contain any material published or written by another person, except as acknowledged in the text.

Student Name: H. F. M. Insaaf
Registration Number: 2017/MIT/024
Index Number: 17550242

Signature: Date: 27-11-2021

This is to certify that this thesis is based on the work of

Mr./Ms.

under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standards.

Certified by:

Supervisor Name: Dr. M. G. N. A. S. Fernando

Signature: Date: 29-11-2021

Abstract

The ultimate aim of the Food Swipe Canteen Food Ordering System is to automate existing traditional manual systems with the support of current technology. In venues with enormous crowds, consumers have to sit in long waits, a substantial amount of time is spent waiting for food to be brought, paying the bill, and getting orders processed. Food Swipe is a solution to rectify this issue. Customers can order their food and make payments online. The time wasted on queues and waiting for cash change will be diminished by the enablement of Food Swipe for the canteen business. Also, the application can manage canteen products, orders, customer details, and sales information which the canteen administration can rely upon. Customers can make cashless payments using wallet points, which can be recharged from the canteen caretakers. The application is built using web technologies such as JS, PHP and HTML.

Acknowledgment

First and foremost, I would like to thank the academic/non-academic personnel of the University of Colombo School of Computing for the support and guidance provided throughout the Master's Program. And also I would like to thank the Lecture panel of the Master's program at UCSC for assistance and support.

Special thanks go to the supervisor Dr. M. G. N. A. S. Fernando, for the advice and guidance given throughout the project. I am also really thankful for the help and guidance provided by project coordinator Miss. H. N. D. Thilini.

Last but not least, I would like to thank all my family members and friends who were with me throughout this project, giving me strength and never letting me go without advice and support whenever required.

Table of Contents

List of Figures

List of Tables

List of Acronym

UI	-	User Interface
FE	-	Front End
JS	-	JavaScript
CSS	-	Cascading Style Sheet
HTML	-	Hypertext Markup Language
iOS	-	iPhone Operating System (Apple)
PC	-	Personal Computer
SDLC	-	Software Development Life Cycle
HTTP	-	Hypertext Transfer Protocol
PHP	-	Hypertext Preprocessor
SQL	-	Structured Query Language
MIS	-	Management Information System
IDE	-	Integrated development environment
POP3	-	Post Office Protocol
IMAP	-	Internet Message Access Protocol
LDAP	-	Lightweight Directory Access Protocol
OTP	-	One Time Pairing
SMS	-	Short Message Service

1. Introduction

1.1 Chapter Overview

The introduction chapter is based on the overview of how to overcome the issues in canteen services in most places like hospitals, corporate offices, factories, colleges, and hostels with immense crowding. When considering these places, it takes more time to spend in queues, placing an order, order processing, payments. Furthermore, it is not advisable and safe to spend a long time in queues and gatherings with the prevailing country situation related to the covid-19 pandemic. An Online canteen ordering system was proposed to prevent relevant situations and ease the process for both customers and canteen caretakers.

1.2 Motivation and Project Background

Running a canteen service in places with huge crowds like hospitals, corporate offices, factories, colleges, and hostels are time-consuming. For most canteens, customers need to visit the canteen and wait in the queue for a long time to order the food, which is not safe for the prevailing country situation related to the covid-19 pandemic. And in some canteens, it is possible to place an order via a phone call, but with many disadvantages,

- Customers will not have a physical copy of the menu while placing an order.
- Lack of visual confirmation that the order was placed correctly.
- The customer has no visual view of the food that has been placed.
- The necessity for the canteen to have an employee answering the phone and taking orders.
- The person who takes the order needs to verify the availability during the call, which is a time-consuming task. If an order is taken for an unavailable item, the caretaker might need to call back customers.

The other primary concern is the payments. The customer has to carry the cash all the time to make the payments. Moreover, the customer has to pay the correct amount; else, there might be issues

in providing the balance money. According to the latest finding, it is unsafe to handle cash because the high transmissible rate of covid-19 is identified through cash.

When it comes to making business decisions, it is very hard due to a lack of proper report maintenance. Current reports are paper-based and could not provide analytical details on the number of items sold each day, which items have more demand, the items having the least demand? Furthermore, customers have no way to review or rate items or make any complaint with the current canteen operation rather than the canteen caretaker, which may not be considered most of the time.

1.3 Problem Justification

Customers spend a considerable amount of time in queues to buy food from a canteen. Considering a canteen premises in a university as an example with a vast customer base, students have to spend more time in the queues rather than the time spent on consuming food, which may lead them to miss classroom sessions. Also, the canteen caretakers have to be quick on getting orders placed from the customers and processing them. One of the other significant issues in the process is dealing with cash where it is not always possible for the canteen caretakers to give the exact balance amount to the customers, increasing the waiting time by a considerable amount. Running the entire business process manually brings huge disadvantages when retrieving business reports (weekly, monthly, yearly). The owner has to go through a vast set of files to retrieve specific data.

1.4 Aim and Objectives

1.4.1 Aim

This project aims to analyze, design, develop and evaluate a web-based Online Canteen Food Ordering System to provide expected outcomes with the system by improving the overall canteen process.

1.4.2 Objectives

- Identify the drawback of the current canteen service.
- Identify the nature of the Online Canteen Food Ordering System.
- Maintain simple, user-friendly user interface design for the Canteen Food Ordering System.
- Implement the Stock management component.
- Implement the system to be flexible and easy to install in the canteen in hospitals, corporate offices, factories, colleges, and hostels.
- Maintenance of the system should be made accessible.
- The system to be flexible to expand and enhance when required.
- The system to be supported on all leading web browsers.
- cashless transactions to be incorporated to the application.
- Customer wallet recharge feature needs to be incorporated into the application.

1.4.3 Process

The number of canteens is selected from different places like universities, hospitals and corporate offices and from these canteens customers, canteen caretakers and canteen administrators taken into consideration to collect data through questionnaires and interviews to study the actual requirements of the canteen system. The whole population is not to be studied. Only a sample representing the population is to be studied. Collected data are analyzed based on the majority of the responses gathered. The data collected from the questionnaires are computed to get the final output of responses.

1.4.4 Proposed Solution

The proposed solution is expected to overcome the drawbacks identified in the current canteen service operations. Online canteen food ordering system dramatically simplifies the ordering process for both the customers and the canteen caretakers.

Customers are presented with an interactive and up-to-date menu, complete with all available options and adjusting prices based on the selected options when the customer visits the ordering web page. Upon customer selection, the items are added to the order. Customers can review the details and get visual confirmation of the order at any time before checking out. This system also dramatically lightens the load on the canteen's end, as the entire process of placing orders are computerized.

Once an order is placed, it is entered into the database and then retrieved, in pretty much real-time, by the application on the canteen's end. Within this application, all items in the order are displayed, along with their related options and details, in a concise and easy to read manner. This allows canteen caretakers and admins to quickly go through the orders as they are placed and produce the necessary items with minimal time and confusion. Once the order is confirmed, the order will be moved to an 'ACCEPTED' state and customers will be notified with an email notification sent to their configured email address. Upon the preparation of the order, it will be moved to a 'READY-TO-PICKUP' state. Customers can then pick the order from the canteen and either dine-in or take away. Once the order is handed over to the customer, the order will be moved to the delivered state. Customers are also allowed to cancel their order before the order is accepted from the canteen.

The system allows both cash and cashless transactions and offers the option of a prepaid accounting system called the wallet. Customers can hand over cash to the administration and recharge the wallet on a weekly or monthly basis. Customers can make the payments via EZ-cash or m-cash as well. When an order is placed, the order amount will be deducted from the wallet, so the customer has no need of carrying cash always. In the case of order cancellation, the amount will be reverted to the customer's wallet. Since this does not require credit/ debit card details to

pay the bills, there are no security threats and no additional cost in implementing a payment gateway in the proposed system.

Customers can rate and review the items purchased and are also capable of logging complaints which the admin can view in the dashboard. This helps canteen management to increase sales, understand the customer mindset, and increase customer loyalty.

Food Swipe web application saves the canteen's time by avoiding the food orders taken over the phone that have to be done manually. By making the ordering process fully computerized, it increases the cost-effectiveness and productivity of the canteen with less workforce. It also allows canteen owners to update the online menu, food items, wallet recharges and customer details quickly and helps to stay in touch with the customers.

1.5 Content of the Report

Chapter 2 explains the analysis and the background study of the application including similar systems that are available in the market. The 3rd chapter explains the Design architecture of the application. The implementation details will be discussed in chapter 4. Chapter 5 explains how testing and the evaluation for the application will be carried out and the results of the tests and evaluations. In conclusion, chapter 6 will discuss future work and improvements of the application.

2. Background

2.1 Introduction

The expectation is to evaluate and understand the system requirements for the proposed canteen system, focusing on functional and non-functional requirements. The system's structure and how the functionalities are placed with the system will be explained in this section. Furthermore, the background of the proposed application on how the existing systems perform with relevant features, drawbacks of existing systems, and how the proposed system overcomes these issues. Furthermore, a comparison of features, design strategies, development strategies, technologies, hardware and software of similar systems, and improvements to the proposed application are discussed in this chapter.

2.2 Requirement Analysis

2.2.1 Functional Requirements

When considering the functional requirement of the proposed application, it is required to focus on each user role interacting with the application. Detailed functionalities of each user role are listed out below.

Customers are provided with the following functionalities in Food Swipe:

- Create an account.
- Log in to the system.
- View the canteen’s menu.
- Select items from the menu.
- Place an order.
- View order details.
- Make payments through the wallet.
- Rate and review items.
- Receive confirmation on order updates.

- View purchased order history.
- Cancel order
- Log complaints

Food Swipe should have the following functionalities for the caretaker of the canteen:

- Log into the system.
- Add / update / delete item and pricing details.
- View orders placed by customers.
- Confirm/ Reject orders placed by customers.
- Confirm order processing so the customers can pick up the order from the canteen.
- Recharge customer wallets.

Also, Food Swipe will have an Admin role with a dashboard allowing the below functionalities.

- View complaints logged by customers.
- Generate Daily, weekly, monthly, and annual sales reports.
- Profit generated through each item, apply a filter based on date range, food type, etc.
- View the ratings and reviews added by customers. Based on review and rating, the admin can decide the next steps to improve the quality. This feature would allow the management to stay connected with consumers and give better service to them.

2.2.2 Non-Functional Requirements

Food Swipe should have below mentioned non-functional requirements as well.

- Support cash payments for customers with insufficient wallet points.
- The system needs to be more user friendly. The user interface to be kept simple.
- Flexibility to expand and enhance when required.
- Flexible and easy to install in canteens in hospitals, corporate offices, factories, colleges, and hostels.
- Responsiveness taken into consideration during the implementation of the system, where users are able to access in mobile, tabs and computers.
- Maintain browser compatibility.

2.3 Review of Similar Systems

Company canteen management system [1] was developed to provide fast services to their company employees using their previously stored records. However new records can be added any time whenever an employee visits the canteen. Each employee is provided with a smart card, and each smart card will have a unique number by which the system will quickly identify their employee and prepare their bills for the services they have taken. It will also help the canteen shop admin to identify such employees whose payment has not been cleared to date. Admin will have the facility of the items available at their shop at a particular time and the exact quantity of a specific item. An employee can also check their account status, services which they have taken and bills which they paid, and due amount details if there is any for their account.

Online Canteen system [2] allows users to order their food on the e-menu card, which will be sent to the chef for preparation as soon as the order is placed. The system is also capable of processing online card payments with the integration of a payment gateway.

The canteen Automation System [3] focuses on improving the ordering process for both customers and the canteen. Customers can go through the menu provided by the canteen and place the order. From the canteen end, the order is processed automatically. This also enables delivery options for the customer.

Mobile Application for Canteen Automation System using Android [4] allows users to go through the menu provided by the canteen admin and select the food they need to order using the android device even though this does not have the capability to make online payments and based on cash transactions.

The number of existing systems mentioned above includes only limited features and focuses on ordering food from the canteen, and most of them do not provide payment facilities and reporting facilities. Existing systems do not solve drawbacks in the entire canteen process. When it comes to the proposed solution, it not only supports ordering, it allows to maintain hassle-free payments with minimal security requirements and no payment gateway integrations required. Also capable

of reviewing, rate and make complaints about the canteen services. Reporting module to generate various reports also included in the proposed application.

Table 1 compares the main functionalities in the systems mentioned above and with the proposed Food Swipe canteen system.

Table 2.1 : Feature Comparison

	Food Swipe	Company Canteen Management System [1]	Online Canteen System [2]	Canteen Automation System [3]	Mobile Canteen Automation System [4]
Customer/ Admin Login	√	√	√	√	√
Add/ Update/ Remove menu items	√	√	√	√	√
Customer place order	√	√	√	√	√
Product Ratings and review	√	x	x	x	x
Customer Complaints	√	x	x	x	x
Customer view purchase history	√	√	√	x	x
Cashless payments	√	x	√	x	x
Payment gateway integration	x	x	√	x	x
Generate sale reports	√	√	x	x	x
Mobile and Web accessibility	√	x	x	x	x

2.4 Related Design Strategies

2.4.1 Alternate Design Solutions

Several alternative design solutions were evaluated to find the best solution which fulfils all requirements most effectively.

Web Applications

Web applications utilize web browsers and web technology to perform tasks over the Internet. Millions of businesses around the globe use the Internet as a powerful and cost-effective communications method. It allows them to exchange information and data with their target market and make fast, secure transactions. However, effective engagement is possible when the business can capture and store all the necessary data and process this information and present the results to the user. Web applications use server-side scripts to handle the storage and retrieval of the information and client-side scripts to present information to users. This allows users to interact with the business using online forms, content management systems, shopping carts, and more. Also, the applications enable customers to access the application regardless of location or device.

Mobile Applications

Mobile applications are away from the integrated software systems generally found on personal computers. Instead, each application provides limited and isolated functionality. Mobile applications are designed to run mobile devices like smartphones and tablets. Mobile apps are divided into two broad categories: native and hybrid applications. Native mobile apps are built targeting specific mobile operating systems such as android, iOS, etc. Native mobile apps enjoy better performance and a more finely-tuned user interface (UI) and usually need to pass a much stricter development and quality assurance process before they are released. Hybrid mobile apps are built using web technologies such as JavaScript, CSS, and HTML 5, focusing on several mobile platforms. Hybrid apps are easy and fast to develop, which is a clear benefit. On the other hand, these apps may lack speed and performance.

2.4.2 Proposed Design Solutions

When considering the above alternative designs, it was decided to develop a web application that perfectly matches and satisfies the client's requirements. The web application can be made responsive that can be accessed on different devices such as PCs, laptops, smartphones, and tablets. Some reasons for proposed solutions over other alternate solutions are listed below:

- The system can be accessed from anywhere at any time.
- Not depend on the client's operating system, which is great to reach all users with different operating systems and devices.
- Development time is less compared to other solutions.
- Client's no need to download or install any additional applications apart from the web browser, which will be there in their devices by default.

2.5 Process Model

The prototyping model will be carried out as the process model for the proposed application. The Prototyping Model is one of the most popularly used Software Development Life Cycle Models (SDLC models). This model is used when the customers do not know the exact requirements during the start of the project. In this model, a prototype of the application is first implemented, tested, and refined as per customer feedback repeatedly until a final satisfactory prototype is achieved, forming the basis for developing the final product. Some of the significant advantages of using this model are as follows:

- The customers get to see the partial application early, which ensures a greater level of customer satisfaction and comfort.
- New requirements can be easily adapted as there is scope for refinement.
- Can figure out missing functionalities quickly.
- Errors can be detected earlier, thereby saving a lot of effort and cost, besides enhancing the quality of the software.

3. Design Architecture

3.1 Introduction

In this chapter, based on the gathered requirements and resource availability of the system, the design architecture of the project and the most suitable development solutions will be explained. It is expected to exhibit high-level diagrams and design structures to implement the best design for the system. Furthermore, UML diagrams for the Food Swipe will be discussed here.

3.2 System Architecture

For the development of the system, two-tier architecture will be used, which is also known as a client-server architecture. The client is on the first tier, whereas the database server and web application server reside on the same server machine in the second tier. This second tier serves the data and executes the business logic for the web application. This allows consolidating the application capabilities and database server capabilities on a single tier, which is responsible for providing the proposed system's availability, scalability, and performance characteristics. The system architecture is shown below in Figure 3.1.

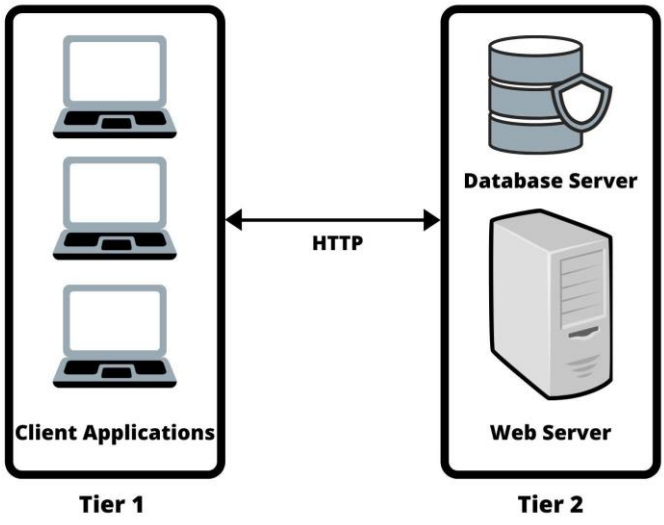


Figure 3.1 : Two-tier connectivity between client and server

The browser uses Hypertext Transfer Protocol (HTTP) to forward the requests from users to the server machine in the second tier. (HTTP is a web protocol that is used for communication) The web server on the second tier invokes the database server to satisfy the application's data requirements. Client applications will be logged into the system via web browsers through desktops, mobiles, or tablet pc.

As discussed in Chapter 2, the server-side implementation will be carried out using the PHP Hypertext Preprocessor (PHP) and HyperText Markup Language (HTML), Cascading Style Sheet (CSS), and JavaScript (JS) will be used as the client-side scripting language. Structured Query Language (SQL) will be used as the Database Management System, the standard programming language for relational databases. Further details will be discussed in chapter 4.

The detailed description of the use case diagram, database design, class design, and User Interface designs is explained below.

3.2.1 Use Case Diagram

Customers are allowed to place orders by selecting preferred items, log complaints, rate and review items. Also, they are capable of order cancellation if required. Admin can generate reports and view complaints logged by the customer, which helps improve customer satisfaction. Furthermore, the canteen caretaker is allowed to add/update and delete items to the canteen application, accept/ reject an order placed by customers and recharge customer wallets. The use case diagram in Figure 3.2 explains the features of the Food Swipe application and the user roles of the system in more detail.

Figure 3.2 : Use Case Diagram of Food Swipe

Table 3.1: Login use case description

Use Case	Login	
Actors	Admin, Caretaker, Customer	
Goal	To login to the Food Swipe application	
Overview	When the system users needs to interact with the system, they will need to login to the application	
Typical course of events		
Actor Actions		**System Response**
1. Enter email and password.		
2. Click on login.		3. System will validate the credentials and directs user to Home page.
Alternative Course		
Step 3	If entered credentials are not valid user will be shown an incorrect username password alert.	

Table 3.2: Customer Registration use case description

Use Case	Registration	
Actors	Customer	
Goal	To create new account in the Food Swipe application	
Overview	When a customer needs to create a user account to access the application.	
Typical course of events		
Actor Actions		**System Response**
1. Enter requested details in the form		
2. Click on Register me button		3. The system will validate the field values.
		4. Create an account in the database and display success message.
		5. Redirect user to Login page.
Alternative Course		

Step 3	If entered incorrect data into fields, display error alert indication the issue.

