Deakin University

Applied Software Engineering

OnTrack Submission

Project Delivery

Submitted By:
Akashdeep Akashdeep
s223040483
2023/10/10 19:34

Tutor:
Stephen Liu

October 10, 2023

Foodie Fare

SRS Document

Table of Contents

1. Introduction

1.1 Overview of the Software and its Purpose

"FoodieFare" is a web-based application designed to provide users with a contactless food menu experience, especially for market stalls. The application uses QR codes, enabling users to swiftly access menus, order, and get instant notifications about their food status.

1.2 Identification of the Target Audience

- Market Stall Owners
- Diners or Customers
- The Development Team

2. Purpose

2.1 Explanation of the Primary Objective of the Software

The primary objective of "FoodieFare" is to minimise physical contact and enhance customer experience through swift menu access and efficient order management for stall owners.

2.2 Clarification of the Intended Benefits for Users

The application aims to provide users with a seamless dining experience, reducing waiting times and ensuring hygienic practices. For stall owners, it ensures efficient order management and insights into popular menu items.

3. Audience

3.1 Description of the Target Users or Stakeholders

3.1.1 Market Stall Owners

Using FoodieFare, stall owners can:

- Update real-time menu availability.
- Manage and prioritise orders.
- Receive analytic data on menu preferences.
- Process digital payments.

3.1.2 Diners or Customers

The primary users of FoodieFare would include:

- Individuals visiting the market for dining.
- Tourists exploring local foods.
- Event attendees during special occasions or festivals.

4. Overall Description

4.1 Comprehensive Overview – Functionality and Purpose

FoodieFare allows users to scan QR codes to access menus and place orders. Stall owners receive order notifications, process them, and notify customers upon readiness. The application also integrates a review system and offers insights to stall owners based on user preferences.

4.2 How the Software is Intended to Operate

Upon scanning the QR code, users access the digital menu. They can select items, customize orders, and pay digitally. Stall owners view and manage these orders, processing them in real-time. Feedback mechanisms are also incorporated.

5. External Interfaces

5.1 Description of the External Interfaces

- Payment Gateways – Integration with platforms like Stripe or PayPal for secure payment processing
- Notification APIs – To send real-time alerts to users about their order status

6. System Features

6.1 FoodieFare's Key Features:

- QR Code Scanning – Instant access to web application
- Digital Payment Options – Multiple methods for swift payments
- Order Tracking – Real-time updates on order preparation and readiness
- User Profiles – For personalised recommendations and order history
- Review System – Users can rate and review stalls, aiding others in selection

7. Non-functional Requirements

- Performance – Quick response times, ensuring orders and menu access within seconds
- Security – Secure payment gateways and user data protection in line with relevant regulations
- Reliability – Real-time data representation without delays
- Usability – Intuitive UI/UX, ensuring easy navigation and efficient task completion

8. Other Requirements

- Scalability – Capable of handling multiple stalls and a growing number of users
- Maintenance – Regular software updates to incorporate new features and ensure smooth functionality
- Compliance – Adherence to data protection regulations and digital transaction norms

- Multi-device Functionality – Consistent performance across various devices and screen sizes

9. Team Composition and Responsibilities

9.1 Team Member 1 (User Registration Lead) - Akash

Role Overview:
Focused primarily on ensuring a seamless user registration and login experience for the "FoodieFare" application.

Responsibilities:

- Overseeing the design and functionality of the Sign-up and Login pages.
- Incorporating a user-centric approach while curating the feedback form.
- Ensuring data security and privacy, especially during the registration and login processes.

Pages:

1. Sign-up Page: Where new users can create their account.
2. Login Page: For returning users to access their account.
3. Feedback Form: To gather valuable input and address user queries or concerns.

Primary Tasks:

- **Design and Development:**
  - Sign-up page
  - Login page
  - Feedback page

- **Database Management:**
  - Set up and manage a database to securely store user registration details.
  - Implement a separate database or section for collating feedback and user queries.

- **Functionality & Security**
  - Ensure efficient functioning of the Sign-up and Login pages.
  - Implement login validation mechanisms to prevent unauthorized access.
  - Integrate password encryption methods using MongoDB

9.2 Team Member 2 (User Experience & Content Specialist) - Heibrance

Role Overview:
Responsible for curating a user-centric experience by overseeing the User Account portal, and emphasising intuitive navigation and display on the homepage, especially for Restaurant Listings and Menus.

Responsibilities:

- Establishing a clear and cohesive Business Logic, ensuring users understand the functionality of the platform.
- Overseeing the User Account portal's design, functionality, and content.
- Enhancing the homepage's interface to intuitively display restaurants, market stalls, and their respective menus.

Pages:

- Business Logic – a page to show how the platform operates.
- User Account Portal – for users to manage their accounts, view purchase history, and edit details.
- Homepage – the landing page which showcases restaurant listings, menus, and promotions and allows efficient stall search functionalities.

Primary Tasks:

- **Business Logic Creation**
  - Curate content and design for the Business Logic page to ensure users grasp the platform's operational framework.
- **User Account Portal**
  - Design the User Account portal
  - Develop the portal, ensuring smooth functionality.
  - Set up a database specifically for the User Account portal to store user-specific data.
  - Incorporate "Edit Details" features to update user information.
  - Implement a "Ticket" section showcasing users' purchase history.
- **Homepage Design & Functionality**
  - Develop rendering for the Homepage
  - Design UI for listing stalls, integrating options to add, edit, or delete listings.
  - Design specific homepage elements like a promotional carousel, detail view of market stalls, and user ratings.
  - Integrate efficient search options, allowing users to locate specific stalls or restaurants easily.

9.3 Team Member 3 (Stall Management Lead) - Preethi

Role Overview:
Oversees the design, functionality, and content of Market Stall Profiles and ensure that the “market stalls” are presented in the same format and easy for people to understand the offerings and make decisions.

Responsibilities:

- Design and layout of Market Stall profiles
- Design presentations of products, covering aspects like price, quantity, and ratings.
- Implement the functionalities associated with the stalls such as search functions, promotions, and filtering.

Pages:

- Stall Profiles and Products – where users will view individual market stall offerings, read about them, and see their product listings.

Primary Tasks:

- **Market Stall Profiles Design:**
  - UI Design for Market Stall profiles
  - Stall details such as Name, Stall number, and general information including ratings.
- **Products Presentation:**
  - UI Design for Market Stall products
  - Each product’s price and availability
- **Search and Filtering Functions:**
  - Develop search function specifically for browsing through the menu.
  - Allow users to filter and find specific types of dishes or categories.
- **Promotions and Transactions:**
  - Design and integrate a promo code function
  - Develop "Add to Cart" function,
- **Backend Management and Testing:**
  - Establish and manage a database dedicated to storing details of the stalls and their products, ensuring data security and efficiency.
  - Conduct thorough testing of all functionalities related to the stall profiles and rectify any glitches or errors found during the testing phase.

9.4 Team Member 4 (Ordering Management) – Sandriya

Role Overview:
Ensure a seamless and intuitive ordering process for FoodieFare users. She will manage the end-to-end experience of selecting items, modifying orders, applying promotions, and finalising payments.

Responsibilities:

- Oversee the creation and optimisation of the checkout process, ensuring it's user-friendly, efficient, and secure.
- Coordinate between the user interface and the backend functionalities to ensure data consistency and smooth transaction flow.

Pages:

- Checkout Cart – where users can view their selected items, edit, apply promotional codes, and proceed to payment.

Primary Tasks:

- **Design UI for Checkout Cart**
- **Order Modification:**
  - Allow users to edit items in their cart – adjusting quantities or removing items
  - Allow user to redirect if they wish to return to the stall listings to add or remove items
- **Promotions and Pricing:**
  - Integrate the Promo Code function into the checkout process
  - Ensure the price section updated to reflect the cost after applying the promo code
- **Backend Order Management:**
  - Set up database for the ordering system, capturing order details such as items ordered, quantities, applied discounts, and final prices.
  - Coordinate with other database, such as the user registration and stall management databases

Team Member 5 (Payment Integration Lead) – Sarfaraz

Role Overview:
Focuses on streamlining the payment process for FoodieFare users and ensures that the integration of multiple payment methods is seamless and that every transaction is secure and efficient.

Responsibilities:

- Implement and manage various payment methods
- Oversee data consistency between the payment system and the backend, ensuring accurate order tracking and receipt generation.

Pages:

- Payment Page – allow users finalize their payments, review their bill summary, and choose their preferred payment method.

Primary Tasks:

- **Bill Summary:**
  - A summary section that provides users with a detailed breakdown of their order, including itemized costs, applied promotions, and the final payable amount.
- **Payment Type Options:**
  - Display a set of payment methods that users can choose from.
- **Payment Implementations:**
  - Incorporate a range of payment solutions:
    - Cash Option: Develop a "Pay at the Counter" choice for users
    - PayPal Integration
    - Card Payment: supports both credit and debit cards
- **Backend Integration:**
  - Collaborate with the database team
  - Ensure data is recorded accurately and is consistent with order data from other systems.
- **Order Confirmations and Receipts:**
  - Develop an automated system to generate a confirmation of order upon successful payment, informing users that their transaction was successful.
  - Produce an electronic order receipt that provides users with a detailed record of their purchase. Receipts are stored in the user's account history and be accessible for future reference.
  - Receipts will serve as a proof of purchase for food pick-up.

10. Sprint Planning

10.1 Sprint 1: Development Phase

Overview:
The primary objective of this sprint is to lay down the foundational components of the FoodieFare platform. Each team member will focus on their respective areas of expertise to develop core features of the application.

1. Team Member 1 (User Registration Lead -Akashdeep):

- Develop the "Sign up" and "Log in" pages.
- Design and initiate the user details database.
- Implement the feedback form and the corresponding feedback database.

2. Team Member 2 (User Account and Homepage Design Lead – Heibrance Lo):

- Create the "Business logic" page.
- Develop the "User Account" portal with features like edit details and ticket history.
- Design and implement the "Homepage", including features like the promotional carousel, ratings, and detailed market stall listings.

3. Team Member 3 (Stall Management Lead - Preethi):

- Develop UI for the "Stall Profiles and Products".
- Incorporate details like stall name, number, and ratings.
- List products with pricing and quantity details.

4. Team Member 4 (Ordering Management - Sandriya):

- Develop the "Checkout Cart" page.
- Incorporate functionality to edit items, apply promo codes, and see the updated price section.

5. Team Member 5 (Payment Integration Lead - Sarfaraz):

- Design the "Payment" page.
- Implement bill summary and multiple payment options.
- Begin database integration for payments.

