simplilearn

FOODBOX

PROJECT

{ Sprint Planning and Project Specification }

Developer Details

Tahoor Khan – Full Stack Java Developer

tahoork136@gmail.com

FOODBOX

This document contains sections as follow:

1. Introduction
2. Project GitHub
3. Modules in the Project
4. Technology used in the project
5. Sprint Planning and Task Completion
6. Product's capabilities, appearance, and user interactions.
7. Project Screenshots (Front-end)
8. Hosting website on AWS S3 instance
9. Conclusions

1. INTRODUCTION –

FoodBox aka Delicioso!! is an online food delivery web application for ordering food items of different cuisines from a restaurant.

FoodBox is a restaurant chain that delivers food items of

different cuisines at affordable prices.

2. PROJECT GITHUB LINK –

Link	**https://github.com/TahoorKhan786/Foodbox_Simplilearn.git**

3. Modules in the Project

- Registration Page.
- Login Page.
- User Login.
- Admin Login.
- Dashboard.
- Search Products.
- Add Cart/View Cart.
- View Previous Active Orders.
- Payment Gateway Page.
- Order Summary Confirmation Page.

4. Technology Used:

a. Front End:

HTML.

CSS3 and Bootstrap4.

TypeScript.

Angular, Angular Material for View.

b. VS Code: As an IDE to design frontend of the application.

c. Git: To connect and push files from the local system to GitHub.

d. GitHub: To store the application code and track its versions

e. Scrum: An efficient agile framework to deliver the product incrementally.

f. Back End:

- Eclipse IDE.
- Java Programming.
- Searching and Sorting
- Spring Boot DevTools.
- Spring Web and Spring Data JPA.

g. Testing and DevOps: Selenium. Jenkins. Docker and TestNG.

5. Sprints Planning and Accomplishments

The project is planned to complete in three sprints, The Task will be accomplished within sprint as below:

SPRINT-1

1. Creation of flow of project
2. User can login/sign-in to the application.
    - As a user, I should be able to register myself.
    - As a user, I should be able to log in to the website.
    - As a user, I can add products to the cart only if I'm logged in.

3. User can Filter/search products

- As a user, I should be able to search the products.
- As a user, I should be able to filter the products from the menu.
- As a user, I should be able to add products to my cart.

4. User can view cart, edit the products in the cart and can proceed to Payment Gateway.

- As a user, I should be able to view the products.
- As a user, I should be able to edit the cart.
- As a user, I should be able to see the total bill of the products in the cart.
- As a user, I should be able to do various operations in the cart.
- As a user, I should be able to check out and make the payment.
- As a user, I should be able to view the Order Summary and print order confirmation.

5. Admin Operations –

- As an admin, I should be able to login to the website.
- As an admin, I should be able to manage the products.
- As an admin, I should be able to manage the purchases.
- As an admin, I should be able to manage users registered on the website.

6. Testing and Bug Fixing

- Testing of Project for further deployment.

Explanation of the product capabilities, appearance, and user interactions

To Explain the product capabilities, there are sub- sections configured to highlight appearance and user interactions for the project.

1. Importing a Spring Boot project in Eclipse IDE with the required dependencies.
2. Writing Java Program to create model/entity classes –