Table 3.3: Place Order use case description

Use Case	Place order	
Actors	Customer	
Goal	Place an order.	
Overview	When the customer need to place an order through food swipe	
Typical course of events		
Actor Actions		**System Response**
1. Select the item need to be ordered		
2. Set quantity for the order.		
3. Select payment option from the cart and proceed with order		4. Create the order record in the DB.
		5. Reduce quantity of the line items from the product DB.
Alternative Course		
Step 4	If the payment is a wallet payment, reduce the wallet balance.	

Table 3.4: Rate & Review use case description

Use Case	Rate and Review	
Actors	Customer	
Goal	To rate and review an item	
Overview	Customer need to provide a review for the items purchased.	
Typical course of events		
Actor Actions		**System Response**
1. Select the item need to be added rating and review.		
2. Enter review comment, select rating and submit.		3. Review should be saved in the DB

	4. Added review and rating should be displayed in product details page.

Table 3.5: Log Complaint use case description

Use Case	Log Complaints
Actors	Customer
Goal	To log complaints on Food Swipe
Overview	When a customer needs to make a complaint on the canteen operations or issues in the food swipe application.
Typical course of events	
Actor Actions	**System Response**
1. Enter complaint title	
2. Enter complaint in details and click send.	3. The complaint record will be stored in DB.
	4. List out the complaint in customer logged complaint screen.

Table 3.6: View Complaints use case description

Use Case	View Complaints
Actors	Admin
Goal	To view complaints
Overview	When a admin needs to view complaints raised by customers related to the canteen operations or issues in the food swipe application.
Typical course of events	
Actor Actions	**System Response**
1. Navigate to view complaint screen	2. All complaints raised should be displayed.
3. Acknowledge the complaints by clicking READ button.	4. The complaint record will be updated in DB.
Alternative Course	

Step 3	Admin can leave the complaint unread if not viewed or taken any actions yet.

Table 3.7: Generate Report use case description

Use Case	Generate Reports	
Actors	Admin	
Goal	To generate reports from the food swipe.	
Overview	When a admin needs to generate various reports related to sales, stocks and customer details.	
Typical course of events		
Actor Actions		**System Response**
1. Select the type of the report to be generated.		
2. Add any filters, such as date if required and click display.		3. The records will be displayed in a tabular format.
4. Click print icon if the reports needed to be taken as hard copies.		
Alternative Course		

Table 3.8: Wallet Recharge use case description

Use Case	Recharge Wallet	
Actors	Caretaker	
Goal	To recharge the customer wallet	
Overview	When a customer needs to recharge wallet balance, customer can reach out caretaker to recharge the wallet	
Typical course of events		
Actor Actions		**System Response**
1. Select the customer account that needs to be recharged.		2. List out all the customers in a dropdown.

3. Enter wallet recharge amount.	4. The data needs to be updated in DB in both customer DB and recharge DB.

Table 3.9: Confirm/Reject Order use case diagram

Use Case	Confirm/Reject Orders
Actors	Caretaker
Goal	To confirm or reject orders
Overview	When a customer place an order, caretaker can either confirm or reject the order placed.
Typical course of events	
Actor Actions	**System Response**
1. Navigate to the order view page.	2. List out all the orders placed by the customers.
3. Confirm the order by clicking the confirm button	4. Update the order status to confirmed in the order DB.
Alternative Course	
Steps 3-4	Can reject order by clicking the reject button and the order status will be updated to rejected in the DB.

Table 3.10: Add/Update Items use case description

Use Case	Add/Update Items
Actors	Caretaker
Goal	To add new item or update existing item
Overview	When a caretaker need to add a new item to the canteen menu or to update a existing menu
Typical course of events	
Actor Actions	**System Response**
1. Enter values to the add item form	

2. Click on save.	3. Item record should be added to the product DB. And display success message.
	4. Display the item in the menu
Alternative Course	
Steps 3-4	If the item already exist, update the record in the DB, and reflect the updates in the menu.

3.2.2 Database Design

Structured query language will be used as the database management system, which is the most widely implemented database language. The Food Swipe database will contain tables illustrated in Figure 3.3.

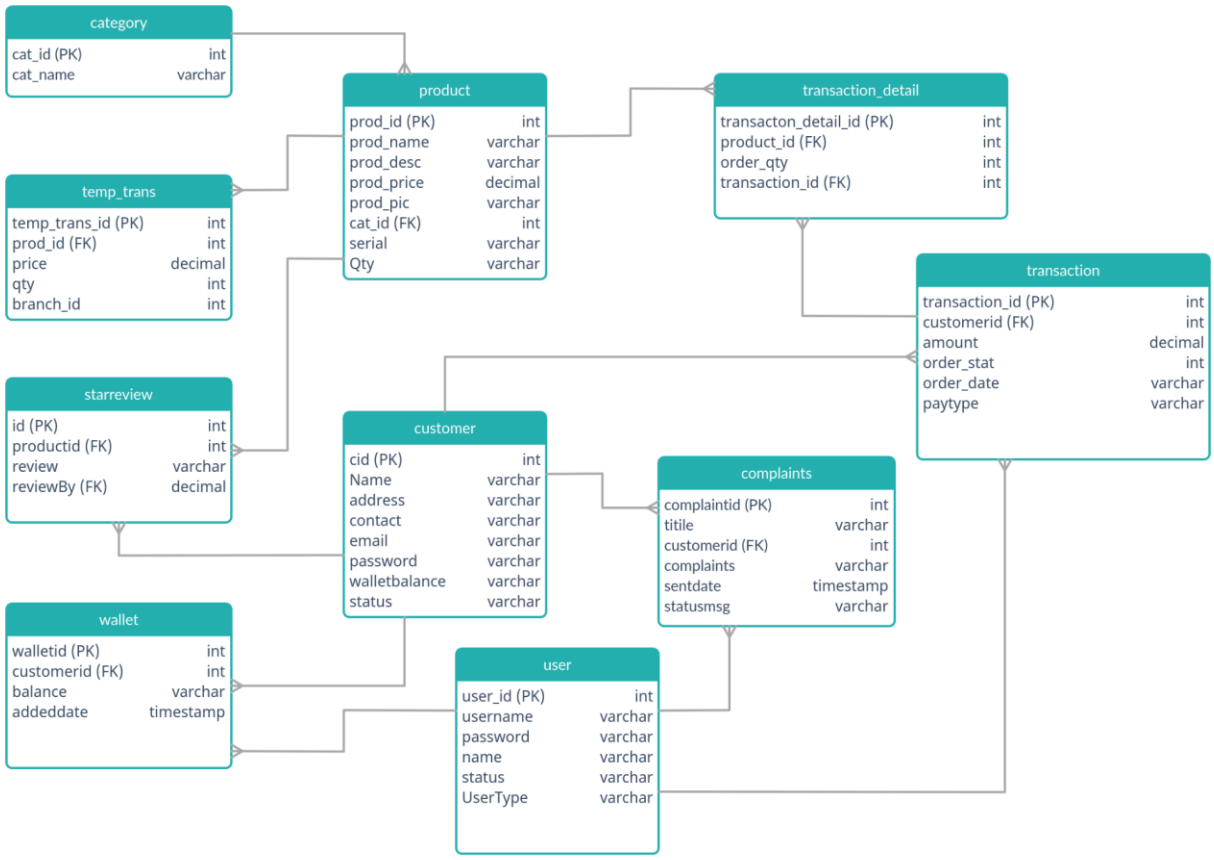


Figure 3.3 : Database Design of Food Swipe

3.2.4 User Interface (UI) Design

As discussed in the above use case diagram, customers, admins, and canteen caretakers can log in to the application to perform their tasks. The login screen will be similar to Figure 3.4 illustrated below.

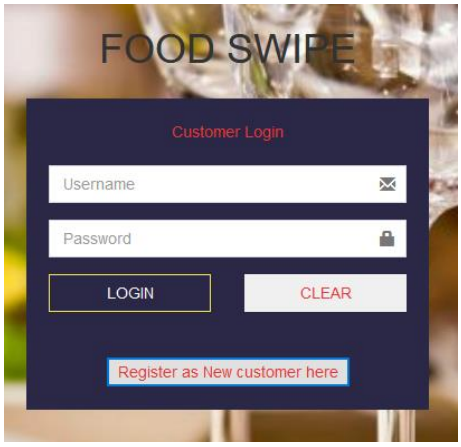


Figure 3.4 : Customer Login Screen

The interface below illustrates the registration screen which customers can use to enroll to the application.

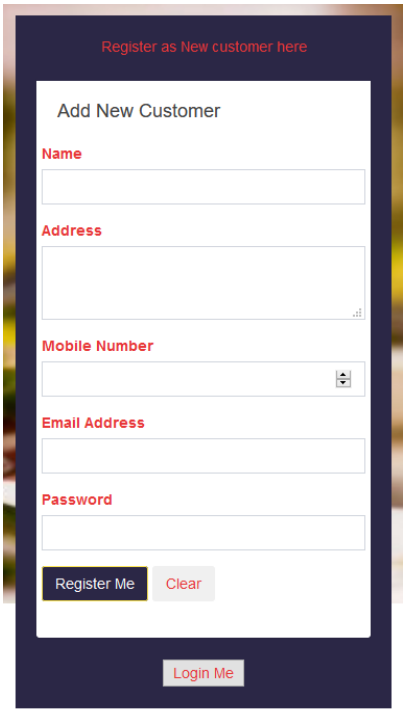


Figure 3.5 : Customer Registration Screen

Interfaces below illustrate the home pages of the Admin and the canteen caretaker.

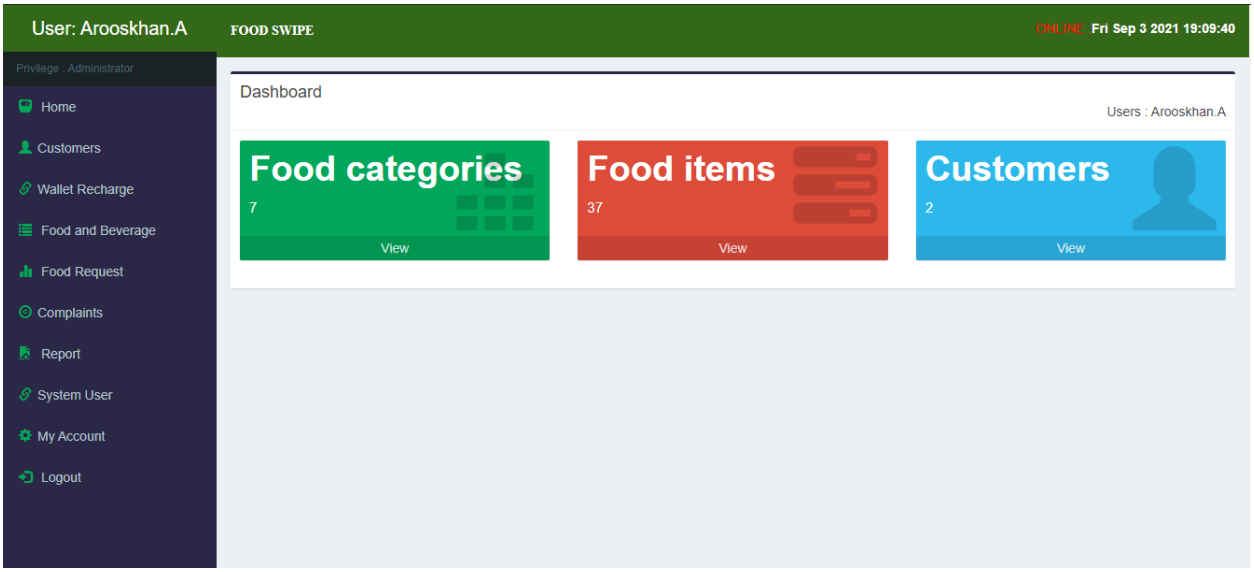


Figure 3.6 : Administrator Home Page

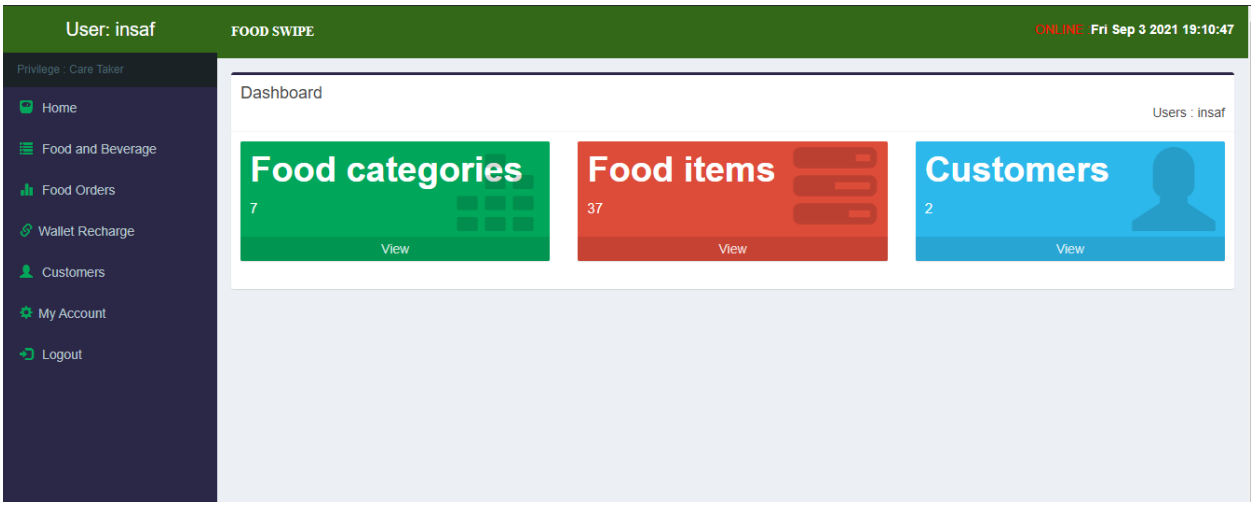


Figure 3.7 : Caretaker Home Page

The below interfaces illustrate the screens of the application used to add/update and delete items, confirm orders placed by customers, and recharge the customer's wallet by the canteen caretaker.

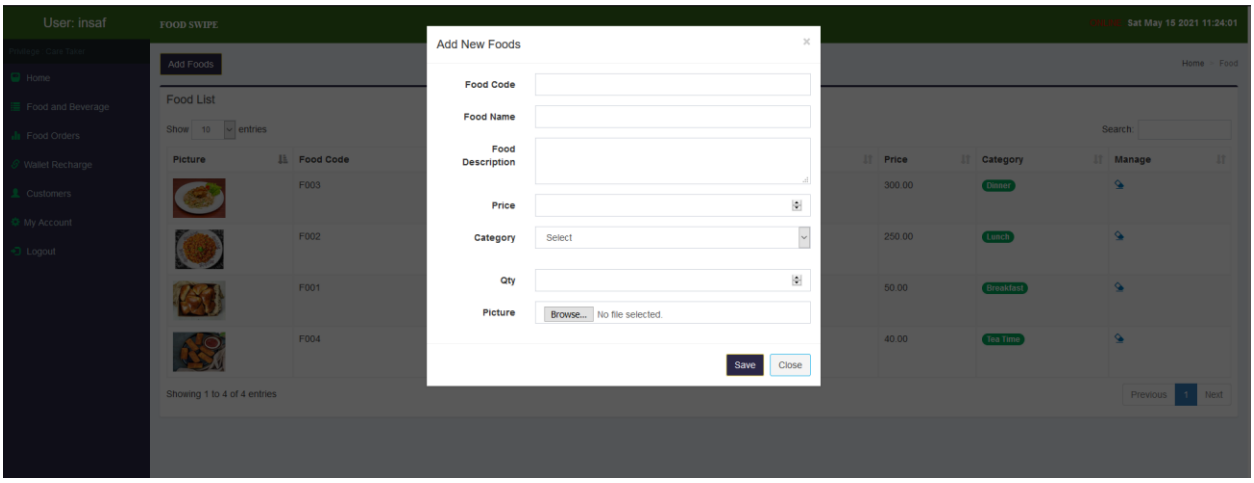


Figure 3.8 : Add New Foods Screen

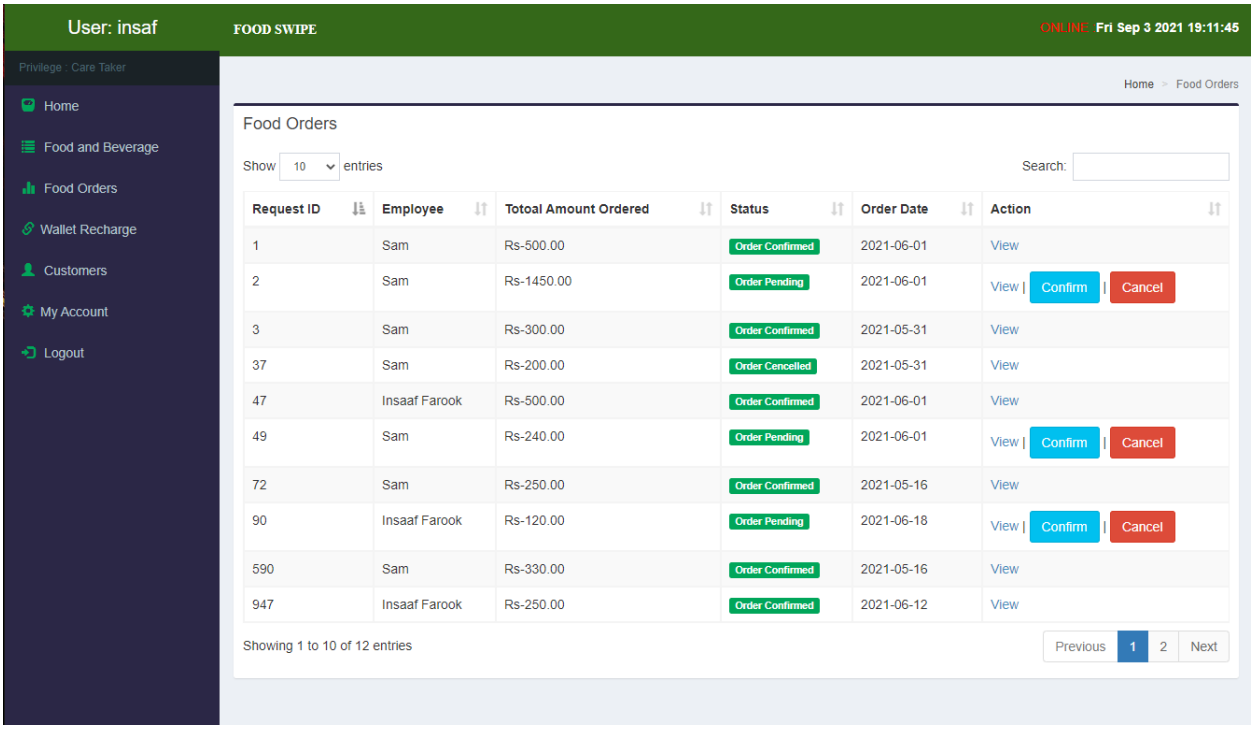


Figure 3.9 : Confirm order screen

The caretaker can view the order placed by the customer and Confirm or cancel the order. The caretaker will be requested for confirmation in the event of order cancellation. Based on the order action, the status of the order will be updated, which will also reflect the customer.