10.2 Sprint 2: Integration Phase

Overview:
The goal of this sprint is to seamlessly integrate the components developed in the first sprint. Team members will collaborate to ensure a unified and smooth user experience across the application.

1. **Team Members 1 & 2 (Akash & Heibrance):**
   - Collaborate on system-wide testing to identify and fix any bugs or issues.
   - Work on refining user experience based on preliminary feedback and testing outcomes.
2. **Team Member 3 (Stall Management Lead - Preethi):**

- Implement and test the "Add to cart" functionality.
- Develop and test the search function for menus and the categorization of the menu for filtering.

3. **Team Member 4 (Ordering Management - Sandriya):**
   - Implement and test the redirection to restaurants for item addition or removal.
   - Develop the connection to the payment page from the cart.

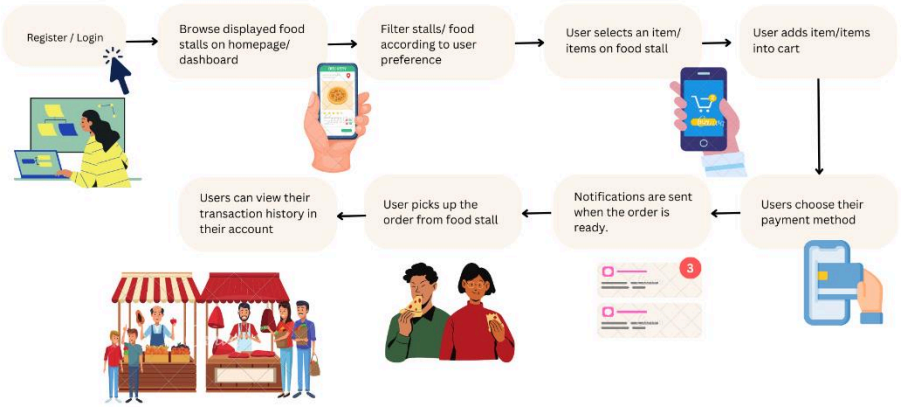
4. **Team Member 5 (Payment Integration Lead - Sarfaraz):**
   - Integrate different types of payment methods, including "Cash (Pay at the counter)", "PayPal", and "Card".
   - Generate and test the confirmation of orders and the order receipt functionalities.

5. Integration of front-end and back-end components will be assisted by all team members and Help each other in resolve any potential conflicts or integration bugs.

By the end of Sprint 2, the FoodieFare application should be in a mature state, with core features implemented and integrated. The platform should be ready for further testing, user feedback collection, and eventual deployment.

11. Application Diagram

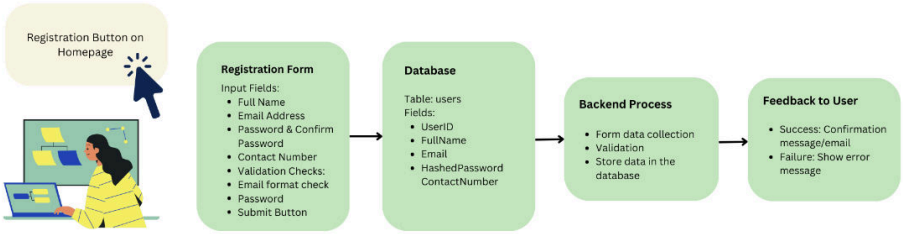
Foodie Fare Application Flow



12. User Stories

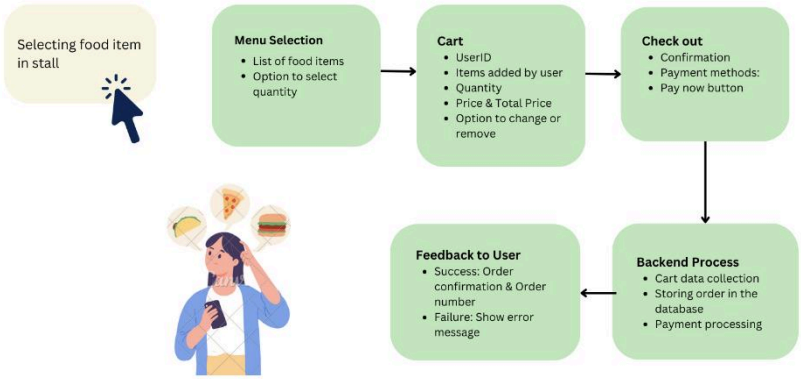
User Registration

As a new user, I want to easily find the registration page so that I can create an account and use FoodieFare



User Ordering Process

As a user, I want to add selected food items to my cart so that I can proceed to purchase them.

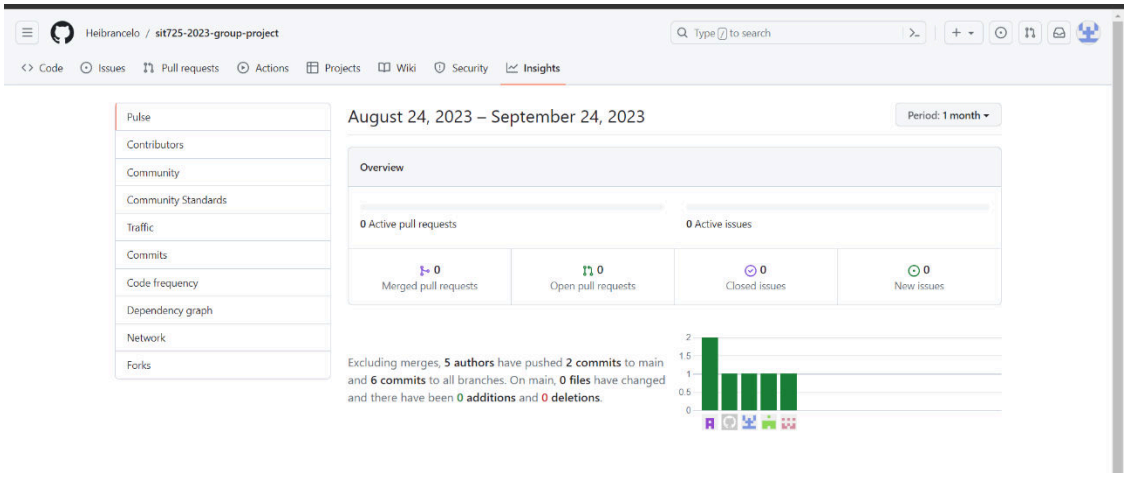


13. UI Layouts (Desktop & Mobile)

Figma link: https://www.figma.com/file/3hO5apinXBqN5LZ1fp4aAH/Foodie-Fare-Wireframe?type=design&node-id=1%3A31&mode=design&t=pRoRRmnbRI3lf7Pl-1

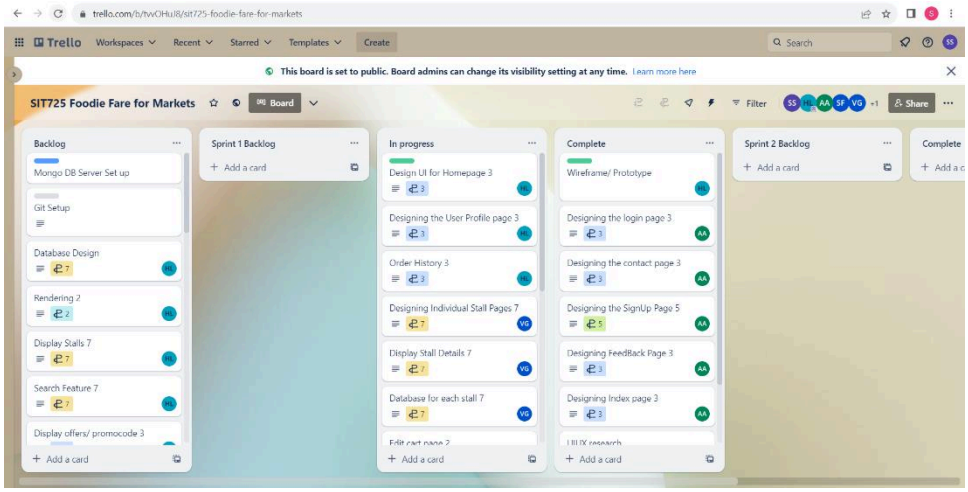
14. Appendix 1: Sprint 1 Review

14.1 Screenshot - Git contributor



14.2 Screenshot - Trello Board

Link: https://trello.com/b/tvvOHuJ8/sit725-foodie-fare-for-markets

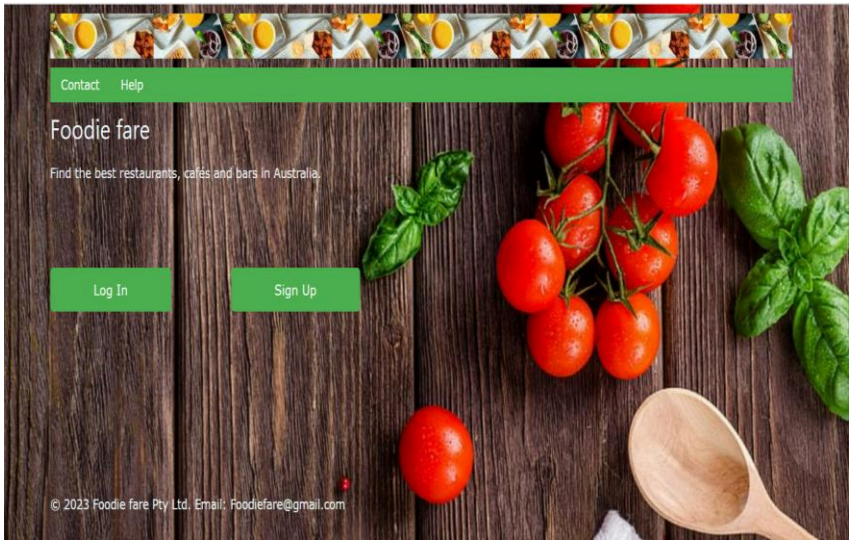


14.3 Table of tasks completed

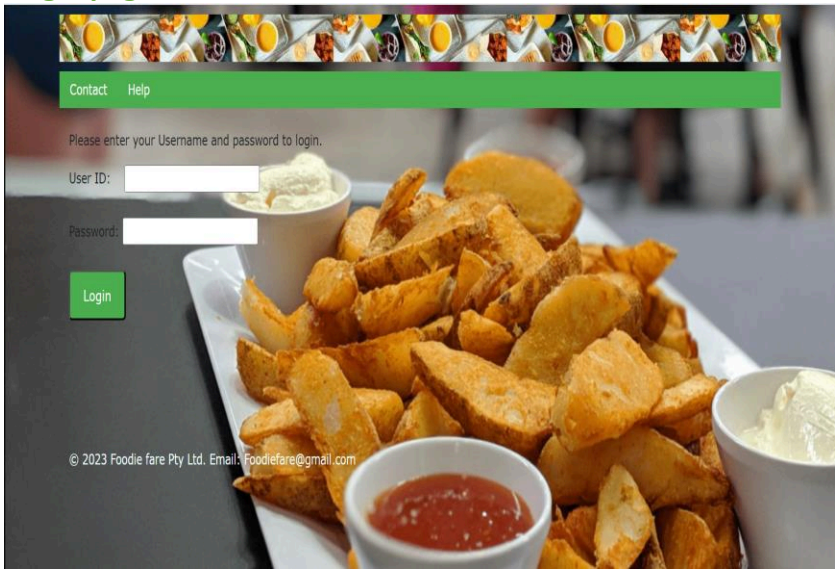
Members	Task completed (Sprint 1)
AkashDeep	• Developed Index page • Developed Login page • Developed signup page • Developed contact page
Heibrance	• Wireframe in Figma • Application Flow • User stories • Developing Dashboard page
Preethi	• Developed food menu page • Retrieve food items from database and display them on page
Sandriya	• Developed for Cart Page • Developed for Bill summary
Sarfaraz	• Developed Payment Page • Created UI components for Credit/Debit, PayPal and Apple Pay