   Admin.java

   Cart.java

   Customer.java

   Product.java

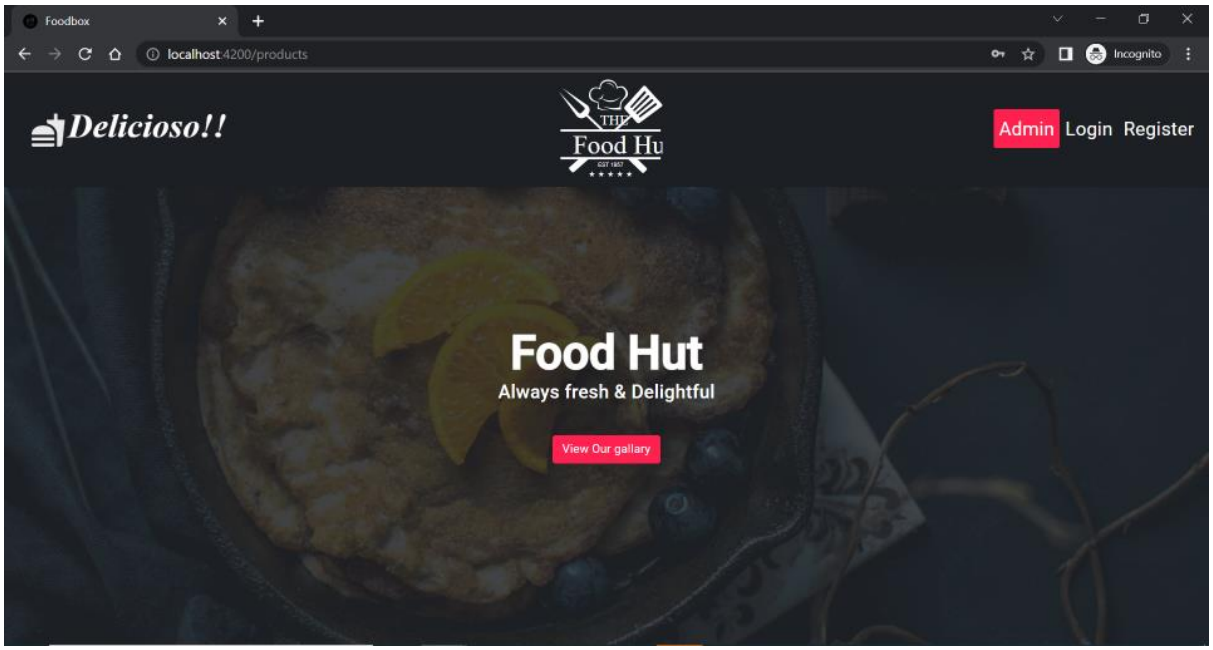
   Purchase.java
3. Writing a Java Program, to create all the repositories.
4. Writing a Java Program, to create the controllers for all repositories.
5. Writing a Java Program, to write config class and exception class.
6. Creating application.properties file to write all the database related code.
7. After Successful completion of Backend part of the application, we will move to the frontend part which will be created using Angular-15/HTML5/CSS3
8. Create a folder where you want to create the application and open cmd on the same location and type ng new <project-name>
9. Project will be created, now open the project folder in VS code and run ng serve command to run the project on localhost.
10. Create all the required modules and components in the project/src/app folder.