The below figures illustrate the screens used by the customer to place orders through the Food Swipe application. Figure 3.10 below displays the menu screen, which will contain item details along with the product image based on the category filter selected. Customers can add the quantity of the item in the textbox and click on the 'Add to Cart' button, which will add the item to the order. Figure 3.11 illustrates the checkout page, which will list out the products added to the order along with the order total. Customers can select the payment option and proceed to payment.

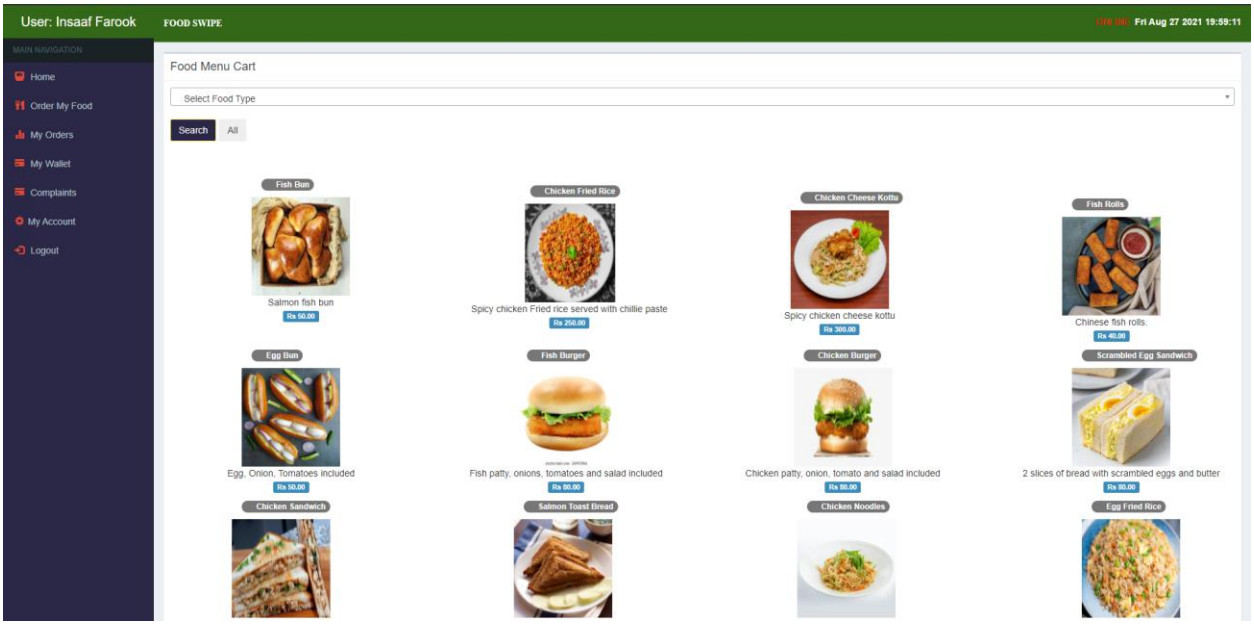


Figure 3.10 : Menu Screen

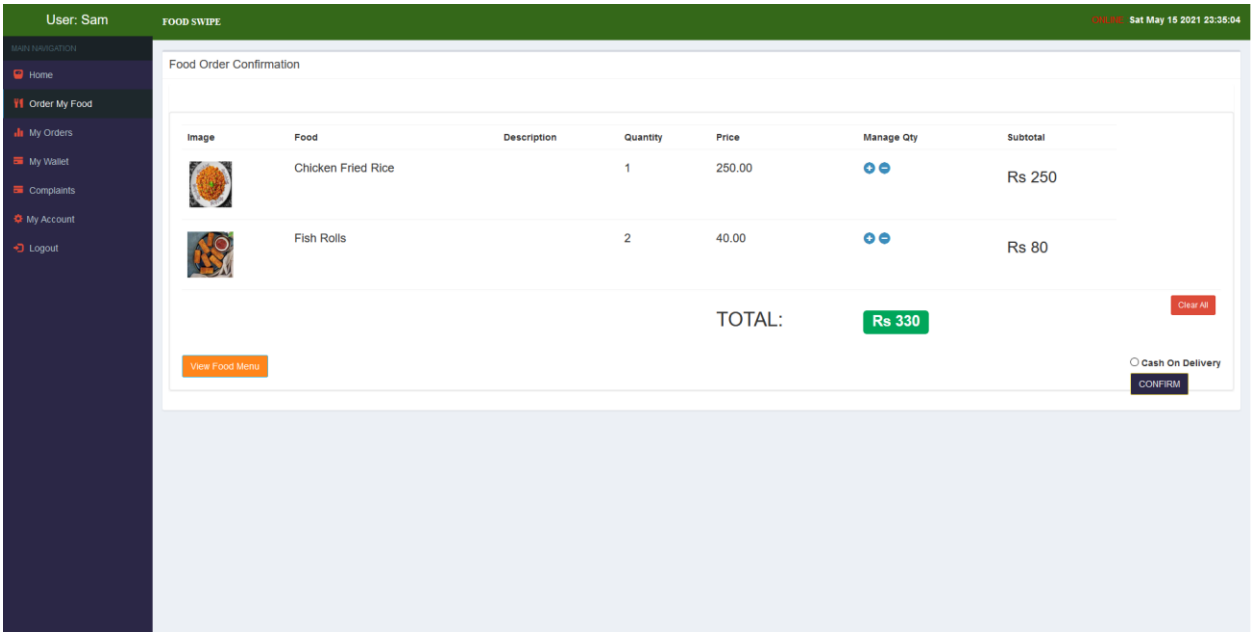


Figure 3.11 : Checkout Screen

Customers can view the recent orders placed via the 'My Ordered Food' tab, as illustrated in Figure 3.12.

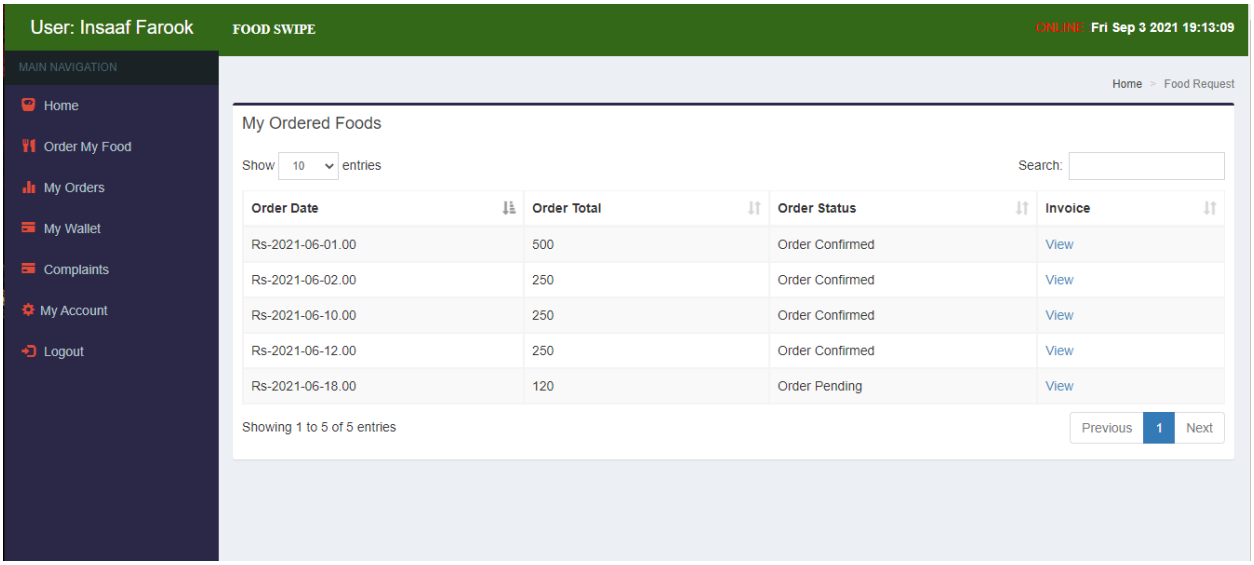


Figure 3.12 : My Ordered Food screen

Figure 3.13 below illustrates the screen that customers use to complain about the canteen that the admin can view.

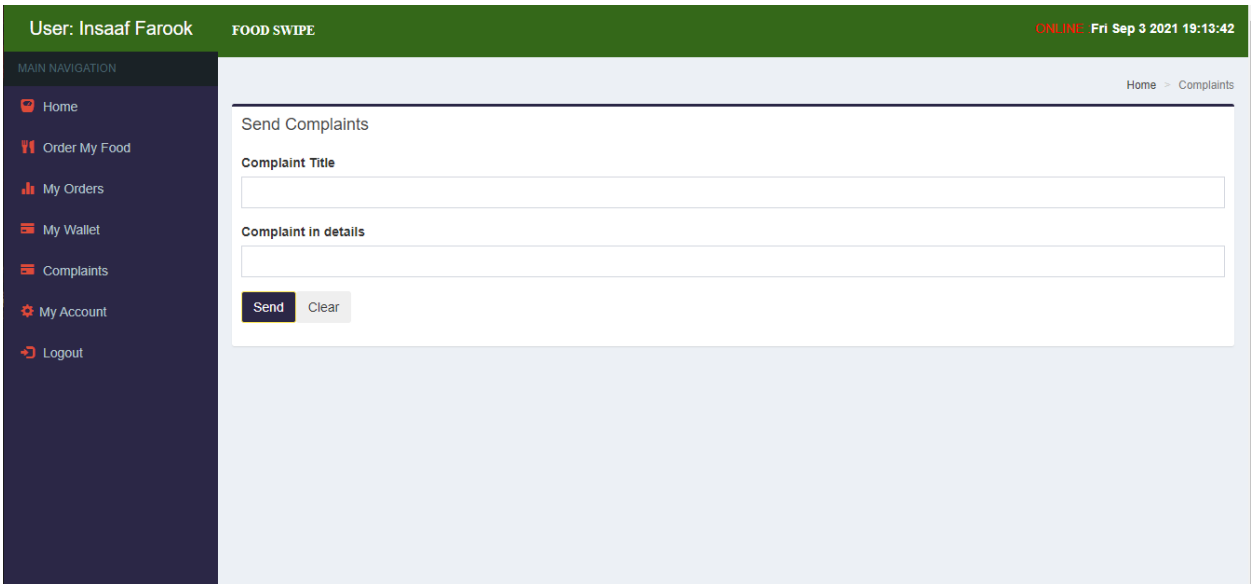


Figure 3.13 : Send Complaint screen

Admin users will have a screen similar to Figure 3.14 to view the complaints logged by the customers.

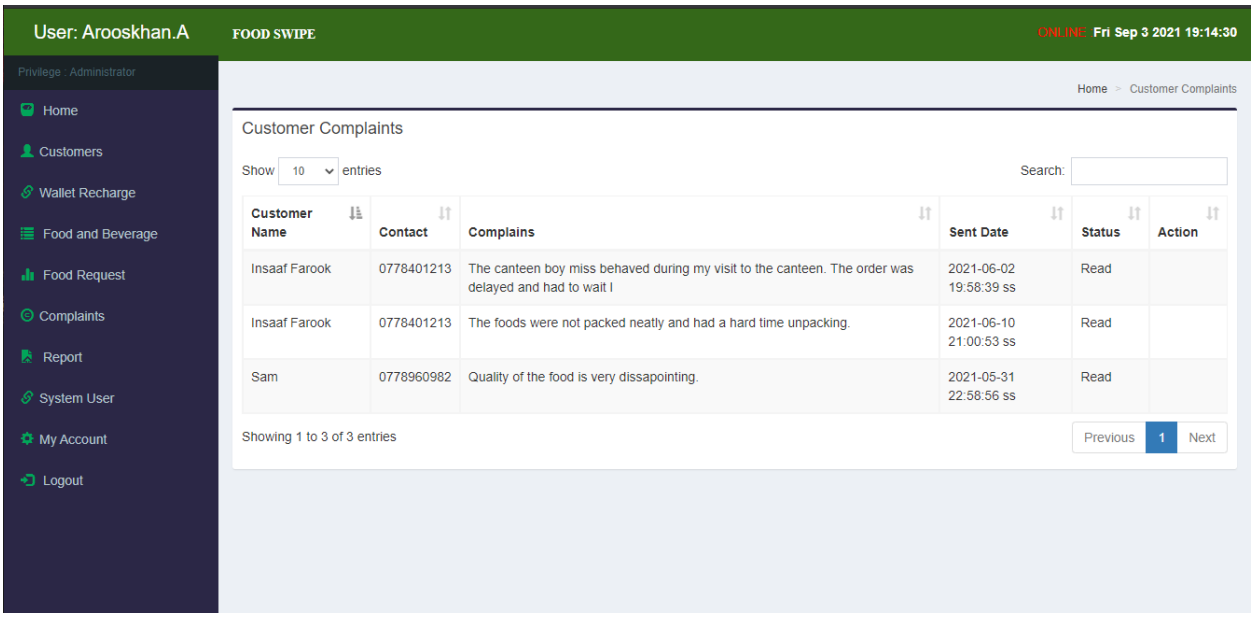


Figure 3.14 : Customer Complaint screen

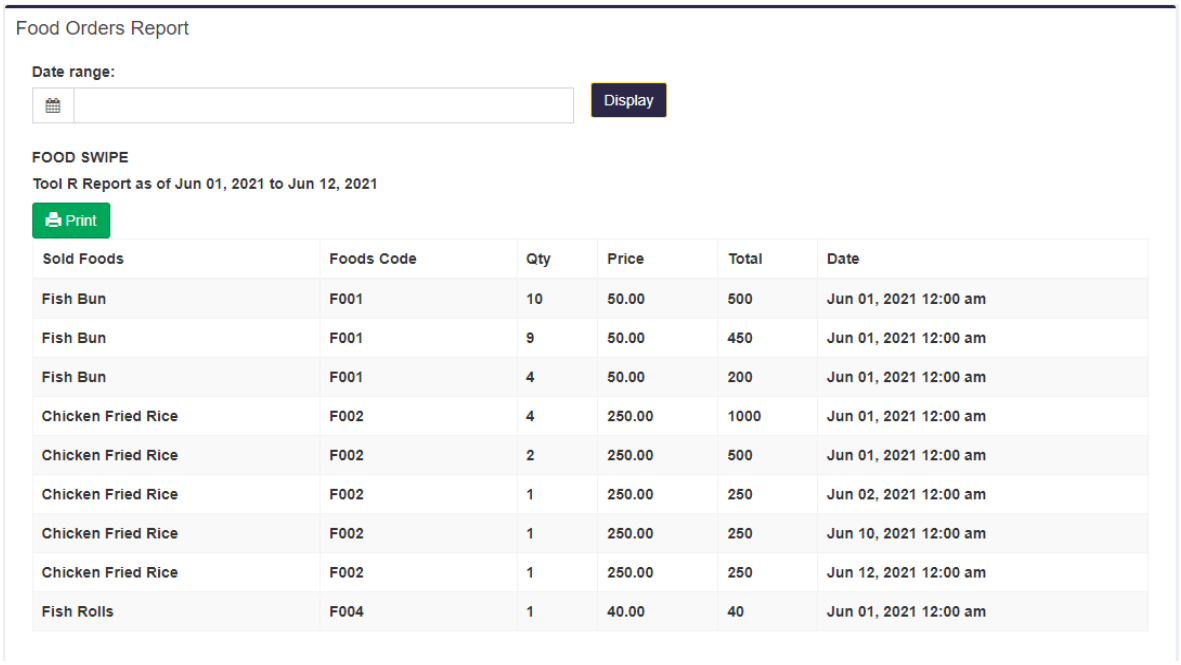
The remaining screenshots of the application screens are included in appendix A.

3.2.5 MIS Reports

The MIS module aims to extract data from the sources and derive insights that drive business growth. When considering the Food Swipe, all the data is being stored in the SQL database, including customer details, stock details of the foods available, the orders placed, etc. These details can be derived and represented in a meaningful way to help the business decision-makers take critical business decisions. Below are some reports that can be derived from the application by the admin user.

Food Order Report

Figure 3.15 below illustrates the visual template of the Food Order report that the admin user can derive from the application. Admin can select various timelines for the reports to be generated. Daily, Weekly, Monthly or Annual reports can be generated, and the report contains information on the items purchased and the quantities. Based on the reports, the admin can figure out the top selling products, lease selling products, and revenue generated from each product.



Sold Foods	Foods Code	Qty	Price	Total	Date
Fish Bun	F001	10	50.00	500	Jun 01, 2021 12:00 am
Fish Bun	F001	9	50.00	450	Jun 01, 2021 12:00 am
Fish Bun	F001	4	50.00	200	Jun 01, 2021 12:00 am
Chicken Fried Rice	F002	4	250.00	1000	Jun 01, 2021 12:00 am
Chicken Fried Rice	F002	2	250.00	500	Jun 01, 2021 12:00 am
Chicken Fried Rice	F002	1	250.00	250	Jun 02, 2021 12:00 am
Chicken Fried Rice	F002	1	250.00	250	Jun 10, 2021 12:00 am
Chicken Fried Rice	F002	1	250.00	250	Jun 12, 2021 12:00 am
Fish Rolls	F004	1	40.00	40	Jun 01, 2021 12:00 am

Figure 3.15 : Food Orders Report

Food Stock Report

Figure 3.16 below illustrates the visual template of the Food Stock report along with the unit prices and the total value of each product. The report allows the admin to figure out what food items need to be immediately restocked and what food items will be returned on the business day.

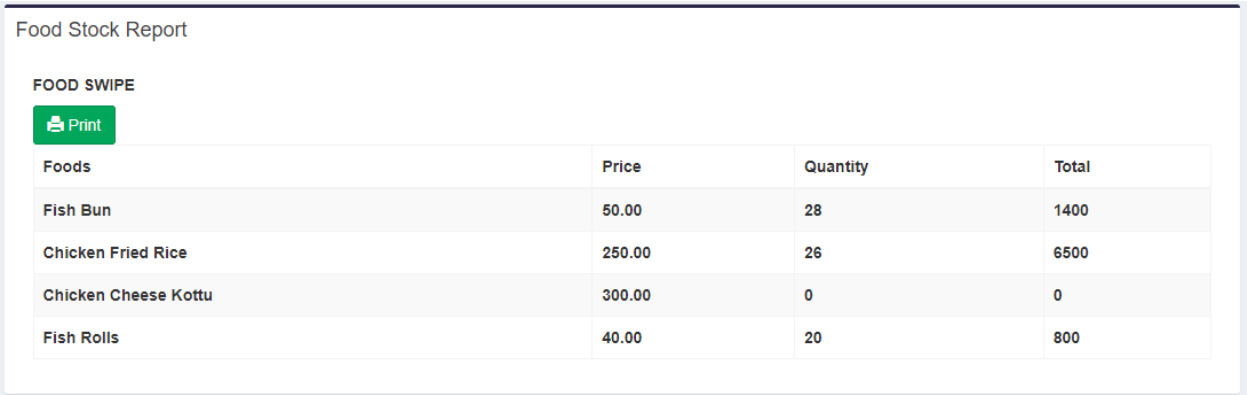


Figure 3.16 : Food Stock Report

Customer Report

Figure 3.17 below illustrates the customers' report's visual template, which allows the admin to check the currently active users that are using the application and their wallet balances.