14.4 Screenshot of the current application

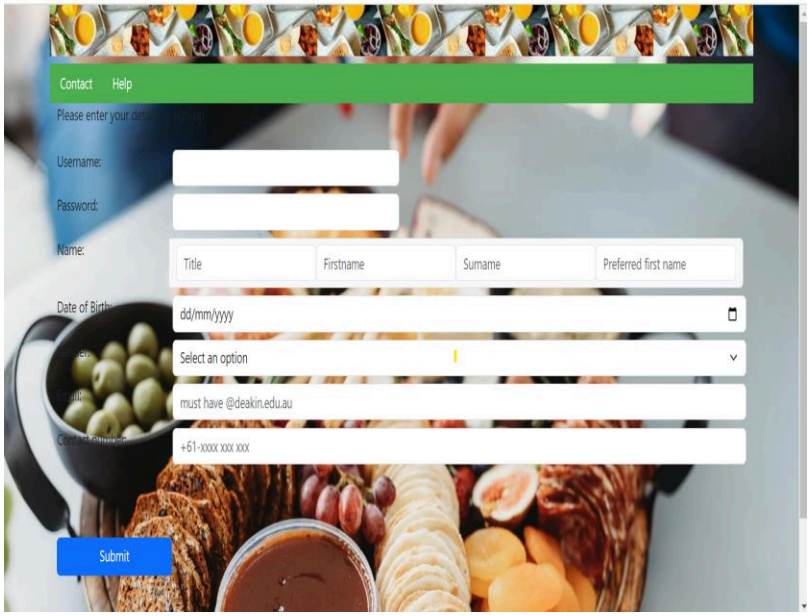
Index page



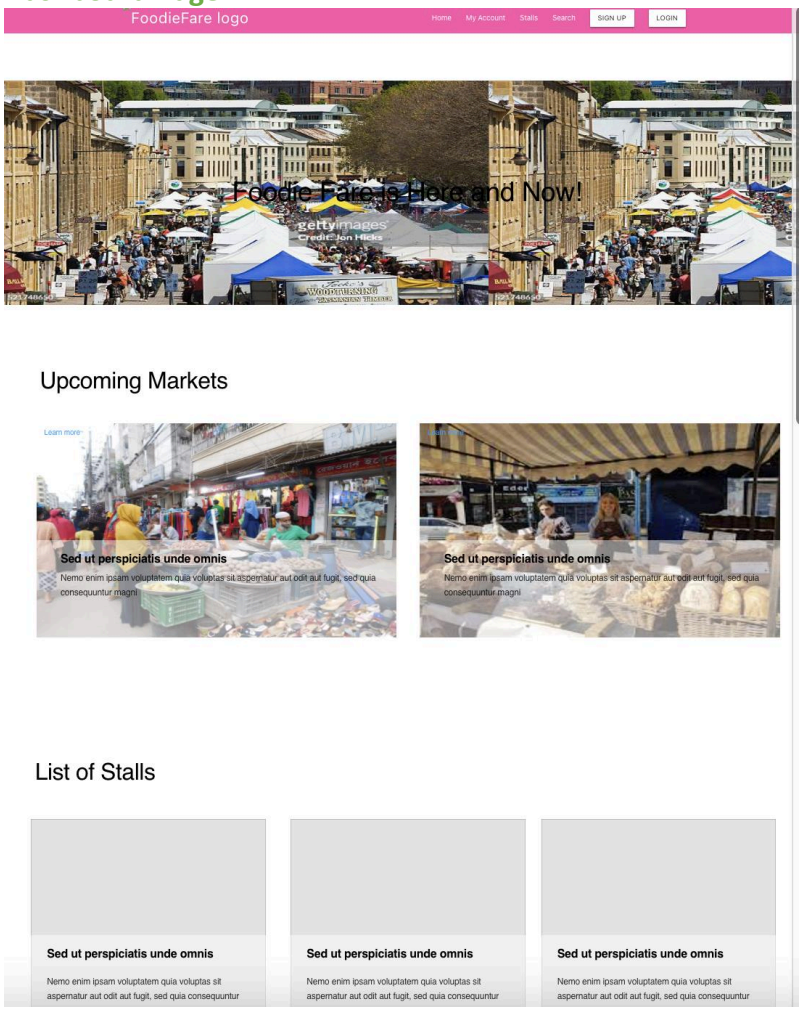
Login page



Signup Page

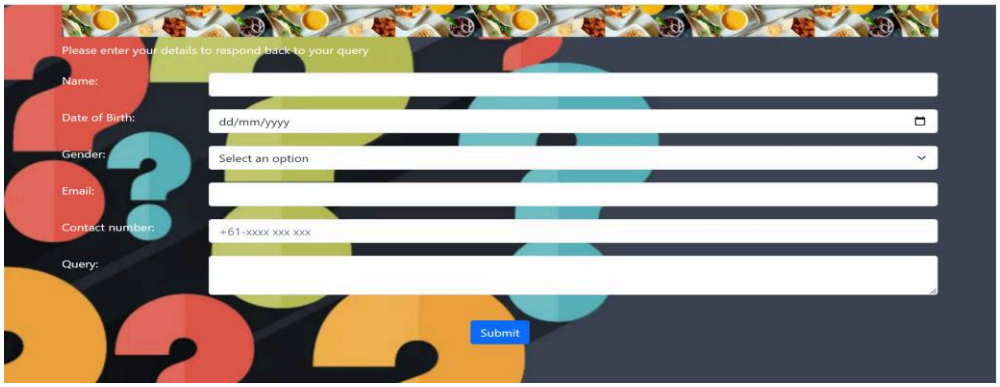


Dashboard Page

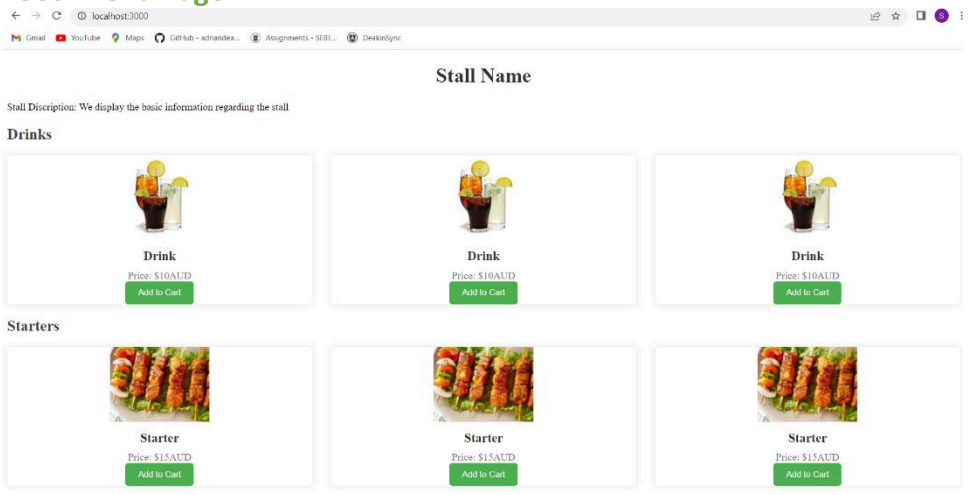


Contact Page

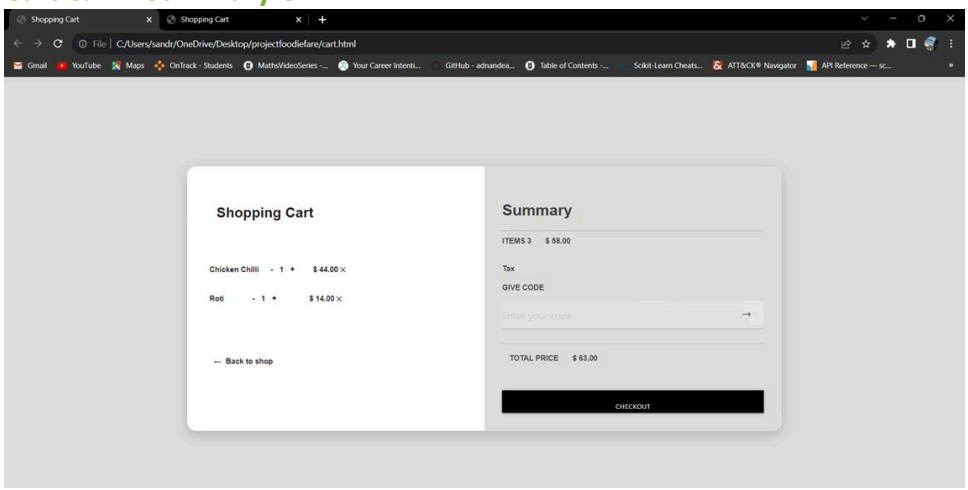
Query Page



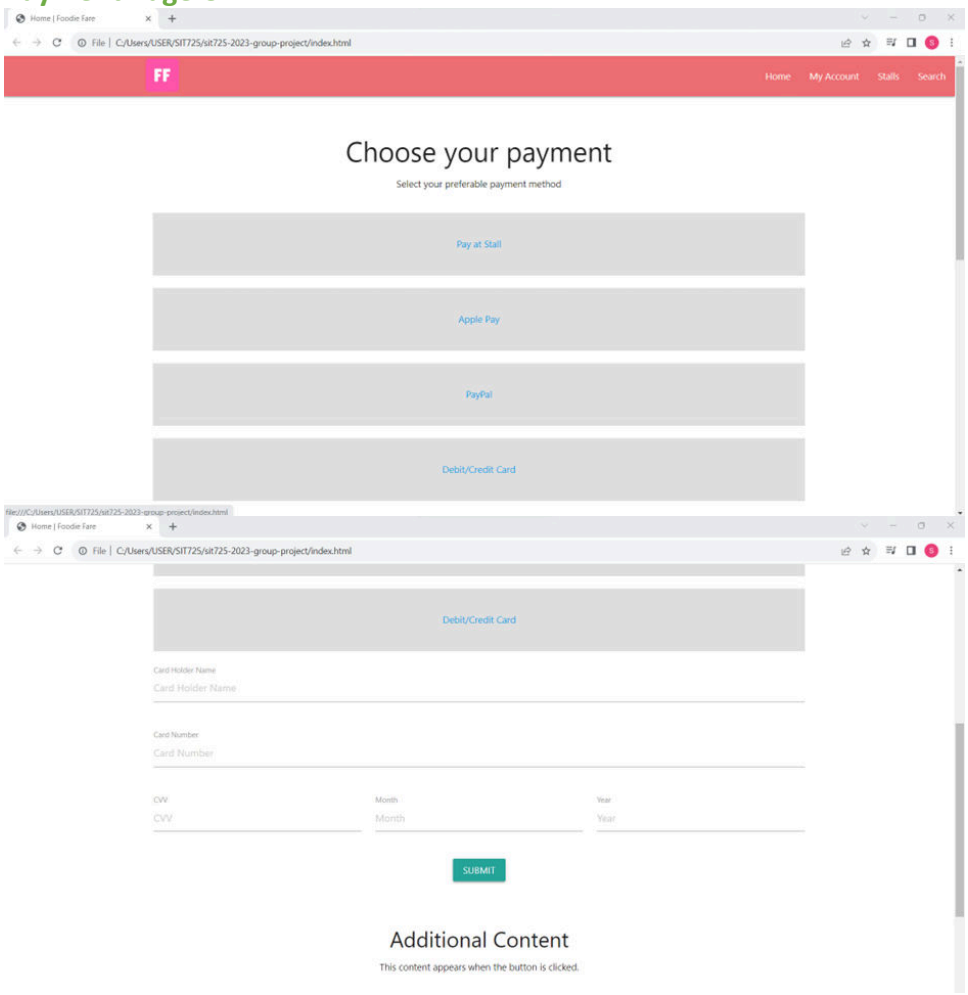
Food Menu Page

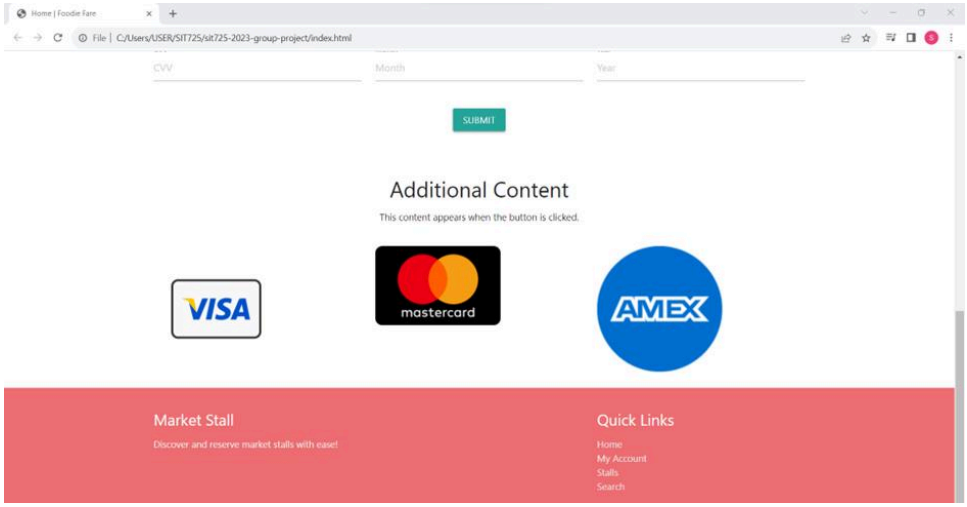


Cart & Bill Summary UI



Payment Page UI




Home | Foodie Fare
File | C:/Users/USER/SIT725/sit725-2023-group-project/index.html
CVV
Month
Year
SUBMIT
Additional Content
This content appears when the button is clicked.
VISA
mastercard
AMEX
Market Stall
Discover and reserve market stalls with ease!
Quick Links
Home
My Account
Stalls
Search

14.5 100 works Reflection (All)

Wireframe, Dashboard (Heibrance)

Looking back on Sprint 1, I see we should have started with a unified base code, in order to have a smooth integration later across different branch on github. Even though we have already started doing our own parts, we would still have to build a foundation for alignment.

I believe that our communication could be improved, despite most of us are usually responsive overall, it would be great to be inform with unexpected incidents and catch concerns earlier. Improving these areas of code uniformity and communication flow will be the most important thing for us working together for the upcoming sprint. Overall, I'm optimistic, in our next sprint when we integrate our code together and back-end aspects for FoodieFare.

Index, Login, Signup, Contacts & Feedback/Queries Pages (Akashdeep)

I have designed front-end UI pages for Login, Sign-up, Contact, Index and Feedback/Queries pages using HTML, CSS along with some JavaScript. These pages contains header, footer with interactive images and user-friendly interface with easy to fill fields and HTML Buttons. I had some idea regarding coding in html, CSS and JavaScript but this unit helped me a lot in improving my coding skills and will further work to improve this website in next sprint which also includes creating database and link these pages with the DB to make them fully functional and easy to use.

Food Menu page (Preethi)