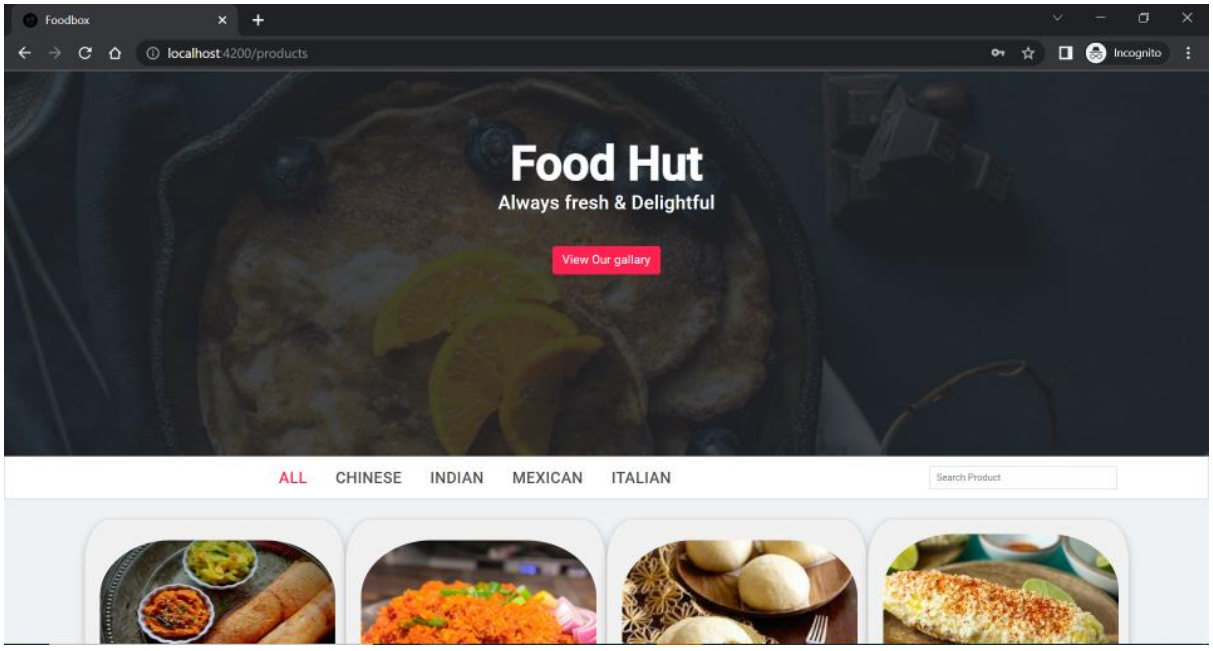
11. Once both the front-end and back-end start working perfectly. The project is pushed to GitHub Repository.