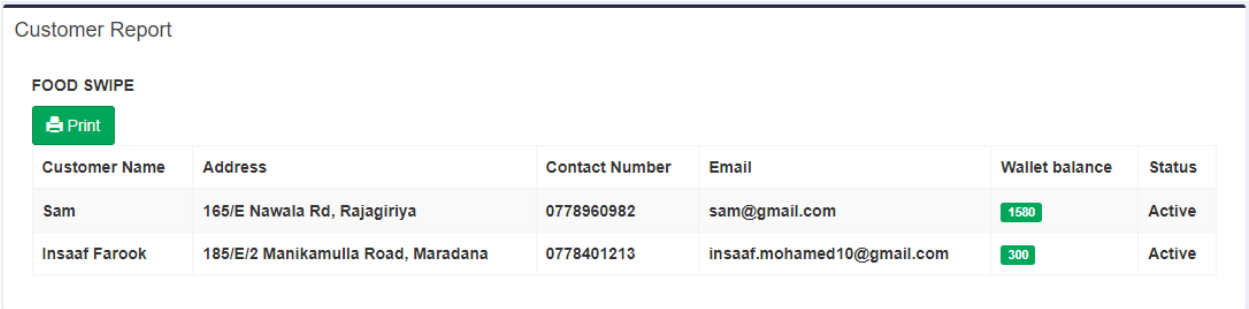


Figure 3.17 : Customer Report

4. Implementation

4.1 Introduction

In this chapter, the implementation details such as tools and technology stacks that will be used to develop the application, significant codes, codes reused in the application, and implementation environment-related details will be discussed here.

4.2 Development Tools & Technologies

4.2.1 Development Tools

Visual Studio Code will be used as the development IDE. Visual Studio Code is a freeware source-code editor made by Microsoft for Windows, Linux and macOS. It has been redefined and optimized for building and debugging modern web and cloud applications.

Xampp will be used as the local test server for the application development. Xampp is a cross-platform, free and open-source, web server solution, which developed by Apache Friends, consisting mainly of the Apache HTTP Server, MySQL database, and interpreters. Most of the actual web server deployments use the same components as XAMPP; hence it makes transitioning from a local test server to a live server easy.

4.2.2 Server-Side Technology

The PHP Hypertext Preprocessor (PHP) is selected as the server-side scripting language for the proposed application. PHP is a popular general-purpose scripting language that is especially suited to web development that allows creating dynamic content that interacts with databases. PHP has below beneficial features that would be helpful during the implementation of the application.

- PHP supports a large number of major protocols such as POP3, IMAP, and LDAP.
- PHP is integrated with several popular databases, including MySQL, PostgreSQL, Oracle, and Microsoft SQL Server.

- PHP is compatible with a large majority of operating systems, including UNIX, Solaris, and Linux.
- PHP is cost-efficient since it is an open-source web language; hence it is entirely free.

4.2.3 Client-Side Technology

HyperText Markup Language (HTML), JavaScript (JS) and Cascading Style Sheet (CSS), will be used as the client-side scripting language. HTML allows creating and structuring sections, paragraphs, headings, links, and block quotes for web pages and applications. CSS describes how HTML elements are to be displayed on the screen. CSS saves much work by controlling the layout of multiple web pages all at once. JavaScript is a scripting or programming language that allows the implementation of complex features on web pages. JS provides the capability to dynamically update content, control multimedia, and animate images.

4.2.4 Data Management and Storage Technology

Structured Query Language, commonly known as SQL, will be used as the Database Management System. SQL is a standard programming language for relational databases, and it is the most widely implemented database language. SQL can be used to share and manage data, particularly data found in relational database management systems, including data organized into tables. Below are some beneficial features of SQL that would bring value to the application.

- Faster query processing where large amounts of data can be retrieved quickly and efficiently.
- Operations like Insertion, deletion, manipulation of data is also done in almost no time.
- SQL maintains well-defined standards.
- High data security.

4.3 Reusable Plugins

Bootstrap is the main plugin used in the development of the Food Swipe application, which eases the development of responsive websites. Bootstrap is a free, open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

4.4 Major Codes

Major code segments in the Food Swipe application are discussed below in detail. Figure 4.1 illustrates the code segment of the login. Based on the user inputs, data will be fetched from the database, and according to the response received, either the user will be directed to the home page or an error message will be displayed.

```php
<?php
include('dist/includes/dbcon.php');
    if(isset($_POST['login'])){
        $user_unsafe=$_POST['username'];
        $pass_unsafe=MD5($_POST['password']);
        $user = mysqli_real_escape_string($con,$user_unsafe);
        $pass1 = mysqli_real_escape_string($con,$pass_unsafe);
        date_default_timezone_set('Asia/Manila');

        $date = date("Y-m-d H:i:s");

        $query=mysqli_query($con,"select * from customer where email='$user' and password='$pass1' and status='Active'")or die(mysqli_error($con));
        $row=mysqli_fetch_array($query);


        $id=$row['cid'];
        $name=$row['Name'];
        $_SESSION['id']=$id;
        $_SESSION['name']=$name;
        $_SESSION['UserType']='customer';
        $counter=mysqli_num_rows($query);

        if ($counter == 0){
          echo "<script type='text/javascript'>alert('Invalid Username or Password!');
          document.location='customerhome.php'</script>";
        }
        elseif ($counter > 0){
            echo "<script type='text/javascript'>document.location='Customer/home.php'</script>";
        }
    }
?>
```

Figure 4.1 : Login Code Snippet

Figure 4.2 and 4.3 below illustrates the email validation function and phone number validation function used during the customer sign up. A proper error message will be thrown in case of an invalid email address and phone number.

```
function ValidateEmail(inputText) {
  var mailformat = /^\w+([\.-]?\w+)*@\w+([\.-]?\w+)*(\.\w{2,3})+$/;
  if(inputText.value.match(mailformat)){
    document.form1.text1.focus();
    return true;
  }
  else {
    alert("You have entered an invalid email address...!\nRetype the correct email address...\n"+
      "Hint:\n Some valid email formats are...\n a@b.cd, ab-cd@ef.gh, ab.cd@ef.ghi, abc_def@mail.com");
    document.getElementById("txtEmail").value="";
    document.form1.text1.focus();
    return false;
  }
}
```

Figure 4.2 : Email Validation Code Snippet

```
15    function phonenumber(){
16      var phoneno = /^\d{10}$/;
17      if(document.getElementById("txtTell").value==""){
18
19      }
20      else {
21        if( document.getElementById("txtTell").value.match(phoneno)){
22          hand();
23        }
24        else{
25          alert("Enter 10 digit Mobile Number");
26          document.getElementById("txtTell").value="";
27          document.getElementById("txtTell").focus()=true;
28          return false;
29        }
30      }
31    }
32
33    function hand(){
34      var str = document.getElementById("txtTell").value;
35      var res = str.substring(0, 2);
36      if(res=="07"){
37        return true;
38      }
39      else{
40          alert("Enter 10 digit of Mobile Number start with 07xxxxxxxx");
41          document.getElementById("txtTell").value="";
42          document.getElementById("txtTell").focus()=true;
43          return false;
44      }
45    }
46
```

Figure 4.3 : Phone Number Validation Code Snippet

Figure 4.4 illustrates the Customer registration PHP code segment. Based on the input, the code will validate whether there is an existing account with the mobile number entered and if so, an error message will be displayed for the customer. Else the account details will be stored in the database. Passwords will be encrypted by MD5 before stored in the database.

```php
<?php
session_start();
include('/dist/includes/dbcon.php');


    $Name = $_POST['Name'];
    $Address = $_POST['Address'];
    $txtTell = $_POST['txtTell'];
    $email = $_POST['email'];
    $Password2 = MD5($_POST['Password']);

    {
        $query2=mysqli_query($con,"select * from customer where contact='$txtTell' ")or die(mysqli_error($con));
        $count=mysqli_num_rows($query2);

        if ($count>0){
            echo "<script type='text/javascript'>alert('Mobile Number already exist!');</script>";
            echo "<script>document.location='customer.php'</script>";
        }
        else{
            mysqli_query($con,"INSERT INTO  customer(Name,address,contact,email,Password)
            VALUES('$Name','$Address','$txtTell','$email','$Password2')")or die(mysqli_error($con));

            echo "<script type='text/javascript'>alert('Successfully registered you may login now !');</script>";
            echo "<script>document.location='customerhome.php'</script>";
        }
    }
?>
```

Figure 4.4 : Customer Registration Code Snippet

Figure 4.5 illustrates the code segment used to display the payment options available for the customer during the checkout. Suppose the customer's wallet balance is greater than the total customer shown the wallet payment option. Else the customer will have only the Cash on Delivery option enabled.

```php
<div class='pull-right'>
    <?php
        $id=$_SESSION['id'];
        $query=mysqli_query($con,"select * from customer where walletbalacne > '$total' and cid ='$id'  ")or die(mysqli_error());
        while($row=mysqli_fetch_array($query)){

    ?>

    <p>Please select your payment methode:</p>
    <input type="radio" id="Wallet" name="paymenthode" value="Wallet">
    <label for="male">Pay From My Wallet</label><br>
    <?php }
    ?>

    <input type="radio" id="Cash" name="paymenthode" value="Cash">
    <label for="female">Cash On Delivery  </label><br>

```

Figure 4.5 : Wallet Payment Handle Code Snippet

Figure 4.6 illustrates the code segment of the Food order report. Based on the date filter selected in the application's front-end, the data will be queried from the database. It will be displayed in tabular format as displayed in Figure 3.15 in the above chapter. This can be printed as a hard copy if required. Window.print() function is used, which prints the content of the current window.

```
    <?php
      if (isset($_POST['display'])) {
        $date=$_POST['date'];
        $date=explode('-',$date);
        $start=date("Y/m/d",strtotime($date[0]));
        $end=date("Y/m/d",strtotime($date[1]));
    ?>

    <div class="col-md-12">
    <?php
      include('../dist/includes/dbcon.php');
    ?>
          <h5><b>FOOD SWIPE </h5>
          <h5><b>Tool R Report as of <?php echo date("M d, Y",strtotime($start))." to ".date("M d, Y",strtotime($end));?></b></h5>
          <a class = "btn btn-success btn-print" href = "" onclick = "window.print()">
            <i class ="glyphicon glyphicon-print"></i>
            Print
          </a>


        <table id="example1" class="table table-bordered table-striped">
          <thead>
            <tr>
              <th>Sold Foods</th>
              <th>Foods Code</th>
              <th>Qty</th>
              <th>Price</th>
              <th>Total</th>
              <th>Date  </th>

            </tr>
          </thead>
        <tbody>

    <?php
       $query=mysqli_query($con,"select * from transaction  join transaction_detail on transaction.transaction_id=transaction_detail.transaction_id  join  product  on
         transaction_detail.product_id=product.prod_id  join  customer on  transaction.customerid=customer.cid     where date(order_date)>='$start' and date(order_date)<='$end'  ")or die(mysqli_error($
         con));

       $qty=0;
       while($row=mysqli_fetch_array($query)){
        $total=$row['order_qty']*$row['prod_price'];
    ?>
          <tr>
            <td><?php echo $row['prod_name'];?></td>
            <td><?php echo $row['serial'];?></td>
            <td><?php echo $row['order_qty'];?></td>
            <td><?php echo $row['prod_price'];?></td>
            <td><?php echo "$total"; ?> </td>
            <td><?php echo date("M d, Y h:i a",strtotime($row['order_date']));?></td>
    <?php
    }
  }
  ?>
```

Figure 4.6 : Report Generation Code Snippet

5. Testing and Evaluation

5.1 Introduction

It is expected that the implemented system will be tested and evaluated with an emphasis on both functional and non-functional requirements. This chapter will detail the testing performed on the developed system, including test cases, test results, and user evaluation.

5.2 Testing Overview

Software testing is an important stage in the lifecycle of software development (SDLC). It can aid in time and cost savings by finding errors and faults early in the development process. Testing provides an endless amount of advantages and bonuses. Testing assists in identifying and preventing bugs. Permits the construction of bug-free, high-quality, and effective software applications. Additionally, creating effective and efficient software results in an enhanced user experience and greater customer service.

Food Swipe is tested to ensure that it meets both functional and non-functional standards. Black-box testing will be used to validate the application's end-to-end flow by simulating end-user behaviors and anticipating appropriate responses based on the requests. The requirements phase begins with gathering needs, followed by test planning, during which risk and mitigation methods are defined. Test cases are produced and executed to cover the requirements specified in the criteria. Any faults or defects detected during the process are fixed and retested. Several black box testing approaches, including Equivalence Partitioning, Boundary Value Analysis, and Error Guessing, are assessed during the test design process.

Additionally, non-functional testing is conducted via black-box testing. A non-functional test is used to assess "how" a program does a particular task, rather than "if" the software is capable of performing that task. The following sections delve into detail about the test cases and their accompanying outcomes.

5.3. Test Cases and Test Results

5.3.1 Functional Test Cases

Below are some critical functional tests that are designed. All of these tests are being executed, and the test results are documented.

Table 5.1 : Customer Registration Test Case

Test Case ID :	T001	**Test Description :**	Verify new customer can register to the application
Prerequisites :		**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to the login screen.		N/A	1. User should be navigated to the login screen.
2. Click on the "Register as a New customer here" link.		N/A	2. User should navigate to the Registration page.
3.1. Enter valid inputs to the fields except Mobile number.		Mobile Number: 1234356	3.1. Error message should be populated as displayed in Figure 5.1.
3.2. Enter valid inputs to the fields except email address.		Email Address: asd123.com	3.2. Error Message should be displayed.
3.3. Click on the clear button.		N/A	3.3. Fields should be cleared.
3.4. Enter valid input to the fields.		Name: Sam Address: 153/E, Rajagiriya Rd, Nawala. Mobile: 0776452341 Email: sam@gmail.com Password: p@ssword	3.4. User should be successfully Registered. Success message should be populated as displayed in Figure 5.2.

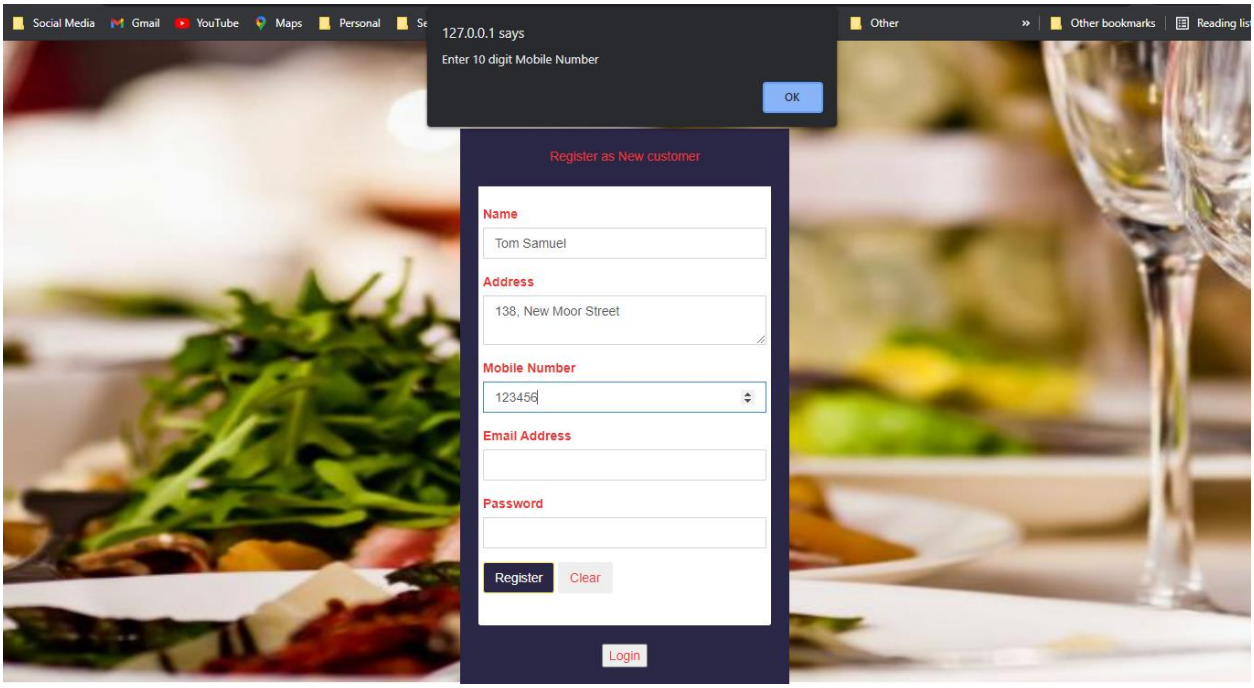


Figure 5.1 : Incorrect Mobile Number Error Message

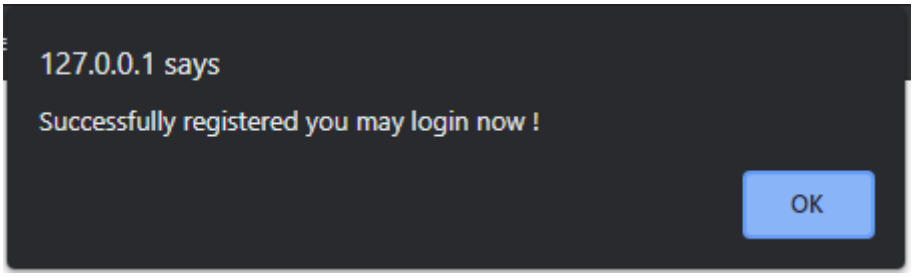


Figure 5.2 : Customer Registration Success Message

Table 5.2 : User Login Test Case

Test Case ID :	T002	**Test Description :**	Verify user can login to the application (Customer/ Admin/ CareTaker)
Prerequisites :	There should be a registered customer, admin and caretaker	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to the login screen.		N/A	1. User should be navigated to the login screen.
2.1. Enter incorrect username and correct password then click on Login		UN: Sam PW: p@ssword	2.1. Error message should be populated as displayed in Figure 5.3.
2.2. Enter correct username and incorrect password then click on Login		UN: sam@gmail.com PW: asd123	2.2. Error message should be populated as displayed in Figure 5.3.
2.3. Enter incorrect username and incorrect password then click on Login		UN: Sam PW: asd123	2.3. Error message should be populated as displayed in Figure 5.3.
3. Enter correct username and password then click on Login		UN: sam@gmail.com PW: p@ssword	3. User should be successfully logged in successfully.

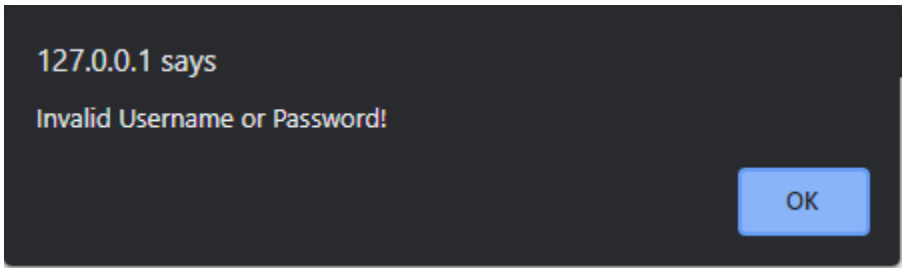


Figure 5.3 : Incorrect Username or Password Message

Table 5.3 : Customer Place Order Test Case

Test Case ID :	T003	**Test Description :**	Verify customer can place a cash order
Prerequisites :	Customer should have logged into the application.	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Click on “Order My Food”		N/A	1. Available foods should be shown.
2. Click on a food item.		N/A	2. Food view page should be loaded.
3. Click on ‘Process’ button		N/A	3. Food Order confirmation screen should be loaded.
4. Update the quantity of item added		N/A	4. Quantity should be updated successfully.
5. Select payment type as ‘Cash’		N/A	5. User should be allowed to select the payment option.
6. Click Confirm button		N/A	6. Order should be successfully submitted and cart should be reverted to empty.
7. Navigate to My Orders and verify whether the order is listed and order status is ‘Order Pending’		N/A	7. Orders should be displayed and status should be ‘Order Pending’ as shown in Figure 5.4.