Designing an engaging and user-friendly connecting page that displays selected stall details including its Name, description, and list of food items with different structures (like Appetisers, Main courses, Desserts, Drinks, and many more). The food items are displayed with photos, names, and prices. Users can select any food item and add them to a cart. This later integrates with the cart and bill page to finalize the order.

JavaScript code is used for loading and displaying the menu items specific to its category. This is also used to filter the menu, hover the items, and Add to the Cart button to pick the items. The CSS is used for a neat structure of displaying the details, styling the menu list with different font sizes, and colours, and enhancing the user experience.

Cart page (Sandriya)

Sprint 1 for our Foodie Fare Website is now completed. The journey was with ups and downs. We have designated each person various tasks to be completed and during sprint1 our team was able to pull that up with all the designing and the front-end part been completed. Initially when I started designing the cart page there were various difficulties like which design would be perfect, how to connect etc. But after my research I was able to do the same. The cart page will help the users to add, edit or remove the items as per their convivence also the bill summary gives you the overall total. For creating the same HTML,

CSS and JavaScript was used. The thing that could be improved was to have regular meeting sessions since we had created UI with different theme which could be avoided with better communication. Lastly, the things that I have learned is to have proper planning, communication and tracking the progress.

Payment Page (Sarfaraz)

The payment page creates a responsive web page for Foodie Fare with a user-friendly interface. It utilizes the Materialize CSS framework for styling and interactivity. The page includes a navigation bar, responsive side navigation for mobile devices, and a payment method selection section. Users can choose between payment options like Pay at Stall, Apple Pay, PayPal, and Debit/Credit Card.

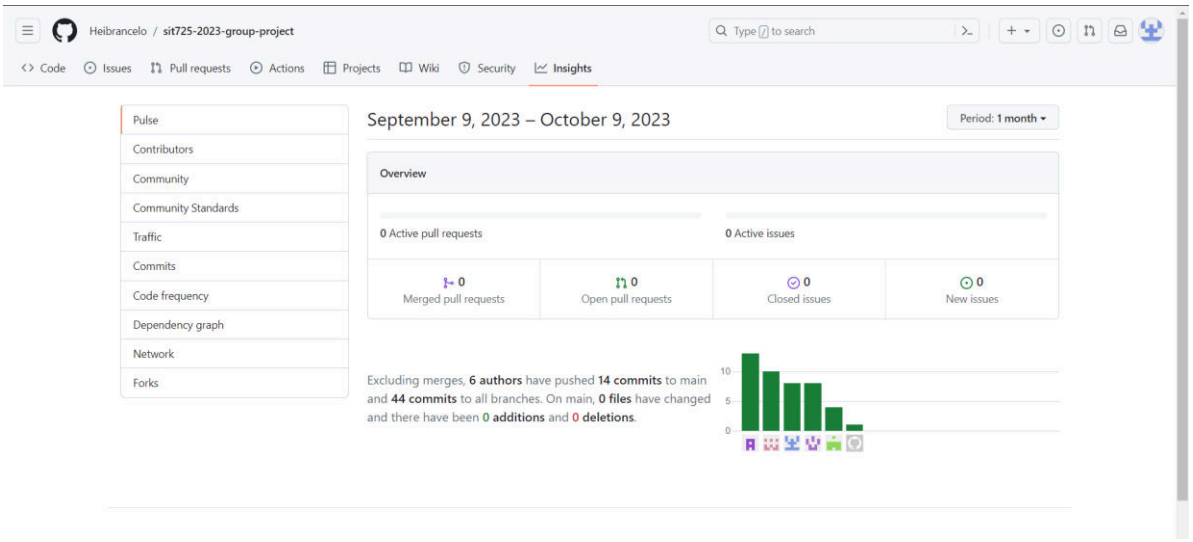
The JavaScript code enhances functionality by toggling the display of different components when different payment options are clicked. Images, forms and bill summary representing are displayed when those payment options are clicked.

The footer contains quick links, providing easy navigation. The code prioritizes user experience and responsiveness, making it suitable for various screen sizes and devices.