Project Screenshots

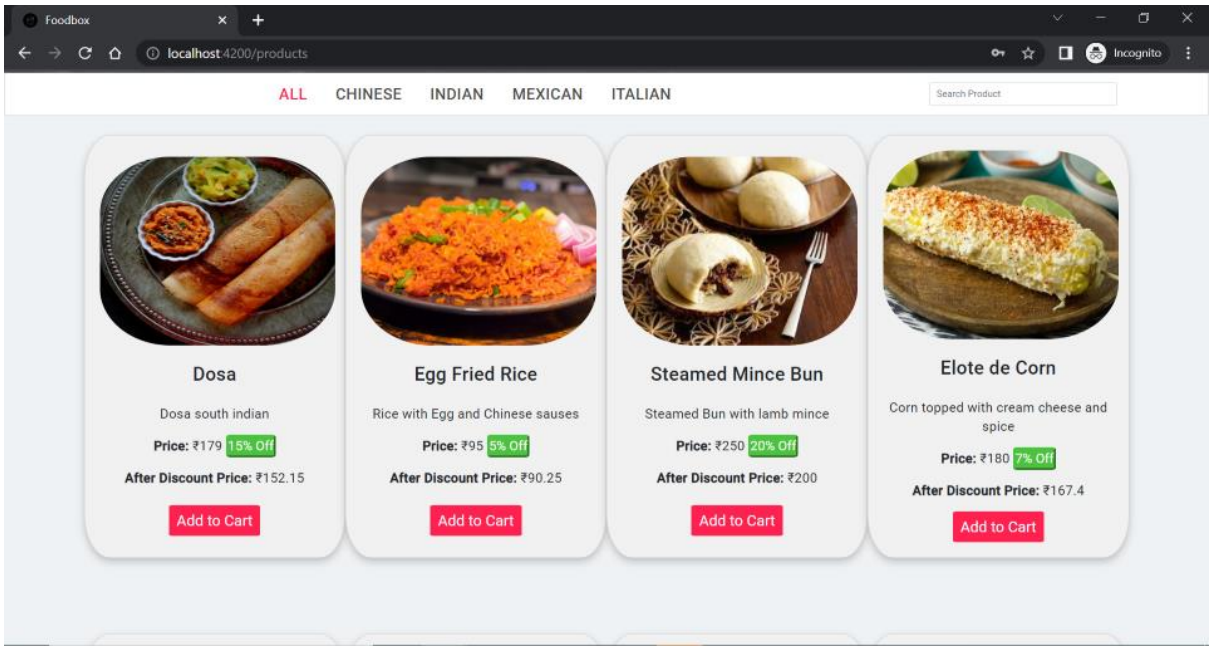
FRONT-END

1. HOME