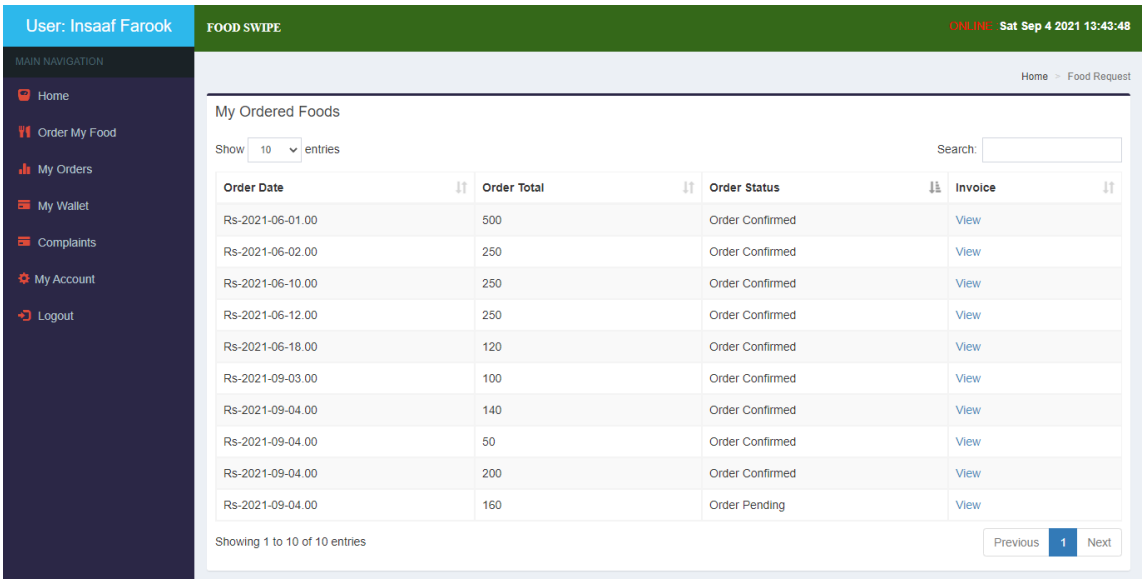


Figure 5.4 : Order Pending Status

Table 5.4 : Payment Method Validation Test Case

Test Case ID :	T004	**Test Description :**	Verify the payment method selection
Prerequisites :	Customer should have logged into the application. Customers should have wallet balance.	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Create an order exceeding the wallet balance		N/A	1. Wallet payment option should not be available.
2. Decrease the order value where order total less than wallet balance		N/A	2. Wallet payment option should be available as displayed in Figure 5.5.
3. Increase the order value where order total is greater than wallet balance		N/A	3. Wallet payment option should not be available as displayed in Figure 5.6.
4. Decrease the order value where order total equal to wallet balance		N/A	4. Wallet payment option should be available as displayed in Figure 5.5.

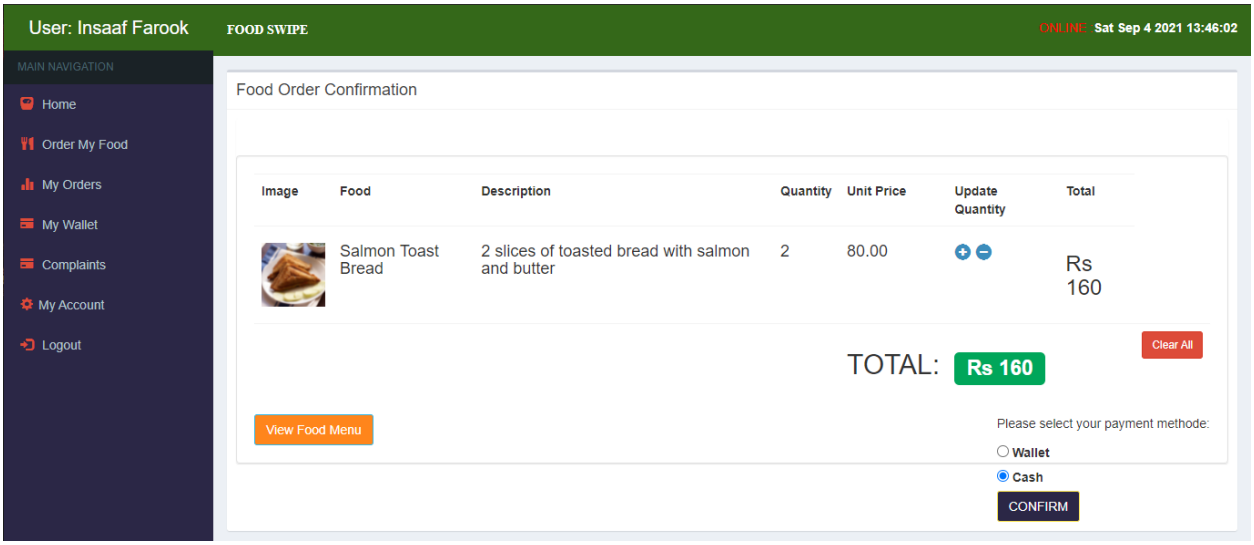


Figure 5.5 : Wallet Payment Available

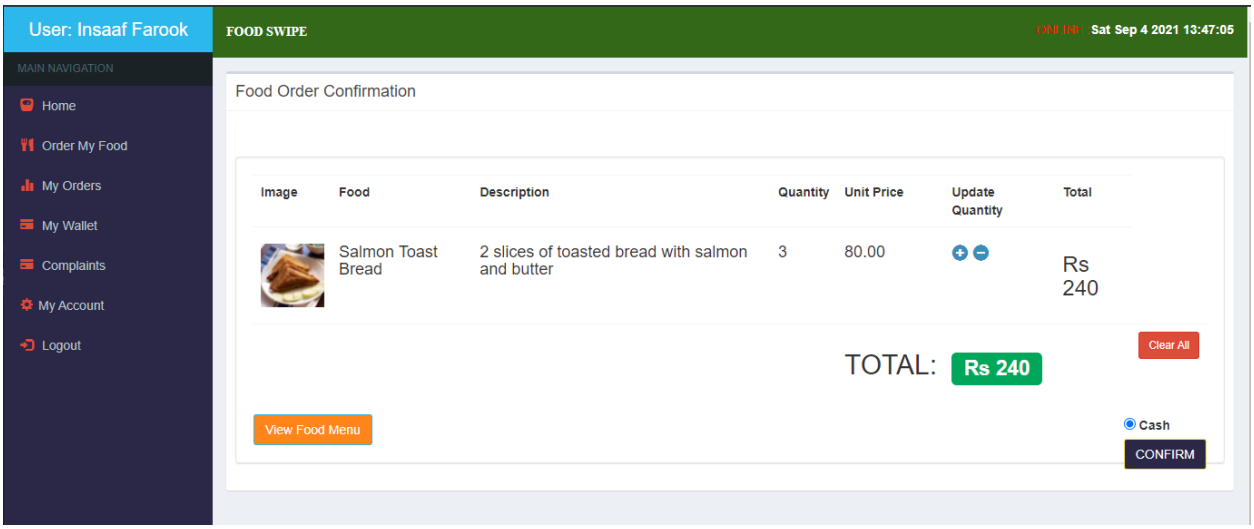


Figure 5.6 : Wallet Payment not available

Table 5.5 : Item Quantity Validation Test Case

Test Case ID :	T005	**Test Description :**	Verify the food item cannot be order if the quantity is not available.
Prerequisites :	Customer should have logged into the application.	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to Food Menu page by clicking Order My Food		N/A	1. All products should be displayed.
2. Update the quantity of an item to 0 from the Database		Fish Rolls	2. Database update should be successful.
3. Select the item that the quantity is updated.		N/A	3. Product details page should be displayed.
4. Verify whether the 'Add Product' button is displayed.		N/A	4. Add Product button should not be displayed and the user should not be allowed to order the item as illustrated in Figure 5.7.
5. Update the quantity of an item greater than 0 from the Database		Fish Rolls	5. Database update should be successful.
6. Select the item that the quantity is updated.		N/A	6. Product details page should be displayed.
7. Verify whether the 'Add Product' button is displayed.		N/A	7. Add Product button should be displayed and the user should be allowed to order the item as illustrated in Figure 5.8

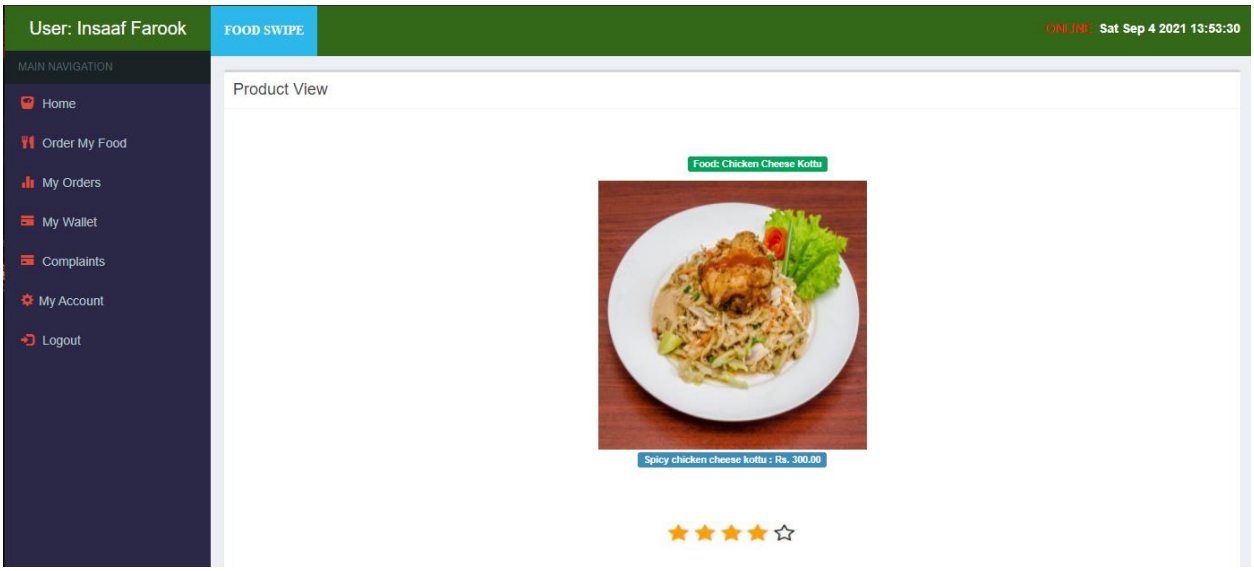


Figure 5.7 : Add Product button not available

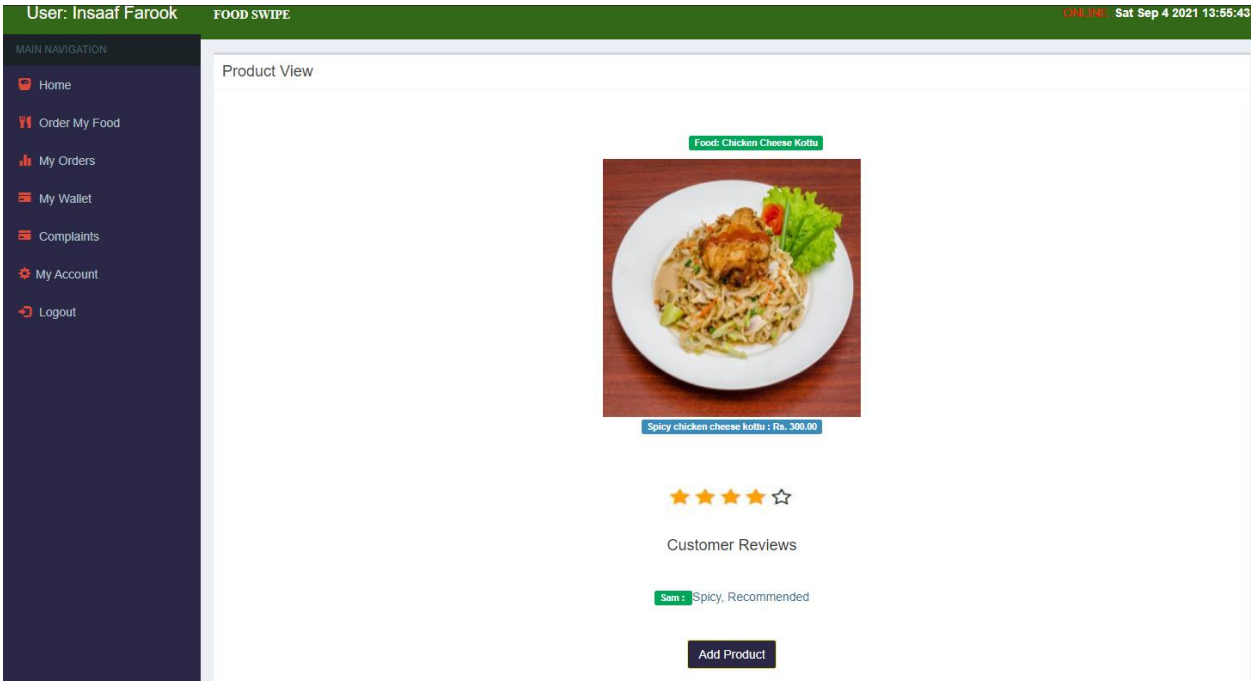


Figure 5.8 : Add Product button available

Table 5.6 : Food Rating and Review Test Case

Test Case ID :	T006	**Test Description :**	Verify user can rate and review a food item can
Prerequisites :	Customer should have logged into the application.	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to Food Menu page by clicking Order My Food		N/A	1. All products should be displayed.
2. Select an item from the menu		Fish Rolls	2. Product details page should be displayed.
3. Select a star rating between 1 - 5.		N/A	3. Should be able to select a rating.
4. Enter some values to the review feild/		N/A	4. Should be able to add the review.
5. Click on submit review button		N/A	5. Order review should be successful..
6. Click ok for successful message		N/A	6. Should navigate to the Food menu screen.
7. Select the same item from the menu and verify the review		Fish Rolls	7. The review added by the customer should be available as displayed in Figure 5.9.

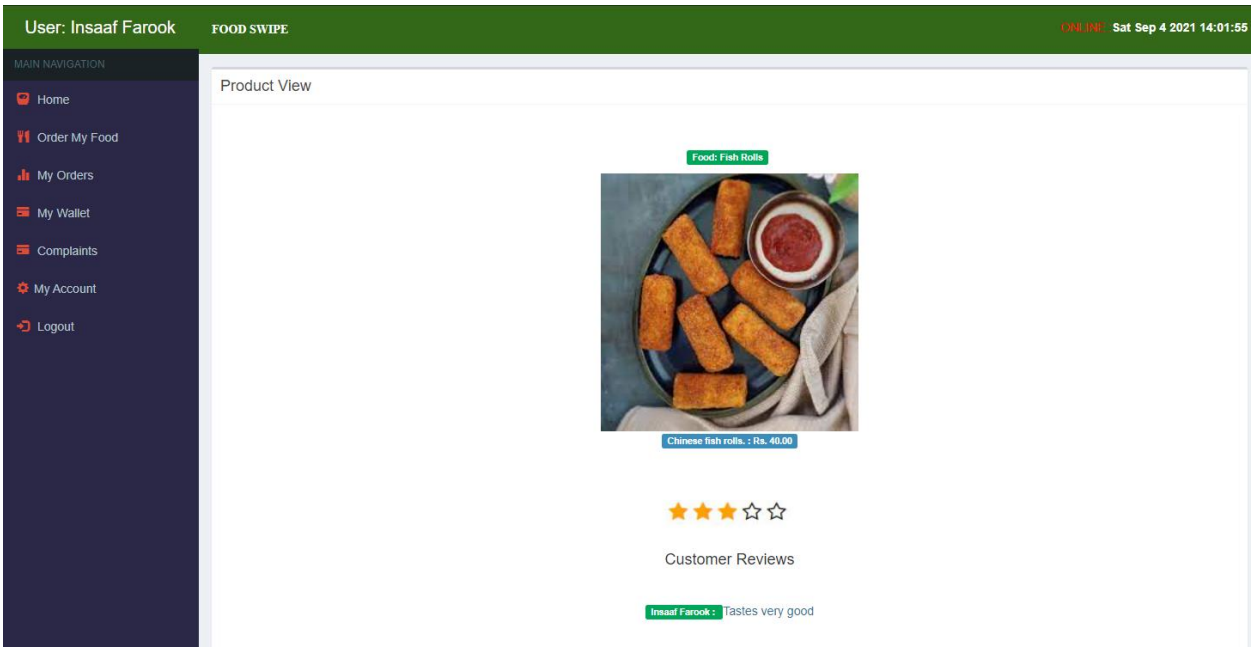


Figure 5.9 : Customer Review

Table 5.7 : Log Complaints Test Case

Test Case ID :	T007	**Test Description :**	Verify user can log a complaint
Prerequisites :	Customer should have logged into the application.	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to Send Complaint by clicking Complaints → Send Complaint		N/A	1. The Send complaints page should be displayed.
2. Enter values for ‘title’ and ‘Message in details' fields.		Title: Poor Quality Msg: Quality of the food is very disappointing.	2. Should be able to enter values.
3. Click on Send button		N/A	3. ‘Successfully sent’ message should be.
4. Click ok for successful message		N/A	4. Should navigate the user to the Send Complaints screen.

Table 5.8 : Add New Food Type Test Case

Test Case ID :	T008	**Test Description :**	Verify admin can add new food type
Prerequisites :	Admin should have logged into the application.	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to Add New Food Type screen by clicking ‘Food and Beverages’ → ‘Food Type’		N/A	1. Add New Food Type s page should be displayed.
2. Enter values for ‘Code’ and ‘Food Type' fields.		Code: BF Food Type: Breakfast	2. Should be able to enter values.
3. Click on Save button		N/A	3. ‘Successfully Save’ message should be.
4. Verify the Food Type is available in List below		N/A	4. Added Food Type should be listed in the list as displayed in Figure 5.10.