15. Appendix 2: Sprint 2 Review

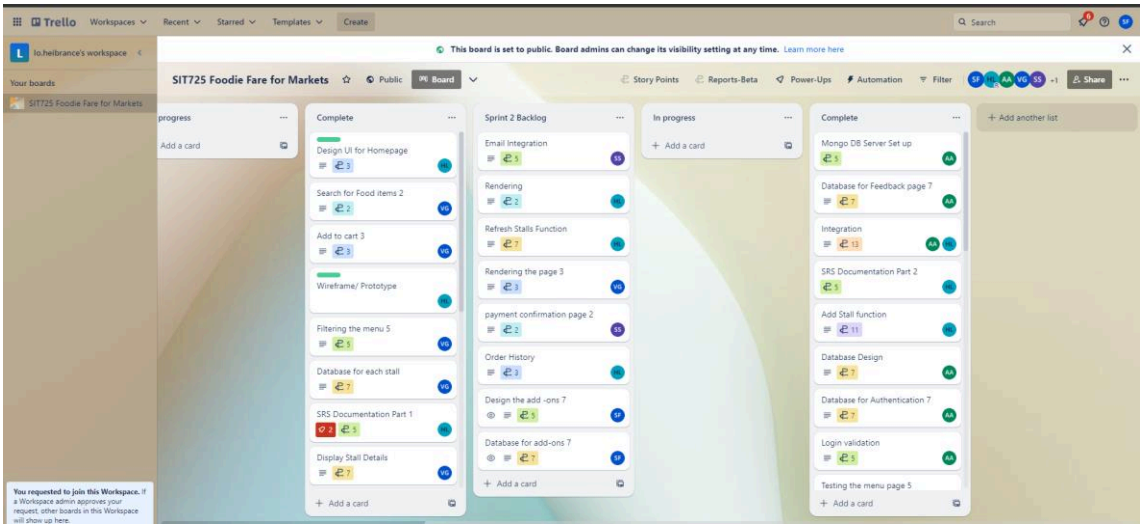
15.1 Screenshot - Git contributor

Git link-- https://github.com/Heibrancelo/sit725-2023-group-project



15.2 Screenshot - Trello Board

Trello link-- https://trello.com/b/tvvOHuJ8/sit725-foodie-fare-for-markets



15.3 Foodie Fare Application Demo Link

Link: SIT725 Foodie Fare Application Demo

15.4 Table of tasks completed

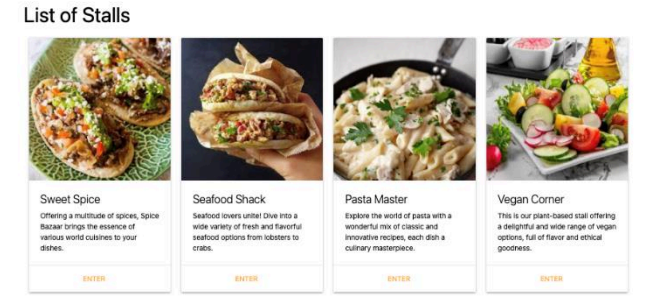
Members	Task completed (Sprint 1)
AkashDeep	• Developed Index page • Developed Login page • Developed signup page • Developed contact page
Heibrance	• Wireframe in Figma • Application Flow • User stories • Developing Dashboard page
Preethi	• Developed food menu page • Retrieve food items from database and display them on page
Sandriya	• Developed for Cart Page • Developed for Bill summary
Sarfaraz	• Developed Payment Page • Created UI components for Credit/Debit, PayPal and Apple Pay

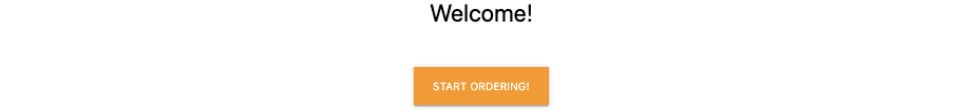
Members	Task completed (Sprint 2)
AkashDeep	• Set-up database connection for the team • Linked my login, signup, help and query pages with the welcome page • Create different collections (Login, Signup, Feedback page) in Mongo DB using InsertOne method in server.js file • Store data for login page in Mongo DB using POST API • Store data for Sign-up page in MongoDB • Store data for Query page in Mango DB using POST method. • Integrated all the HTML pages for the team together
Heibrance	• Developed Dashboard page • Developed User profile page • Created function for adding stalls to database and showing on webpage

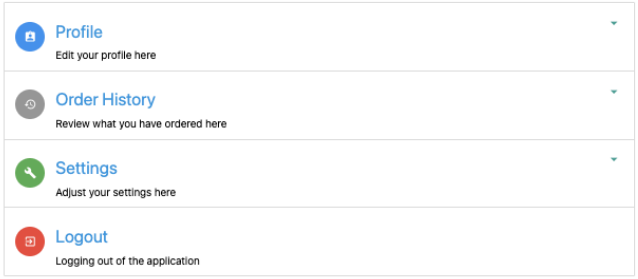
	• Integrating pages with other's
Preethi	• Developed code to display the food menu • Navigate from the home page and link to the cart page. • Display the stall name and the list of items dynamically from the Database using the ID from the previous page. • Filter to search food items. • Filter to display veg and non-veg food items. • Add to cart button to select/order the food item
Sandriya	• Edit items • Remove Items from cart • Developed for Applying Discount • Redirect to payment page and menu page • Developed for Calculating the subtotal and total based on the items added/removed. • Writing GitHub Readme
Sarfaraz	• Developed further features for different payment options • Stored relevant data for each of the payment methods in MongoDB using POST Method • Developed receipt/ticket generation for Pay at Stall option • Created validation for each input field for debit/credit payment mode • Developed payment history page to display payment mode, payment time and payment date using GET method to fetch data from Payment Details collection in MongoDB. • Merged different git branches into the main branch while resolving conflicts.

15.5 Screenshot of the current application

Dashboard, Account (Heibrance)

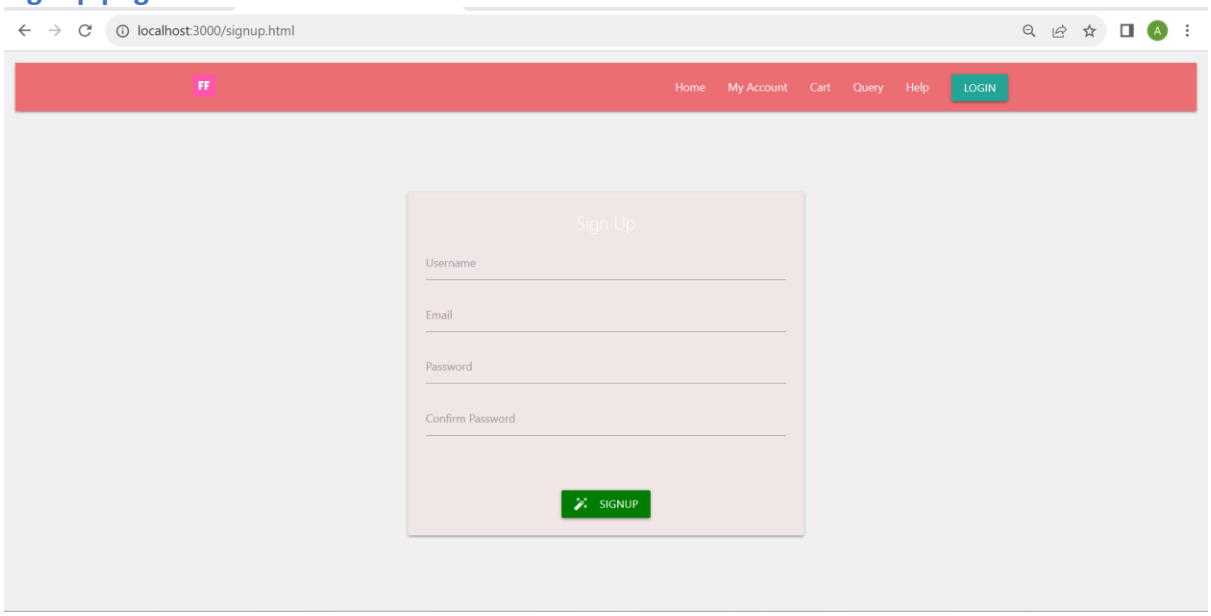




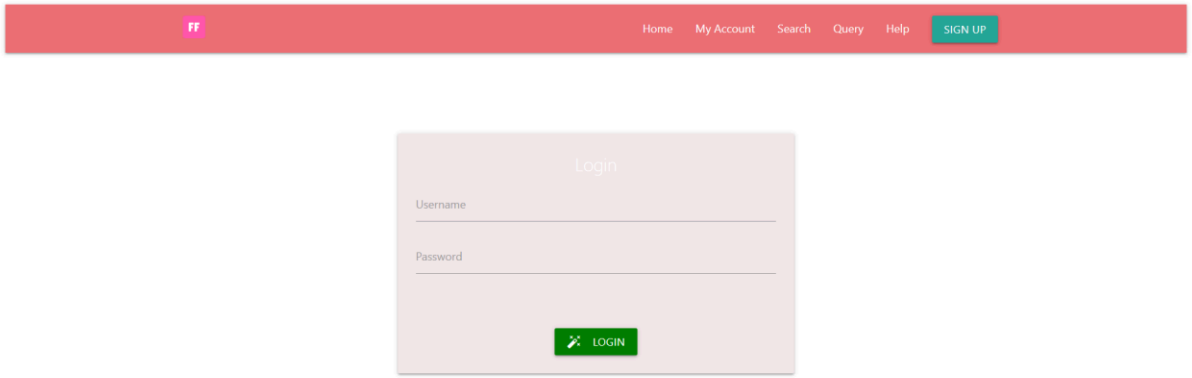


Login, Signup, Contacts & Feedback/Queries Pages (Akashdeep)