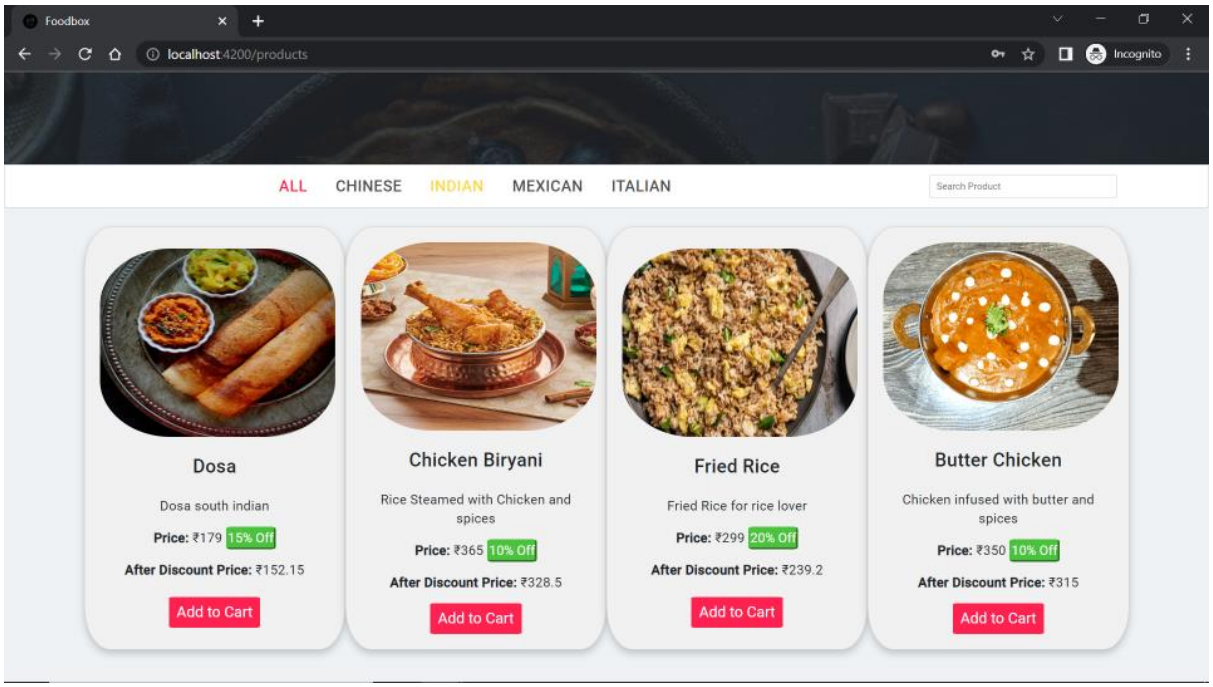




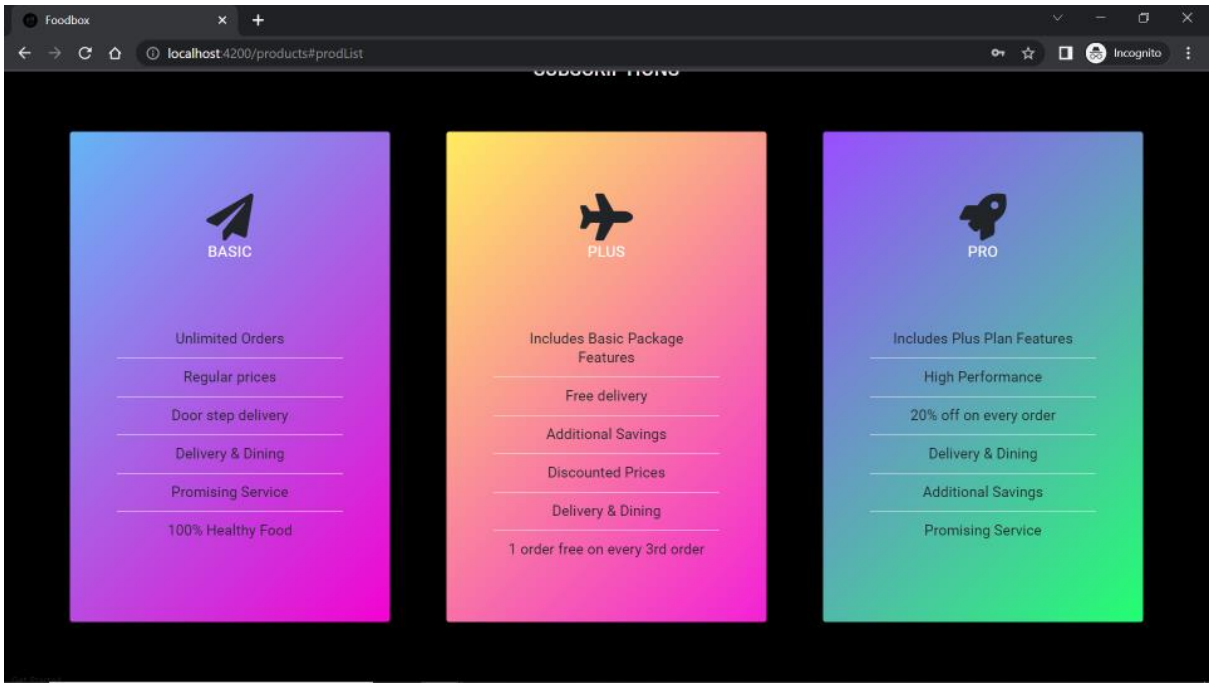
2. Products

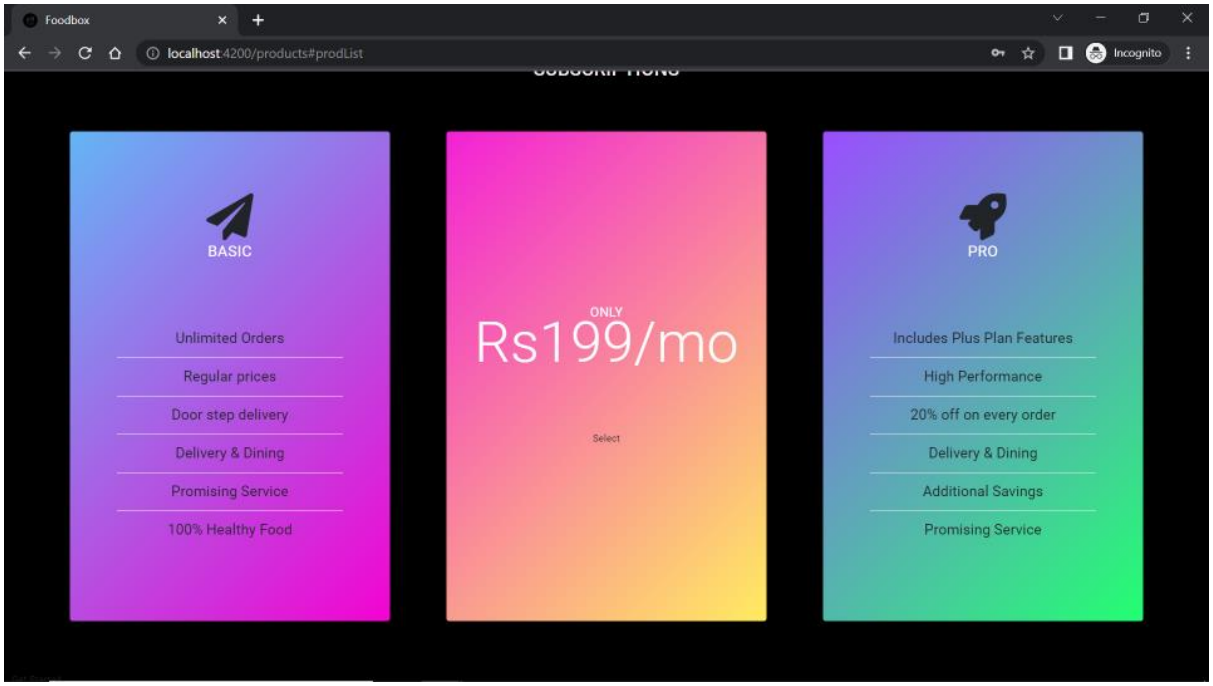