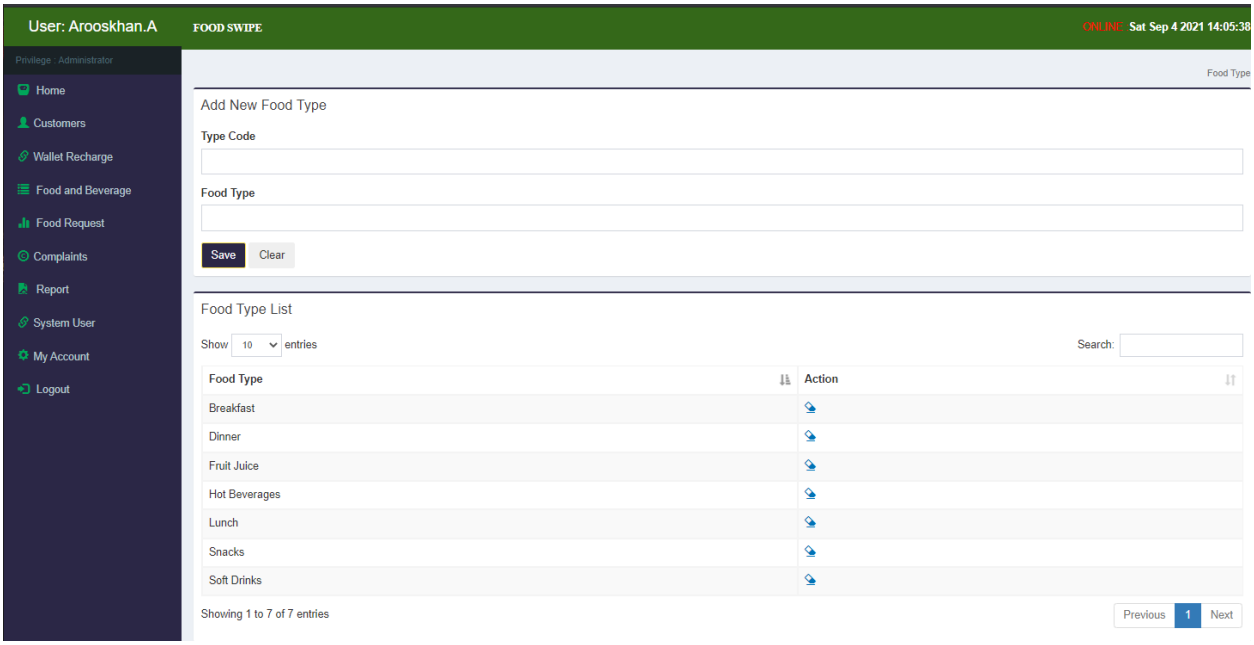


Figure 5.10 : Food Types list

Table 5.9 : Update Food Type Test Case

Test Case ID :	T009	**Test Description :**	Verify admin can modify food type
Prerequisites :	Admin should have logged into the application..	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to Add New Food Type screen by clicking 'Food and Beverages' → 'Food Type'		N/A	1. Add New Food Types page should be displayed.
2. Click on the edit icon of the Food Type that needed to be updated		Food Type: Breakfast	2. Update Food Type Details modal should be displayed.
3. Clear the existing value and enter new value		Food Type: Lunch	3. Should be able to enter values
4. Click on 'Save' button		N/A	4. 'Successfully Save' message should be.

Table 5.10 : : Add New Food Test Case

Test Case ID :	T010	**Test Description :**	Verify the user can add new food. (Admin, Caretaker)
Prerequisites :	User should have logged into the application..	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to Add New Food Type screen by clicking 'Food and Beverages' → 'Foods'		N/A	1. The Food List page should be displayed.
2. Click on the Add Food button.		N/A	2. Add new Foods modal should be displayed.
3. Enter values for the fields.		Food Code: F035 Food Name: 7up 100ml Food Description: Chilled and Un-chilled bottles are available Price: 100.00 Category: Soft Drinks Qty: 60 Picture: img.png	3. Should be able to enter values
4. Click on 'Save' button		N/A	4. Should be saved successfully and listed in the Product List as displayed in Figure 5.11.

Table 5.11 : Update Food Item Test Case

Test Case ID :	T011	**Test Description :**	Verify the user can modify a food item. (Admin, Caretaker)
Prerequisites :	User should have logged into the application..	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to Add New Food Type screen by clicking 'Food and Beverages' → 'Foods'		N/A	1. The Food List page should be displayed.

2. Click on the Edit icon relevant to the food item that to be updated.	N/A	2. Update Food Details modal should be displayed.
3. Clear the populated values and enter new values for the fields.	Food Code: F035 Food Name: 7up 100ml Food Description: Chilled and Un-chilled bottles are available Price: 100.00 Category: Soft Drinks Qty: 33 Picture: img.png	3. Should be able to enter values
4. Click on 'Save' button	N/A	4. Should be saved successfully and listed in the Product List as displayed in Figure 5.11.

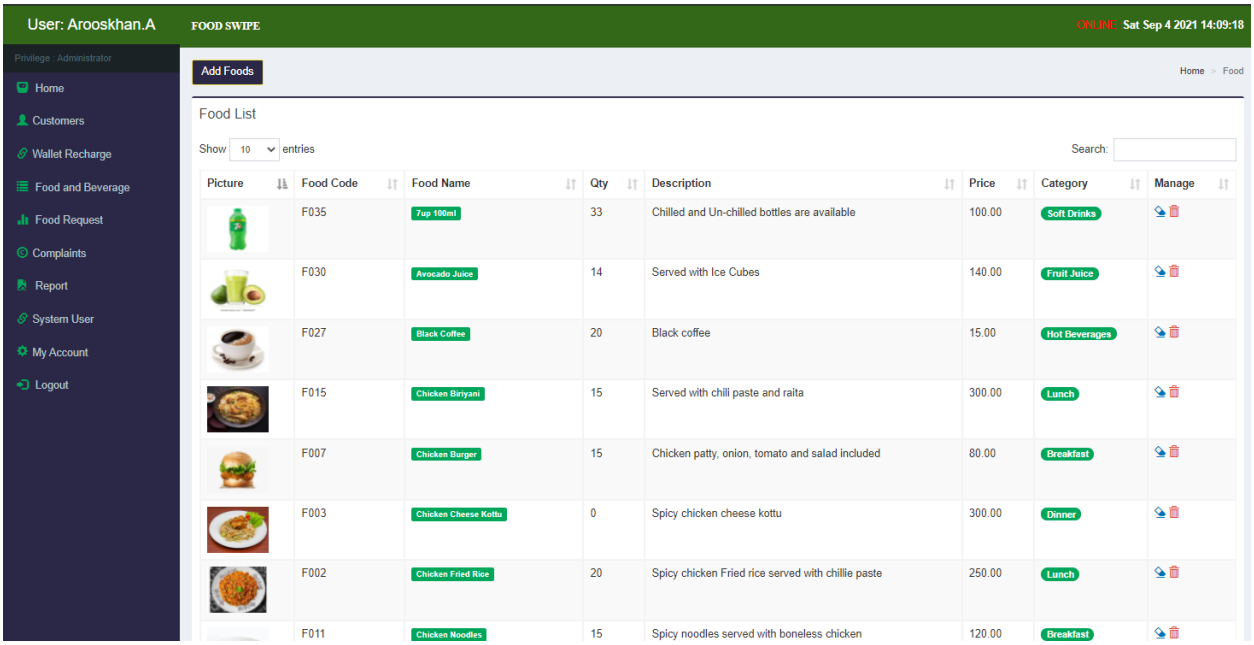


Figure 5.11 : Food List Screen

Table 5.12 : Order Confirmation Test Case

Test Case ID :	T012	**Test Description :**	Verify the user can Confirm Pending Order (Admin, Caretaker)
Prerequisites :	1. User should have logged into the application. 2. Customer should have placed an order.	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to Orders screen by clicking 'Food Request'		N/A	1. The Order List page should be displayed.
2. Click on the Confirm Order button .		N/A	2. Confirmation modal should be displayed with "Are you sure" text .
3. Click on cancel button from the modal		N/A	3. Modal should be closed and order should remain in Pending state
4. Click on the Confirm Order button		N/A	4. Confirmation modal should be displayed with "Are you sure" text
5. Click on Ok button		N/A	5. Order should be successfully confirmed and order status should be updated to Confirmed.
6. Login with customer user and verify the order status.		N/A	6. Order status should be displayed as Confirmed.

Table 5.13 : Order Cancellation Test Case

Test Case ID :	T013	**Test Description :**	Verify the user can Confirm Pending Order (Admin, Caretaker)
Prerequisites :	1. User should have logged into the application. 2. Customer should have placed an order.	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to Orders screen by clicking 'Food Request'		N/A	1. The Order List page should be displayed.

2. Click on the Cancel Order button .	N/A	2. Confirmation modal should be displayed with “Are you sure” text .
3. Click on cancel button from the modal	N/A	3. Modal should be closed and order should remain in Pending state
4. Click on the Cancel Order button	N/A	4. Confirmation modal should be displayed with “Are you sure” text
5. Click on Ok button	N/A	5. Order should be successfully cancelled and order status should be updated to Cancelled.
6. Login with customer user and verify the order status.	N/A	6. Order status should be displayed as Cancelled.

Table 5.14 : Acknowledge Complaint Test Case

Test Case ID :	T014	**Test Description :**	Verify the admin can acknowledge complaints logged by customer
Prerequisites :	1. Admin should have logged into the application. 2. Customer should have logged in a complaint.	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to Customer Complaints screen by clicking 'Complaints'		N/A	1. The Customer Complaints page should be displayed.
2. Verify the complaint status logged by the customer.		N/A	2. Status should be displayed as 'NOT Read'.
3. Click on Read button		N/A	3. Status of the complaint should be updated as 'Read' as displayed in Figure 5.12.
4. Login with customer user and verify the complaint status.		N/A	4. Complaint status should be displayed as 'Read'.

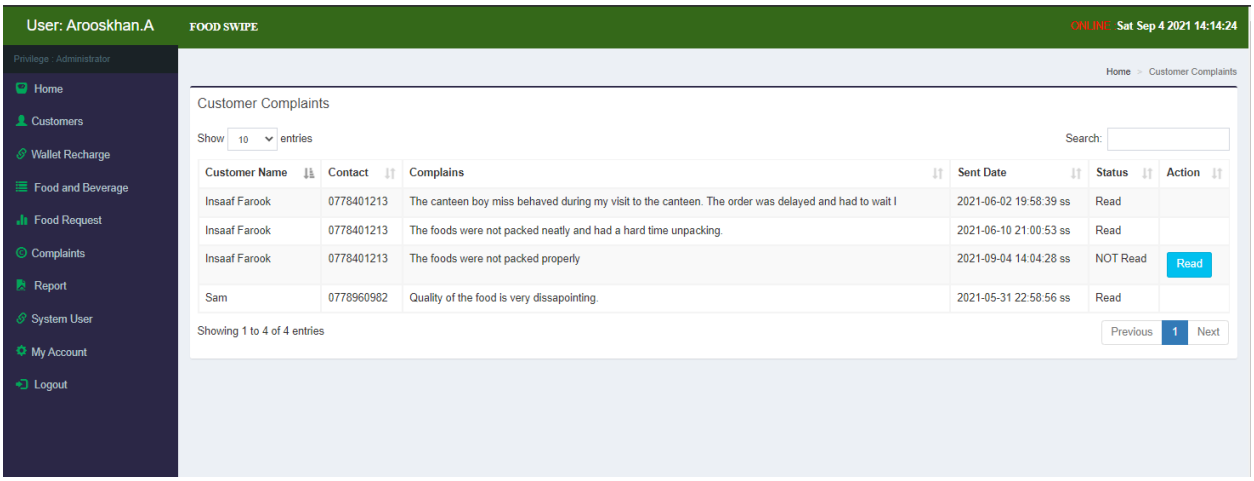


Figure 5.12 : Customer Complaints

Table 5.15 : Add New System User Test Case

Test Case ID :	T015	**Test Description :**	Verify the admin can Add new users to the application
Prerequisites :	1. Admin should have logged into the application.	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to Add New User screen by clicking 'System Users'		N/A	1. The Add New User page should be displayed.
2. Select User Type and enter values for Name, Username and password		N/A	2. Values should be successfully added.
3. Click on Save button		N/A	3. New User should be created and displayed in Users List as illustrated in Figure 5.13.

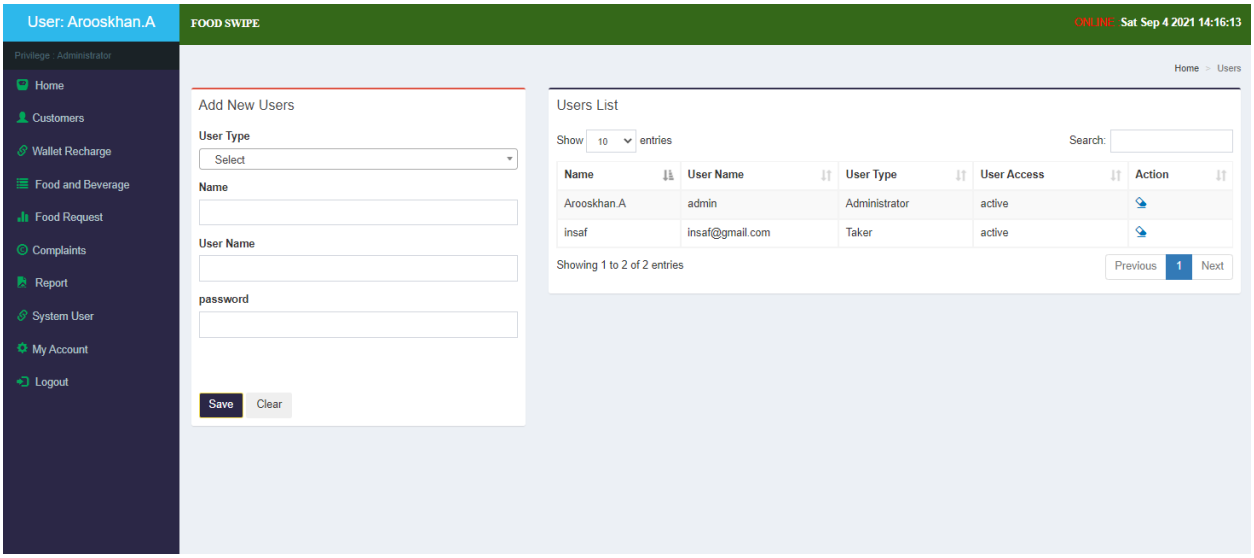


Figure 5.13 : System Users List

Table 5.16 : Food Stock Report Test Case

Test Case ID :	T016	**Test Description :**	Verify the admin can view Food Stock Report
Prerequisites :	1. Admin should have logged into the application.	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to Food Stock report screen by clicking 'Report' → 'Food Stock Report'		N/A	1. The Food Stock report page should be displayed.
2. Click on Print button		N/A	2. Browser print option should be displayed.

Table 5.17 : Customer Report Test Case

Test Case ID :	T017	**Test Description :**	Verify the admin can view Customer Report
Prerequisites :	1. Admin should have logged into the application.	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to Customer report screen by clicking 'Report' → 'Customer Report'		N/A	1. The Customer Report page should be displayed.
2. Click on Print button		N/A	2. Browser print option should be displayed.

Table 5.18 : Food Order Report Test Case

Test Case ID :	T018	**Test Description :**	Verify the admin can view Food Order Report
Prerequisites :	1. Admin should have logged into the application.	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to Order report screen by clicking 'Report' → 'Food Order Report'		N/A	1. The Food Order Report page should be displayed.

2. Select date range	From Date: Today To Date : Previous date	2. User should not be allowed to select the To date
3. Select date range	From Date: Past Date To Date : Past Date + 2	3. User should be allowed to select the date range, And orders placed during that period should be listed.
4. Select date range	From Date: Future Date To Date : Future Date + 2	4. User should be allowed to select the date range, And orders placed during that period should be listed.
5. Click on Print button	N/A	5. Browser print option should be allowed as displayed in Figure 5.14.

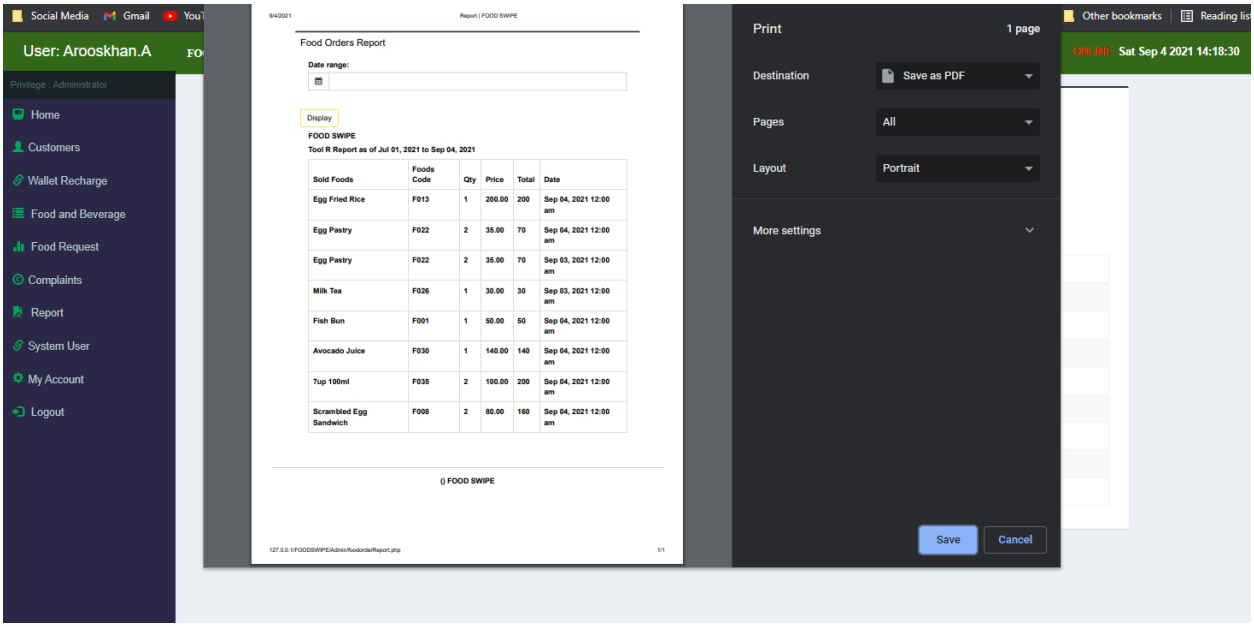


Figure 5.14 : Food Order Print report

Table 5.19 : Wallet Recharge Test Case

Test Case ID :	T019	**Test Description :**	Verify the caretaker can recharge the customer's wallet.
Prerequisites :	1. Caretaker should have logged into the application.	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to the Wallet screen by clicking 'Wallet Recharge'.		N/A	1. The Wallet page should be displayed.
2. Select Customer from the dropdown		Sam	2. All the active customers will be listed out.
3. Enter wallet amount		1000	3. Wallet amount should be added.
4. Click Save button		N/A	4. Successfully updated message should be displayed.
5. Verify the Wallet Recharge table.		N/A	5. The Wallet Recharge table should have the new record added as displayed in Figure 5.15.

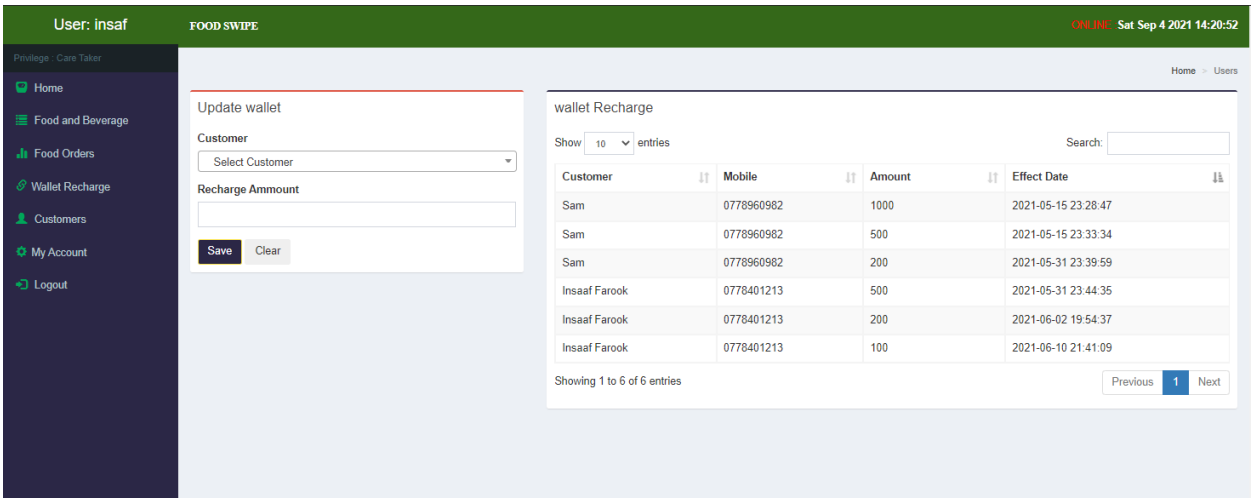


Figure 5.15 : Waller Recharges List

Table 5.20 : Update Customer Test Case

Test Case ID :	T020	**Test Description :**	Verify the caretaker can update customer details
Prerequisites :	1. Caretaker should have logged into the application.	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Navigate to the Customer List screen by clicking 'Customers' → 'Customer List'.		N/A	1. The Customer List page should be displayed.
2. Click on the edit icon relevant to the customer that needs to be updated.		N/A	2. Update Customer Details modal should be displayed.
3. Enter values for the fields		N/A	3. Should be able to update the values.
4. Click Save button		N/A	4. Customer' successfully updated' message should be displayed.