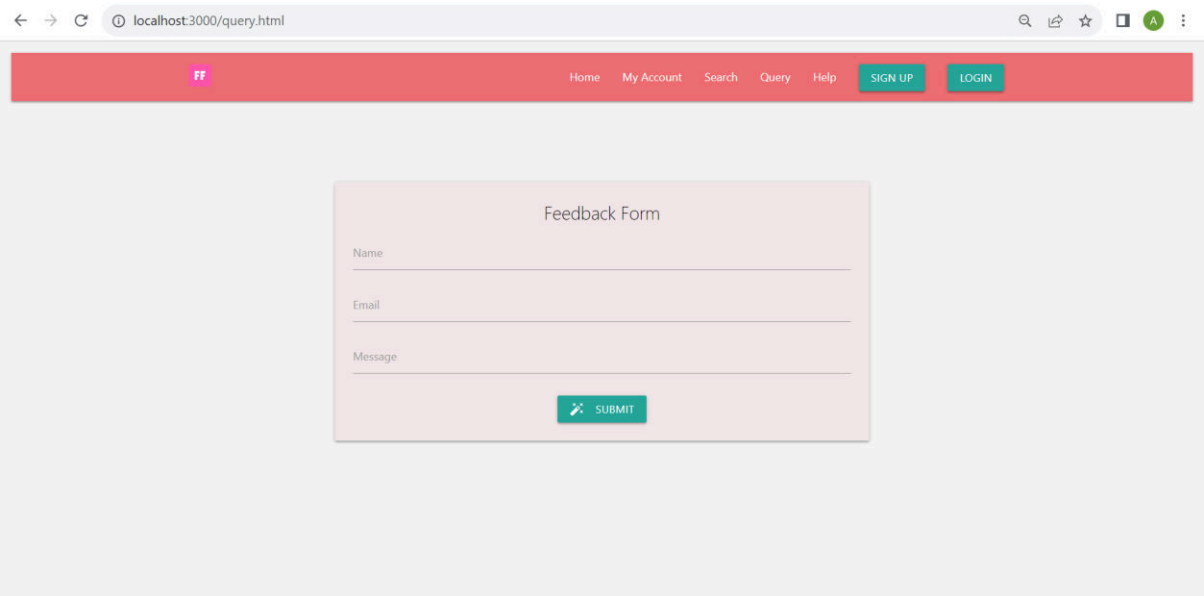
Signup page



Login page

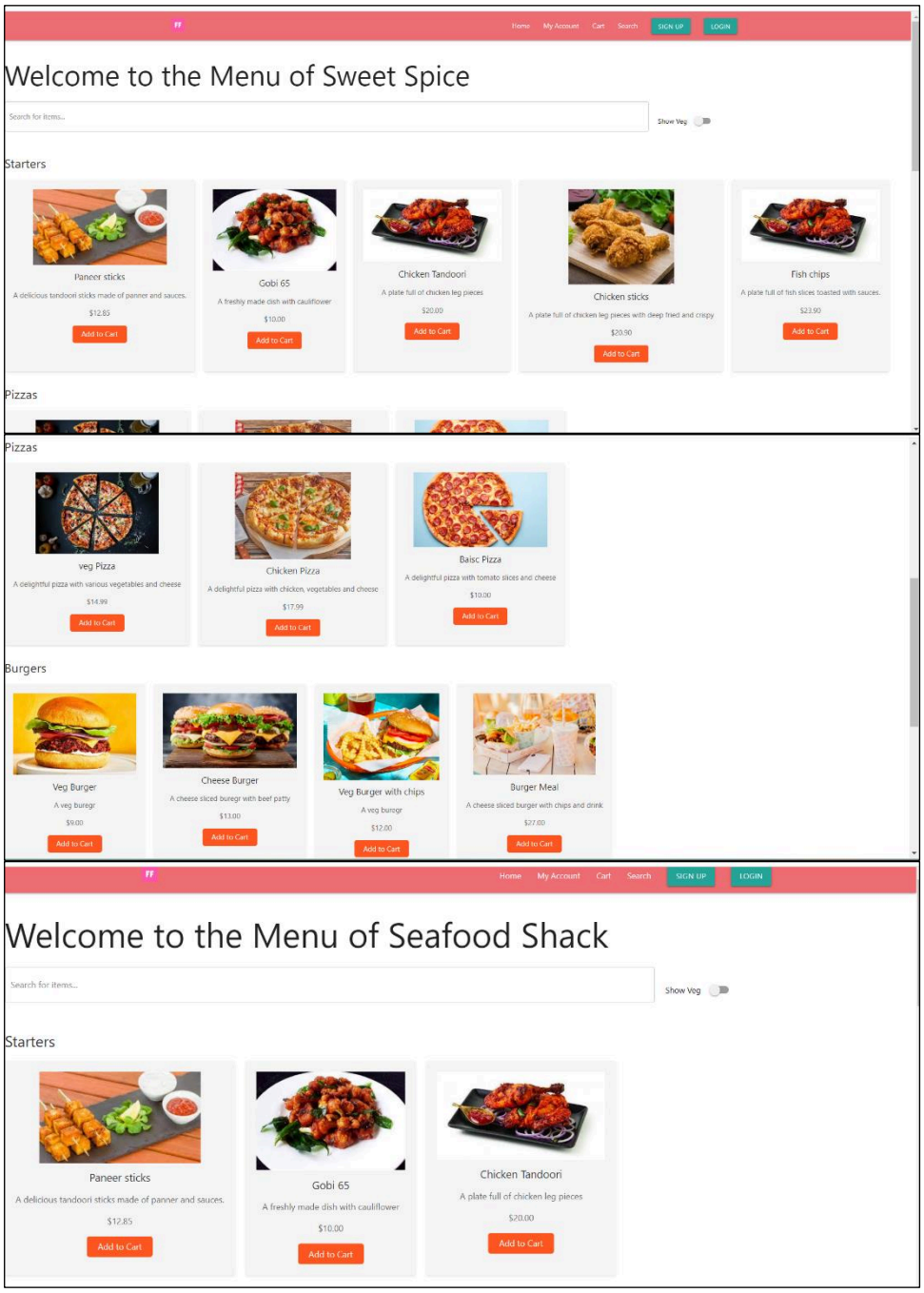


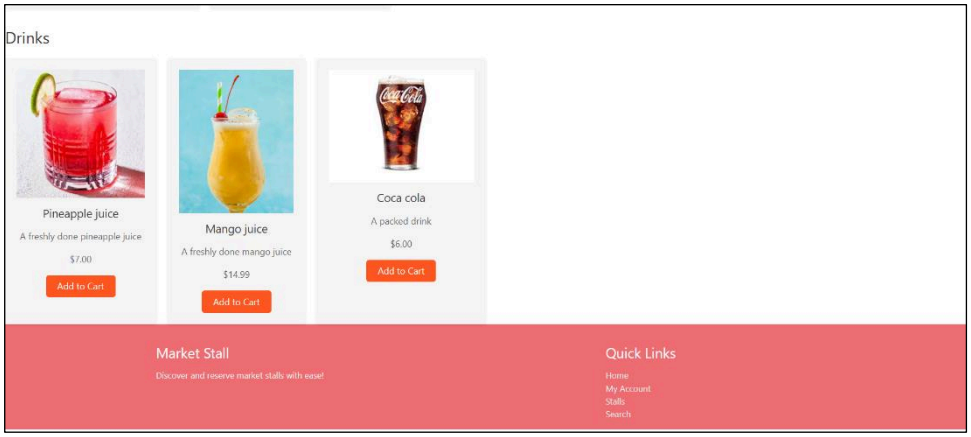
Feedback page



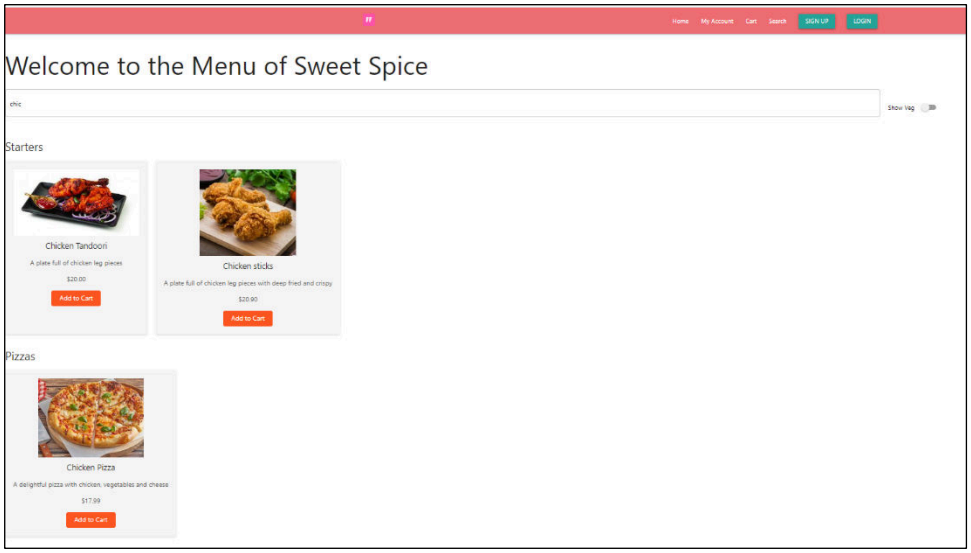
Contact page

Food Menu page (Preethi)