Fetching all the products based on category Indian

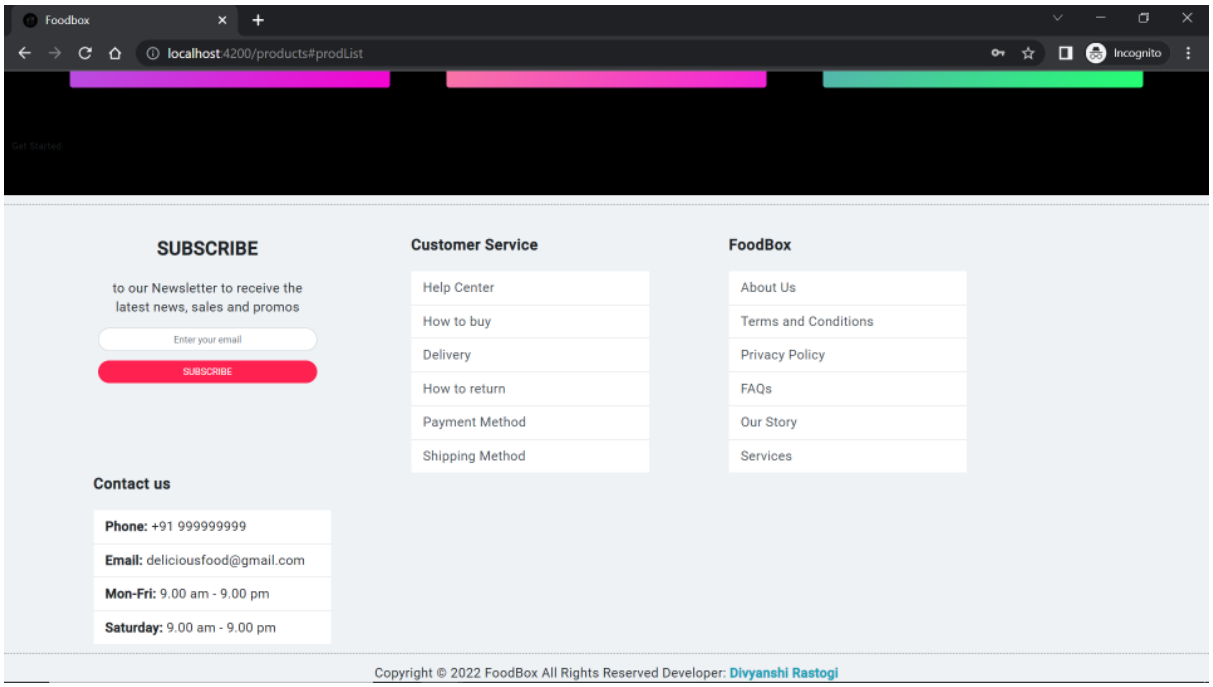