5.3.2 Non-Functional Test Cases

Non-functional tests are equally crucial as functional tests that affect the client's satisfaction. These tests are designed to test the application's readiness as per non-functional parameters, which are never addressed in functional testing. Non-functional tests are designed focusing on parameters such as security, usability, Efficiency.

5.3.2.1 Security Testing

The purpose of the security test is to identify the loopholes and weaknesses in the application by uncovering vulnerabilities, threats, and risks in the application, which might result in a loss of information, revenue and reputation.

Table 5.21 : Password Encryption Test Case

Test Case ID :	T021	**Test Description :**	Verify whether the passwords are stored encrypted in the database
Prerequisites :	N/A	**Test Status :**	PASSED
Test Step		**Test Data**	**Expected Results**
1. Register as a new customer and verify the password field in the database. (T001)		N/A	1. Passwords should be stored encrypted in the customer table as displayed in Figure 5.16.
2. Add a new system user and verify the password field in the database. (T015)		N/A	2. Passwords should be stored encrypted in the customer table as displayed in Figure 5.16.
3. Customer update password from the My Account page		N/A	3. Passwords should be stored encrypted in the customer table as displayed in Figure 5.16.
4. Admin/Caretaker update password from the My Account page.		N/A	4. Passwords should be stored encrypted in the customer table as displayed in Figure 5.16.

cid	Name	address	contact	email	password	walletbalacne	status
43	Sam	165/E Nawala Rd, Rajagiriya	0778960982	sam@gmail.com	8cdd21051a8dd11a0e3dc8300f36d31d	1580	Active
44	Insaaf Farook	185/E/2 Manikamulla Road, Maradana	0778401213	insaaf.mohamed10@gmail.com	2f3370e934d41dda50f4befc7d3445e3	200	Active
45	Tom Samuel	138, New Moor Street	0778358224	tom_s@gmail.com	5caf72868c94f184650f43413092e82c	0	Active

Figure 5.16 : Customer Table records

5.3.2.2 Usability Testing

Usability testing will be carried out to measure how easy and user-friendly the application is. A small set of targeted users are provided access to the application to identify usability defects. Apart from the tests that are mentioned in the above chapter, there will be ad hoc and exploratory tests will be executed to check how flexible the application is to handle controls and the ability to meet its objectives. Below are some ad hoc test scenarios that will be executed.

Table 5.22 : Ad Hoc Test Cases

Test ID	Description	Expected	Test Status
T100	Verify customer can view the order placed and print	Order should be able to be viewed in the orders page. Should be able to print orders.	PASSED
T101	Verify customers can filter food type by selecting from the dropdown.	Only the food with selected food type should be displayed.	PASSED
T102	Verify use can discard the food review changes back navigating back to Food Menu	Users should be able to discard changes.	PASSED
T103	Verify Customer can filter number entries to be shown in My Ordered Food page	Customers should be able to filter the number of entries.	PASSED
T104	Verify Customer can search for the order by simply typing order date in the search field.	Customers should be able to search the order by entering the order date in the search field.	PASSED
T105	Verify Customer can filter number entries to be shown in Wallet Recharges page	Customers should be able to filter the number of entries.	PASSED
T106	Verify Customer can search for the wallet recharge record by simply typing the recharged date in the search field.	Customers should be able to search the recharge record by entering the recharge date in the search field.	PASSED
T107	Verify Customer can filter number entries to be shown in complaints page	Customers should be able to filter the number of entries.	PASSED
T108	Verify column filters can be applied in tables.	Should\be able to filter columns in the tables.	PASSED

5.3.2.3 Performance Testing

Performance test allows evaluating the device's speed, responsiveness, and stability, network, application or the device under the workload. According to the current implementation of the application, it is required to examine the front end performance. Google developer tool will be used as the tool for the front end performance test. This allows us to identify the loading, scripting and rendering times of the page. According to the standards, each page should render within 3 seconds. Below are some critical tests that are carried out in pages that contain more processing and renderings.

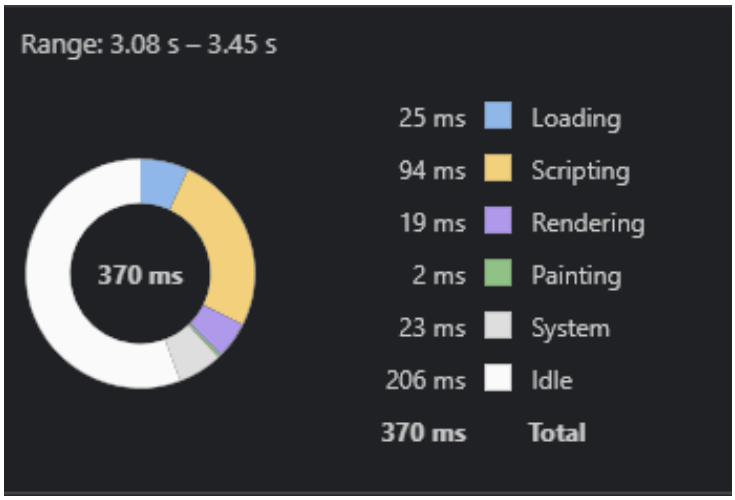


Figure 5.17 : Food Menu Performance Stats

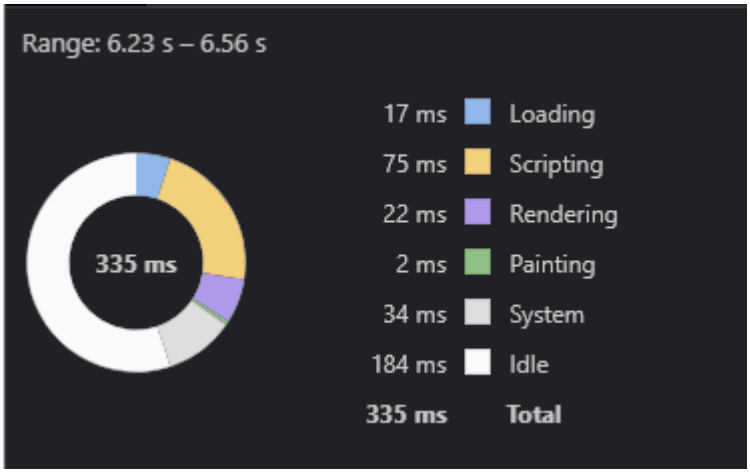


Figure 5.18 : Food View Performance Stats

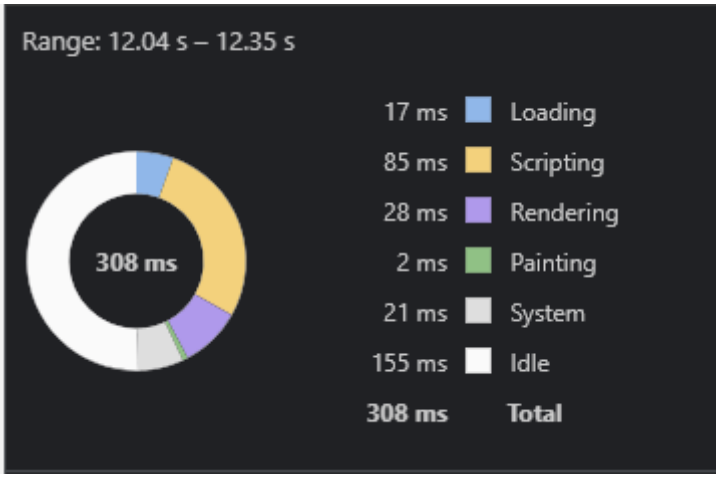


Figure 5.19 : Submit Order Performance Stats

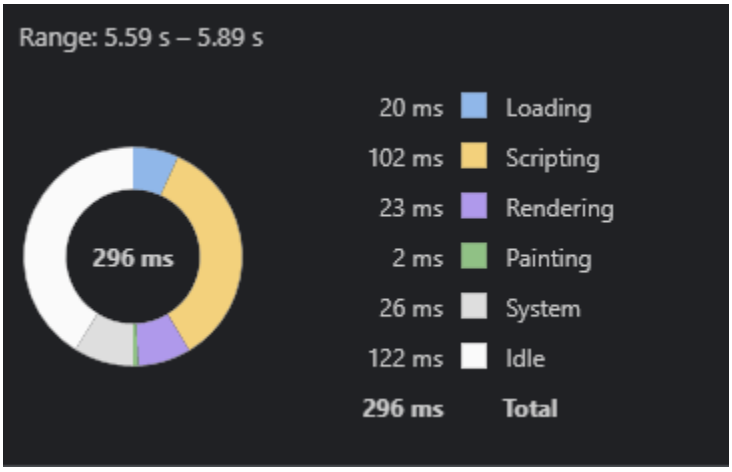


Figure 5.20 : View Complaints Performance Stats

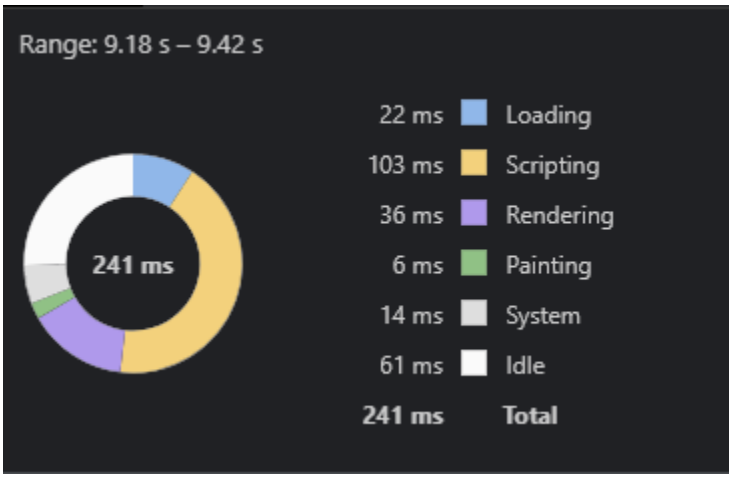


Figure 5.21 : View Order Performance Stats

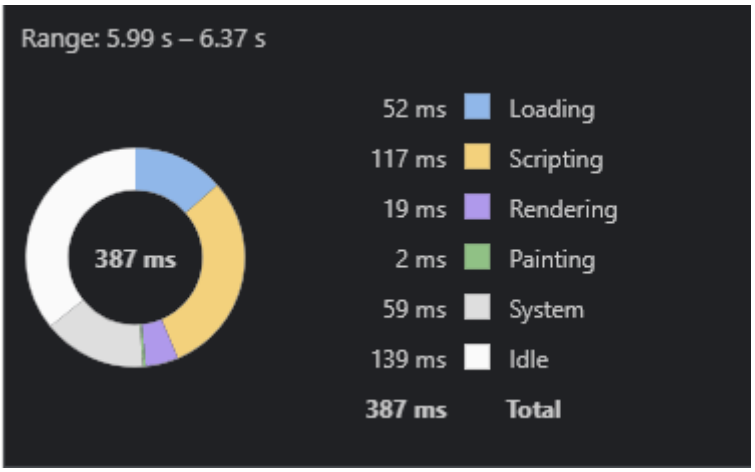


Figure 5.22 : Food Stock Report Performance Stats

5.4 User Evaluation

As the final stage of the testing phase, user evaluation was carried out. This focused on how well users can learn and use the application to perform the actions/ functionalities. It also refers to how users are satisfied with the application developed. This information can be gathered from the users with a variety of feedback gathering approaches. Questionnaires are the main instrument for collecting data in surveys, and that was carried out to gather the details mentioned above from the end-users. Both open-ended and closed-ended questions are included in the questionnaire targeting all the user roles associated with the application. Due to the current pandemic situation it was not practical to conduct feedback gathering from a vast audience. Approximately 25 users are given access to the application in order to derive feedbacks. The questionnaire included in Appendix B. Evaluation is conducted based on four categories: load time, user-friendliness, and functionality. Below are the results that are identified through the user evaluation.

Load Time

More than half of the users mention that the website is loaded without an issue on time. Figure 5.7 illustrates the detailed results.

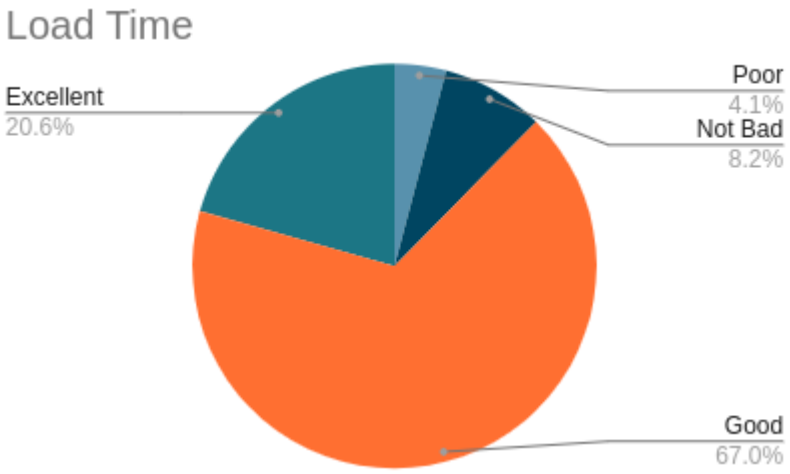


Figure 5.23 : Load Time Evaluation Results

User Friendliness

According to Figure 5.8 nearly 80% of the users are satisfied with the application's appearance, and they feel the application is more user friendly. It is observed that the application is clearly represented to the end-users. Observed a lower rating from the canteen caretakers who are above 45 years old.

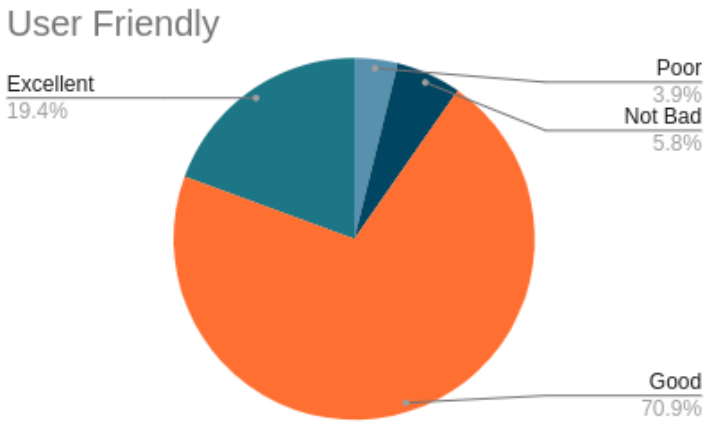


Figure 5.24 : User Friendly Evaluation Results

Functionality

Almost 90% of the users are happy with the functionalities built in the application, and they were able to get the required things done with less effort and time. Figure 5.9 illustrates that detail below.

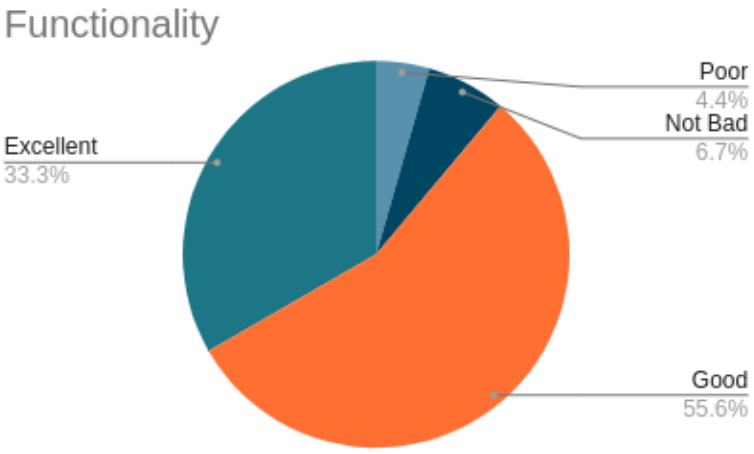


Figure 5.25 : Functionality Evaluation Results

Overall Rating

When considering the overall rating reserved for the application, it is evident that the objectives of the application are met, and more enhancements can be included in the application. Results represented in Figure 5.10 make it evident.

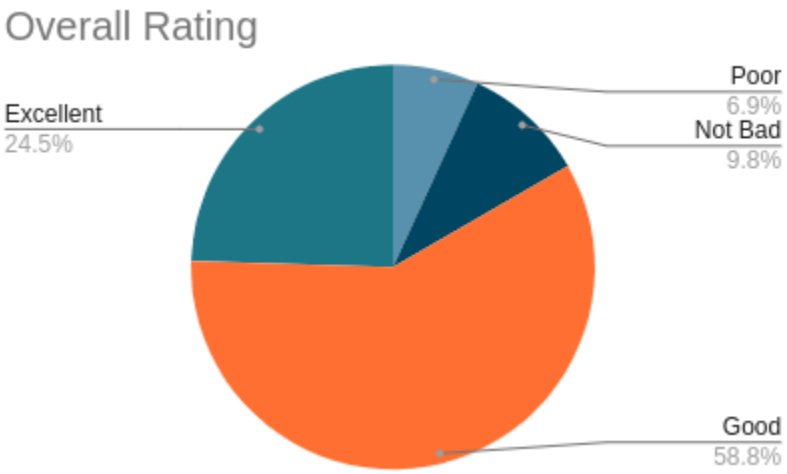


Figure 5.26 : Overall Rating Results

6. Conclusion

Canteen that operates in places with colossal crowding suffer various issues when they carry out business in a manual way which is a drawback in the current era. This study powerfully demonstrates that the business can be taken to the next level by using current technology but further raises fundamental questions about how this can be made possible and how effective this would be. In the current context, customers visit the canteen and get the orders placed. Based on the crowd, the time may take longer than expected. The time spent on queue, time taken to prepare the order and time processing the bill payments are taken can be overtaken by implementing the web application discussed in the above chapters.

As discussed in the background chapter, there are canteen management systems that are available in the market. Some focus only on the internal business of an organization's canteen, while the others have limited set of functionalities when compared to the current study. The developed application can handle moneyless transactions without integrating a payment gateway with higher security concerns and increasing the development cost.

Many challenges were faced during the development of the application. The canteen admins and the caretakers had very little technology knowledge hence it was difficult to convey the right idea to them to gather information. Some of the staff were not fit to handle computerized systems and had to groom them on this. The other most significant challenge was COVID 19 pandemic. It was a considerable struggle contacting the users to gather information which is a considerable risk that might build the wrong product. Also, this caused difficulties in evaluating the application due to unavailability to reach a vast set of system users to get more accurate details.