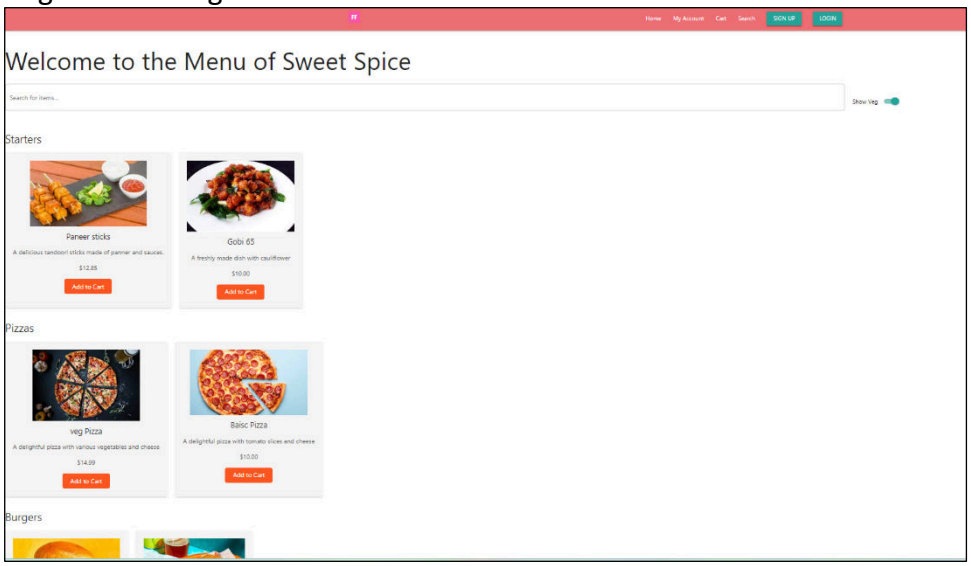




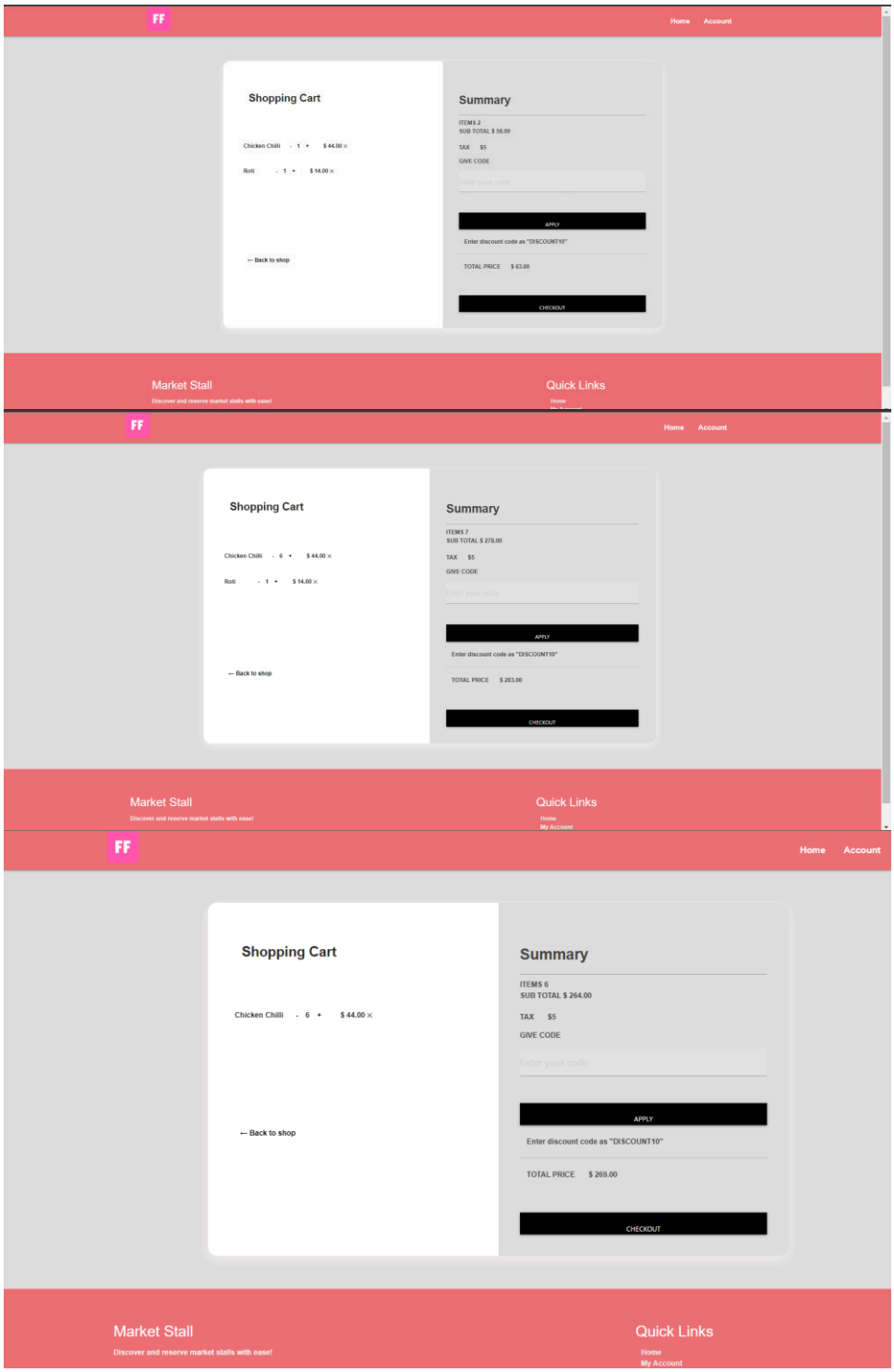
Search filter

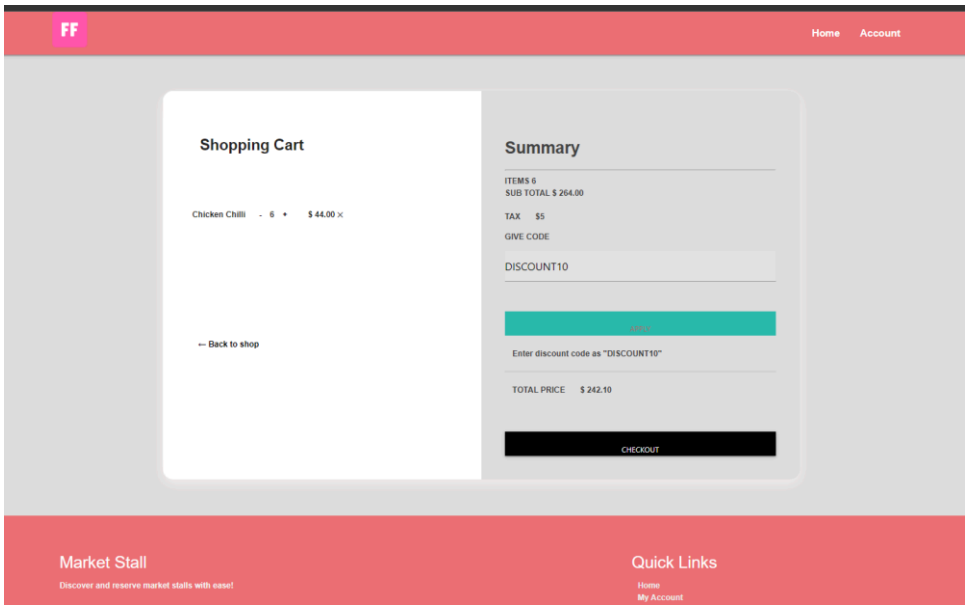


Veg and non-veg filter

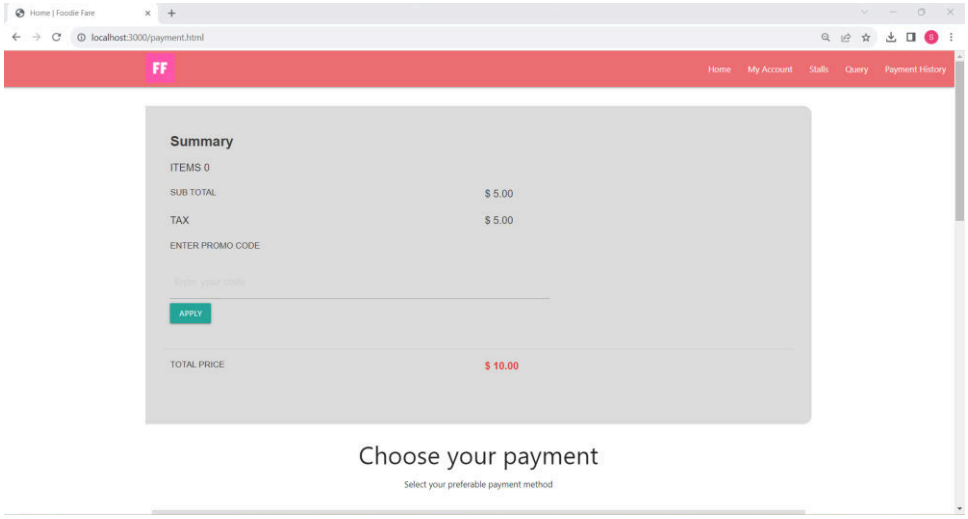


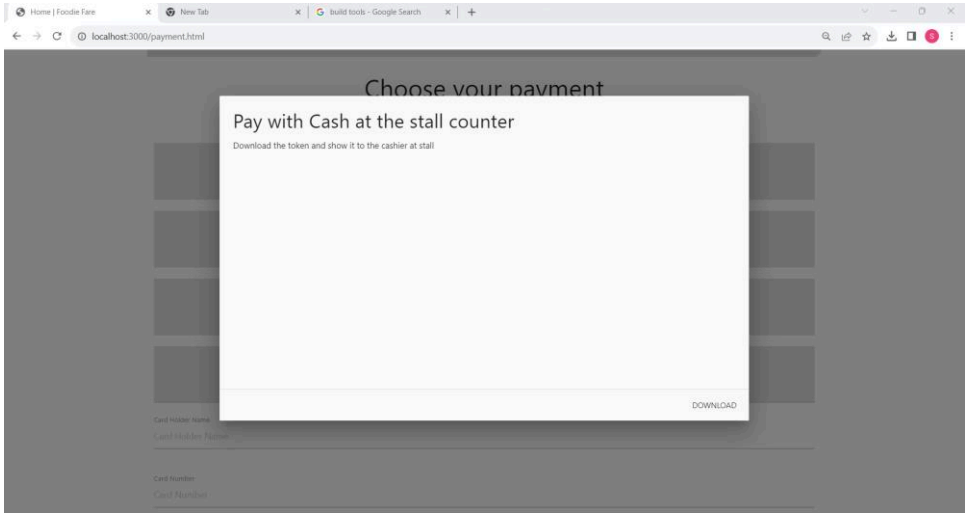
Cart page (Sandriya)