3. Added few subscriptions cards.

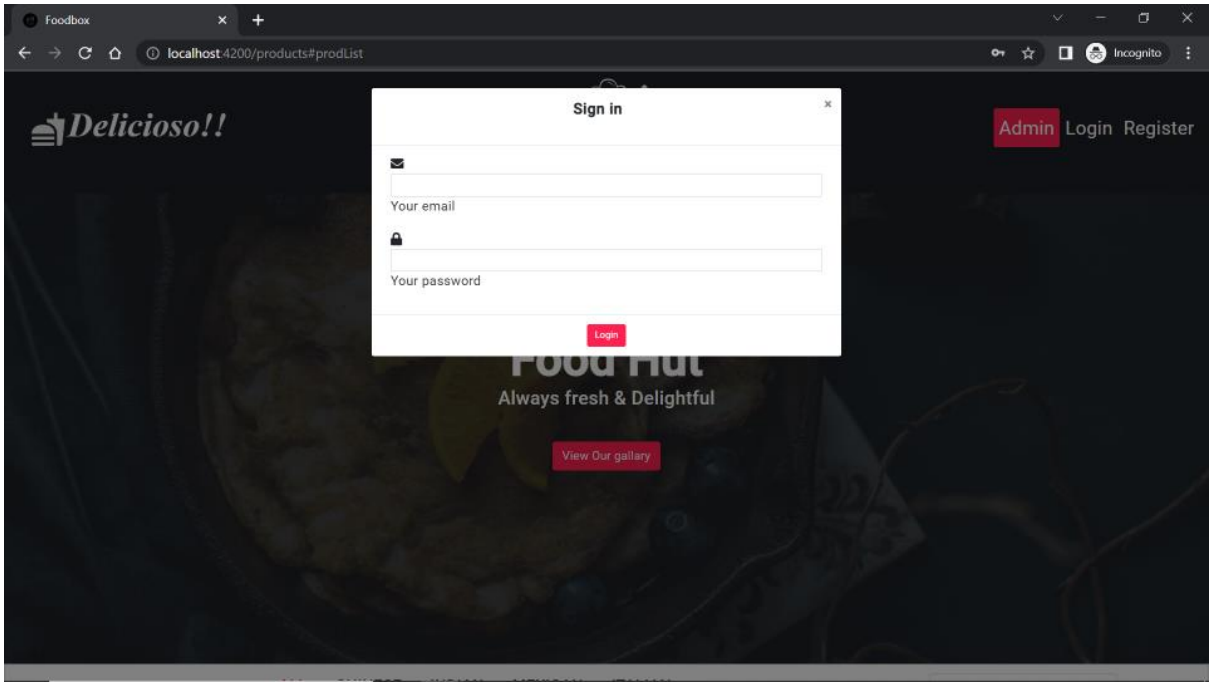