Some exciting results derived from the user evaluation can be considered an enhancement for the application as future work. Some customers feel that it is better to use One True Pairing (OTP) for login and the notifications to be sent as Short Message Service (SMS) to the registered account. The addition of these two features to the application would improve the application quality.

References

[1] FreeProjectz. 2021. Company Canteen Management System. [online] Available at: https://www.freeprojectz.com/paid-projects/php-mysql/company-canteen-management-system (Accessed 10 November 2020).

[2] Ashik, S., Gopi, T. and Sanjudharan, M., 2018. ONLINE CANTEEN SYSTEM. [online] Available at: https://www.researchgate.net/publication/333201476_ONLINE_CANTEEN_SYSTEM (Accessed 20 November 2020).

[3] Shah, M., Shah, S., Shaikh, M. and Tiwari, K., 2018. Canteen Automation System. International Research Journal of Engineering and Technology (IRJET), [online] 05(01). Available at: https://www.irjet.net/archives/V5/i1/IRJET-V5I1320.pdf (Accessed 20November 2020).

[4] International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), 2020. MOBILE APPLICATION FOR CANTEEN AUTOMATION SYSTEM USING ANDROID. 09(03). (Accessed 20 November 2020).

7. Appendices

Appendix A: User Interface

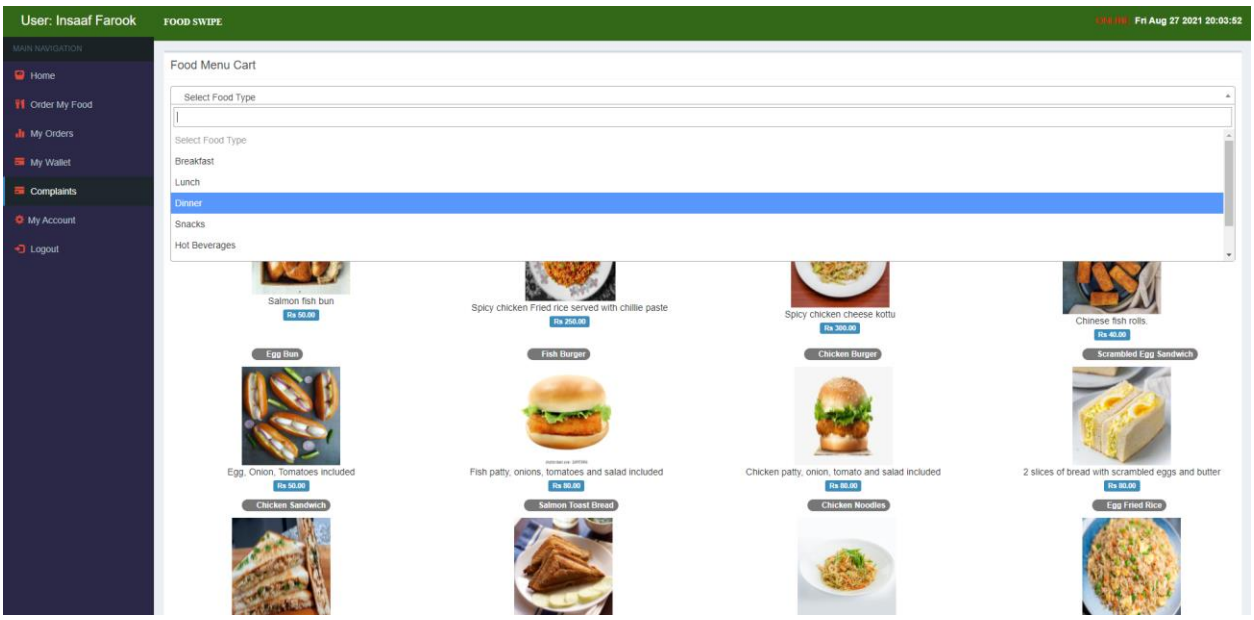


Figure 7.1 : Menu Category Filtration

Figure 7.1 displays the feature that allows user to filter foods by food type.

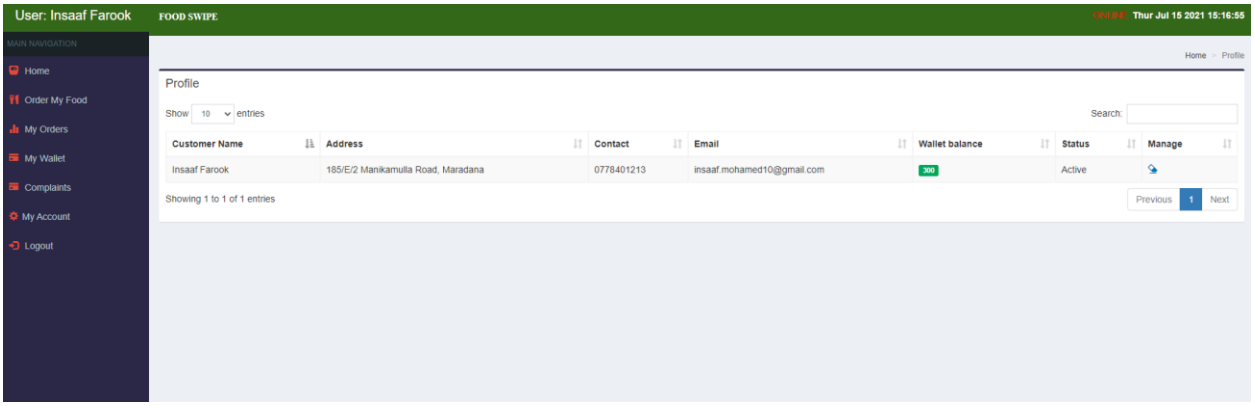


Figure 7.2 : Wallet Balance Screen

Figure 7.2 illustrates the screen that customers uses to view their current wallet balance along with the details.

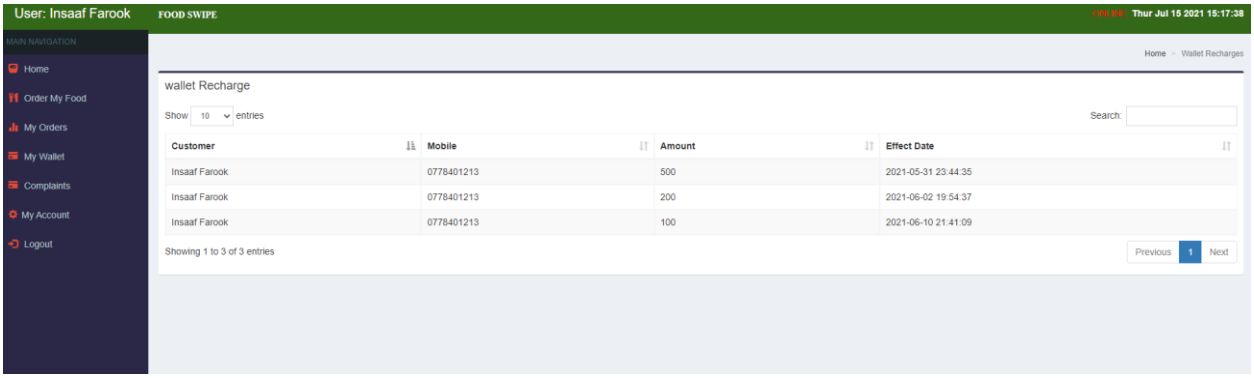


Figure 7.3 : Wallet Recharges Screen

Figure 7.3 illustrates the screen that shows the recharges done by the customer.

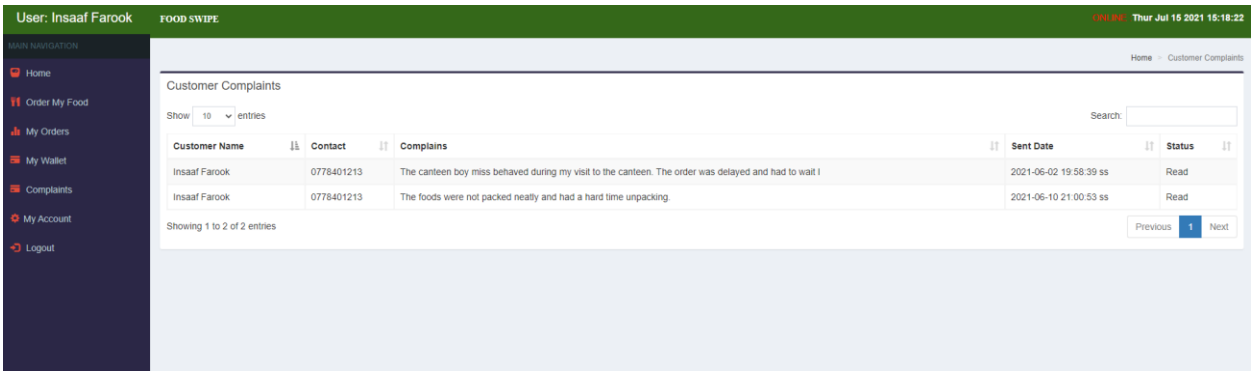


Figure 7.4 : Logged Complaint Screen

Figure 7.4 illustrates the screen that allow customers to see the complaints logged and the current status of it.

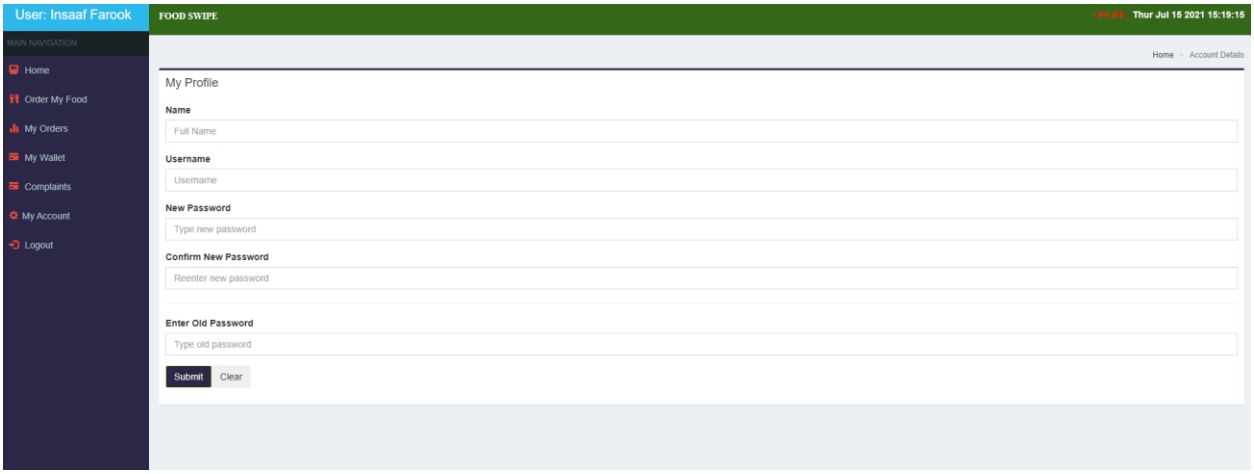


Figure 7.5 : Customer Profile Update Screen

Figure 7.5 illustrates the profile screen of the customer, where he can update his profile details and password.

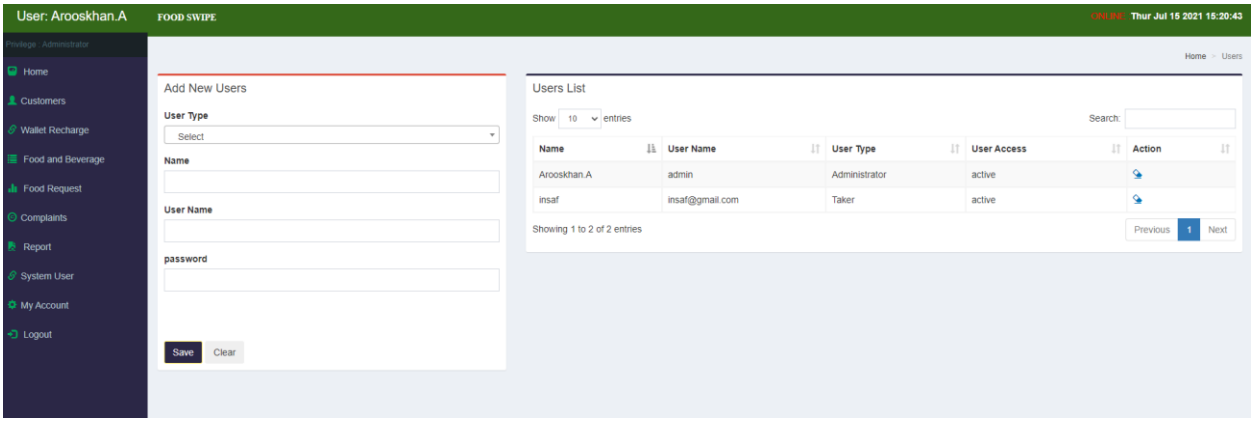


Figure 7.6 : Add User Roles Screen

Figure 7.6 illustrates the screen that the admin uses to add caretaker and admin user accounts to the system.

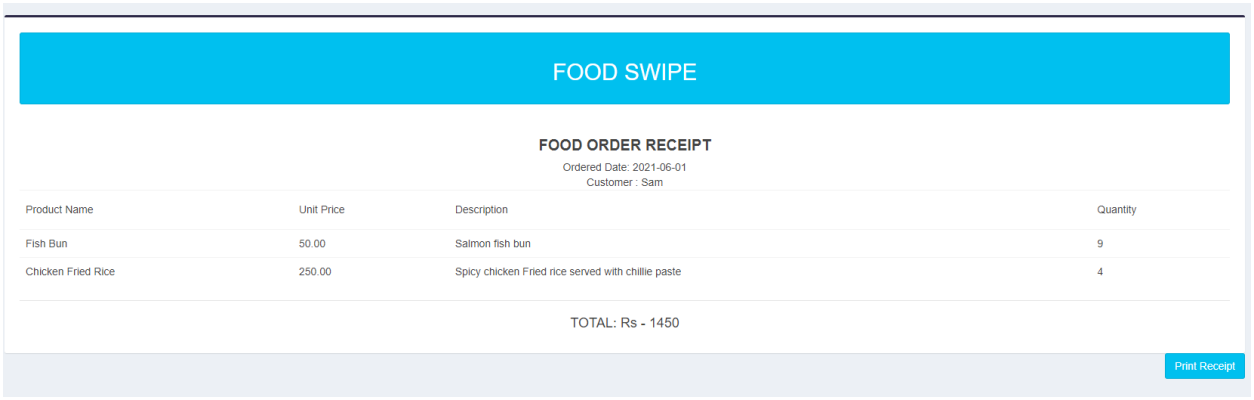


Figure 7.7 : Order Receipt Print Screen

Figure 7.7 illustrates the receipt that the customers can view once the orders are placed.

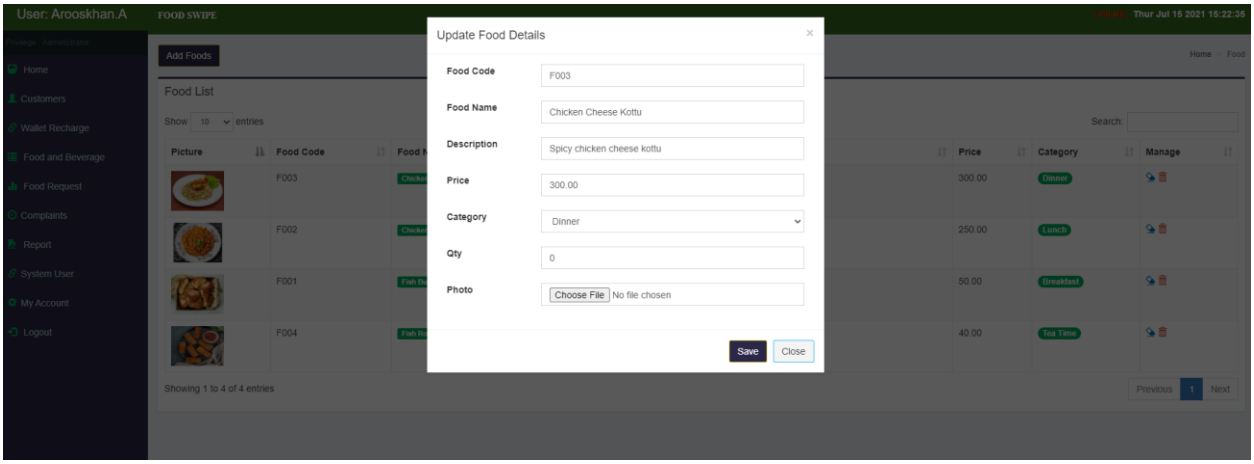


Figure 7.8 : Update Food Record Screen

Figure 7.8 illustrates the pop up modal used to update Food details.

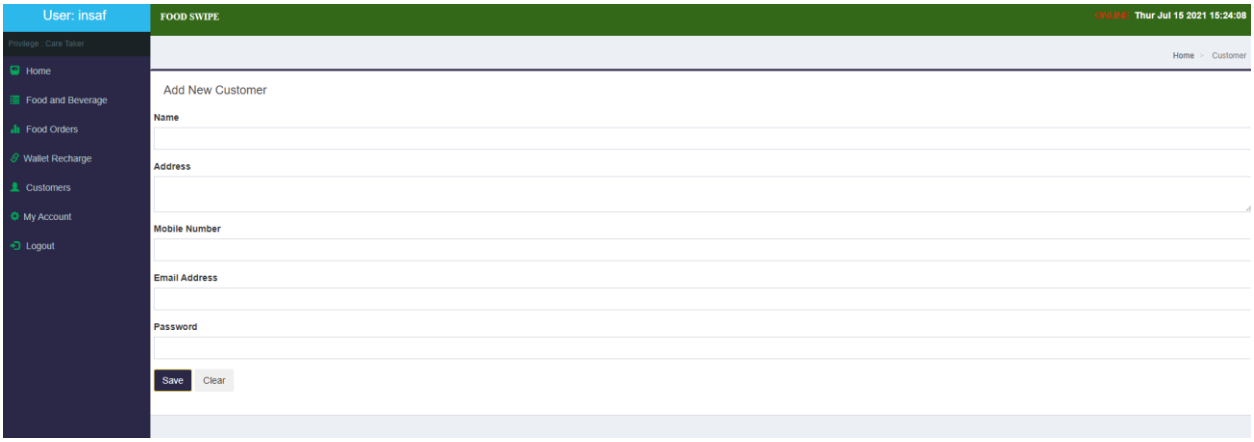


Figure 7.9 : Add New Customer Screen

Figure 7.9illustrates the screen used by administrator and caretaker to create customer accounts on behalf of customers.

Appendix B: Questionnaire

1. Please select the category that best describes your role:
   [] Customer [] Admin [] Caretaker

2. Does the application load time seem reasonable compared to your network speed?
   [] Poor [] Not Bad [] Good [] Excellent

3. How easy is the application to use?
   [] Poor [] Not Bad [] Good [] Excellent

4. The content of the application clearly represented?
   [] Poor [] Not Bad [] Good [] Excellent

5. The application navigation is easy to understand:
   [] Poor [] Not Bad [] Good [] Excellent

6. If a Customer user, what is your preferred payment/delivery method?
   [] Cash [] Wallet

7. If a Customer user, how useful is the wallet payment feature?
   [] Poor [] Not Bad [] Good [] Excellent

8. What do you like least/most about the application?
   ……
   ……

9. Which features of the application are most important to you?
   ……
   ……

10. What is the most important feature you think that should be added to the application?

……

……

11. Overall, how would you rate our site?

[] Poor [] Not Bad [] Good [] Excellent

****