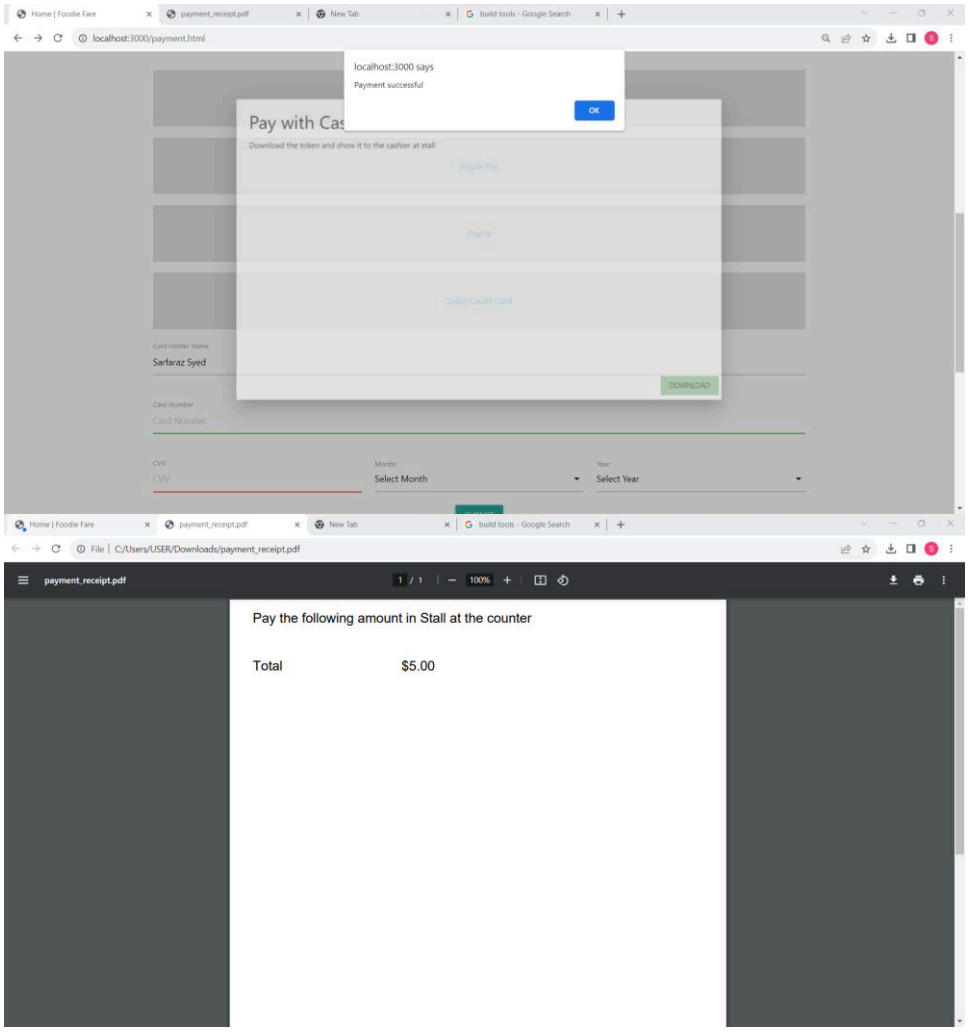


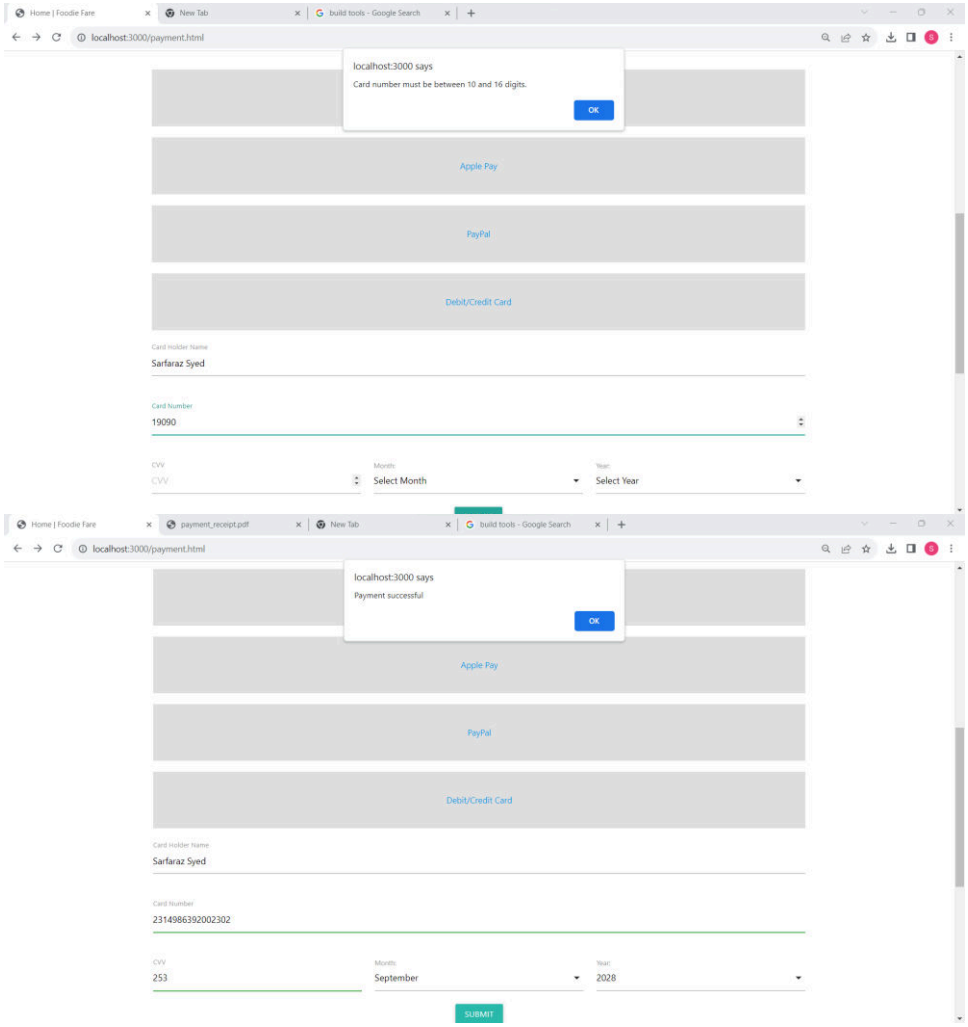


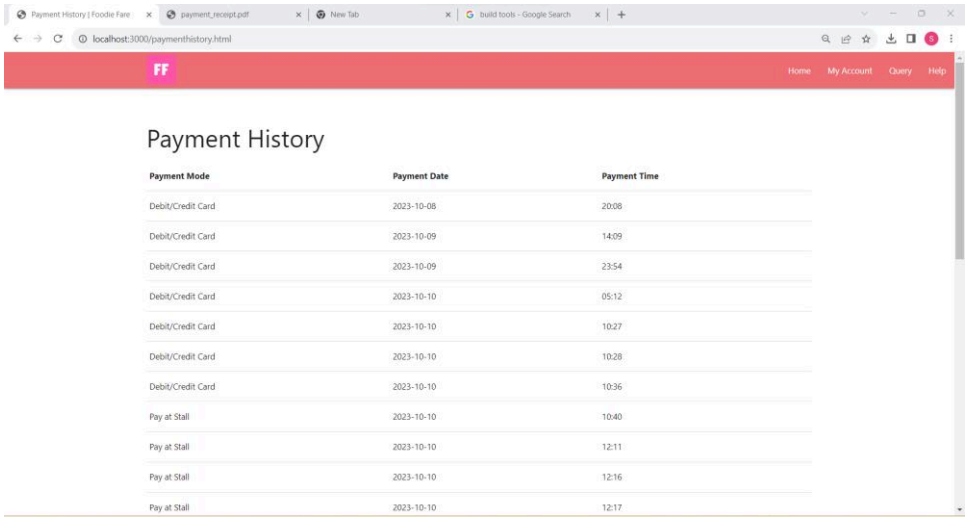
Payment Page (Sarfaraz)




Choose your payment
Pay with Cash at the stall counter
Download the token and show it to the cashier at stall
DOWNLOAD


localhost:3000 says
Payment successful
OK
Sarfaraz Syed
Select Month
Select Year
payment_receipt.pdf
Pay the following amount in Stall at the counter
Total
$5.00


localhost:3000/payment.html
localhost:3000 says
Card number must be between 10 and 16 digits.
OK
Apple Pay
PayPal
Debit/Credit Card
Card Holder Name
Sarfaraz Syed
Card Number
19090
CVV
Month:
Select Month
Year:
Select Year
localhost:3000/payment.html
localhost:3000 says
Payment successful
OK
Apple Pay
PayPal
Debit/Credit Card
Card Holder Name
Sarfaraz Syed
Card Number
2314986392002302
CVV
253
Month:
September
Year:
2028
SUBMIT


localhost:3000/paymenthistory.html
FF
Home
My Account
Query
Help
Payment History
Payment Mode
Payment Date
Payment Time
Debit/Credit Card 2023-10-08 20:08
Debit/Credit Card 2023-10-09 14:09
Debit/Credit Card 2023-10-09 23:54
Debit/Credit Card 2023-10-10 05:12
Debit/Credit Card 2023-10-10 10:27
Debit/Credit Card 2023-10-10 10:28
Debit/Credit Card 2023-10-10 10:36
Pay at Stall 2023-10-10 10:40
Pay at Stall 2023-10-10 12:11
Pay at Stall 2023-10-10 12:16
Pay at Stall 2023-10-10 12:17

15.6 100 works Reflection (All)

Dashboard, Account (Heibrance)

During Sprint 2, I accomplished the UI and eventually established a connection to the database. The 'Add cards' feature, which enables users to add to the database, is now complete and operational. However, group communication has been difficult due to a member frequently showing up at the last minute and being unresponsive. Despite working on separate GitHub branches, there wasn't much activity on GitHub itself. Overall, we made progress in backend development during Sprint 2, but communication and collaboration are areas that need improvement within our group. In the near future, I would emphasise the importance of communication and implement regular stand-up meetings to ensure all member are up to speed.

Index, Login, Signup, Contacts & Feedback/Queries Pages (Akashdeep)

In sprint 2, I have improved the UI of my pages and made them more interactive and user friendly with easy to fill fields and interactive buttons. I have set up the Mango DB database connection for my team and created GET & POST methods to store and fetch data for my login, signup and query page. These GET & Post methods will run onclick of ‘Login’, ‘Signup’ and ‘submit’ buttons on the respective pages and will create a new record in the database in their specific collection accordingly that I have created using server.js file.

Food Menu page (Preethi)

The foodie fare website sprint 2 has been completed now, as part of the project my task is to design a page that displays the list of food items when any particular restaurant is selected. This page is navigated when the user clicks any stall to order food and from this page, we navigate to the cart(next) page. For this particular page, we pass the stall name and ID from the previous page so that the name and food menu are dynamically displayed based on parameters passed internally. Users can click on the Add to Cart button displayed after each item to select and order them. This whole item is passed to the next cart page for bill summary. On this page, we also have features like search where users can search for any food item, and a filter to display veg and non-veg items. Lastly, I feel we should have had more communication and kept track of completing the project in time but by this project, I had improved my coding skills in HTML and JavaScript.

Cart page (Sandriya)

The foodie fare website sprint 2 has been completed now. There were many things that we were able to pull up, but few things are still missed or lagged. My tasks required me to get the menu items that are added into cart should be displayed but unfortunately before the sprint 2 I was not able to get the previous code. Hence, I decided to go with the predefined values where the items in the cart could be edited by increasing or decreasing the quantities, removing the items, etc. Also, the bill summary page functionality was being added. It included subtotal, tax, also displaying the promo codes and when applied the exact promo

code the discount was being applied which will be seen in the total amount. The checkout page was integrated with the payment page and the back to shop page used to redirect to the previous menu pages. The header and footer have been added to the page. Overall, in sprint 2 was able to pull up with the functionality. Lastly the things that I learned in this sprint was how important communication is and it is important to meet the deadlines or else the entire project can lag.

Payment Page (Sarfaraz)

In Sprint 2, significant enhancements were made to diversify payment options, store method-specific data in MongoDB through the POST method and implement receipt generation for the "Pay at Stall" feature. Additionally, validation protocols were established for each input field in the debit/credit payment mode, ensuring data accuracy. A payment history page was introduced, leveraging the GET method to retrieve and display payment details such as mode, time, and date from the MongoDB collection. Git branches were effectively managed, resolving conflicts during the seamless merging of changes into the main branch, fostering collaborative development practices and version control.