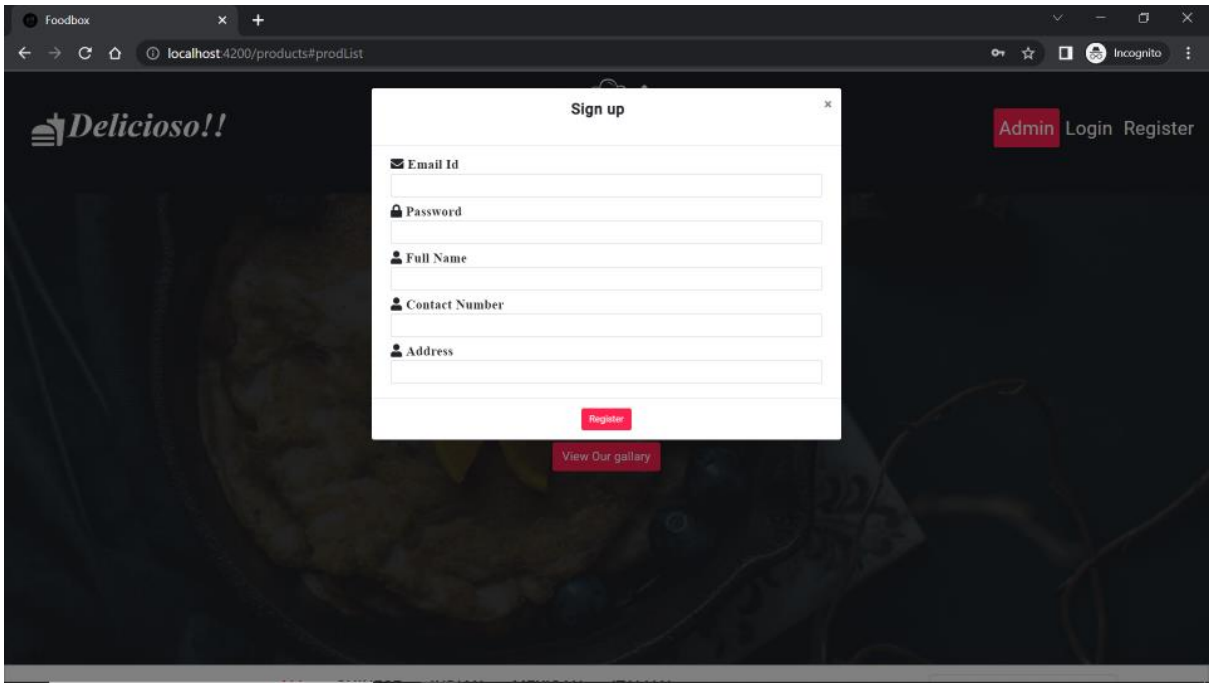
4. Footer



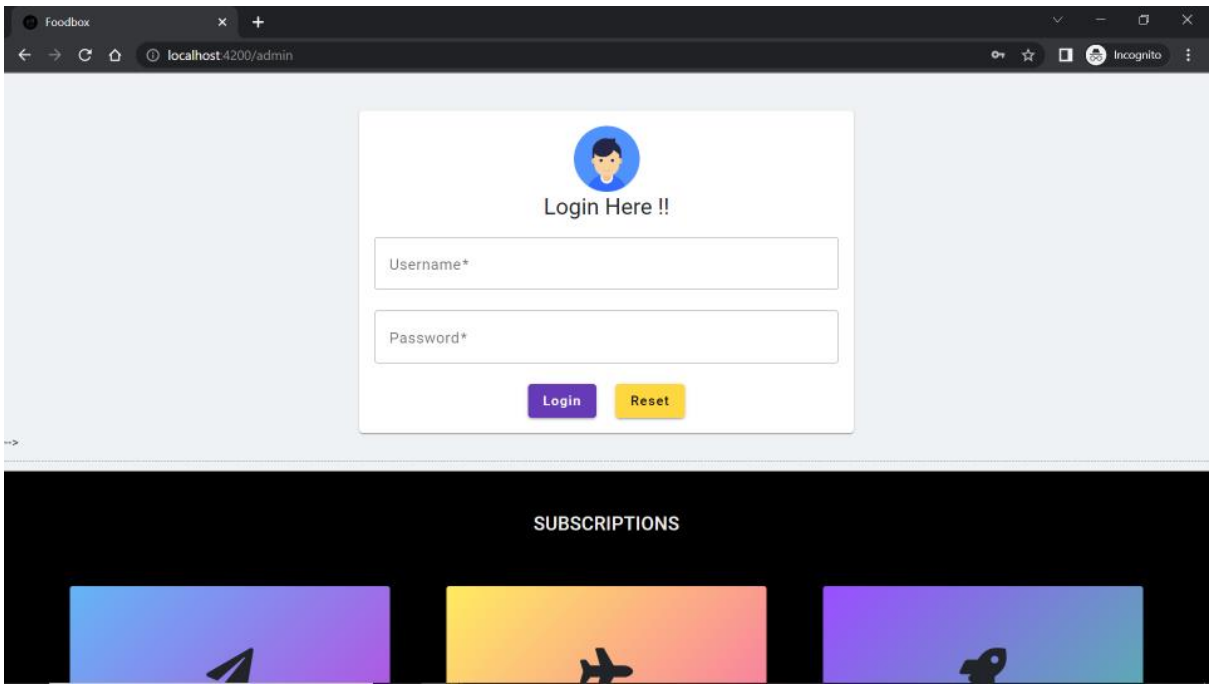
5. User Login



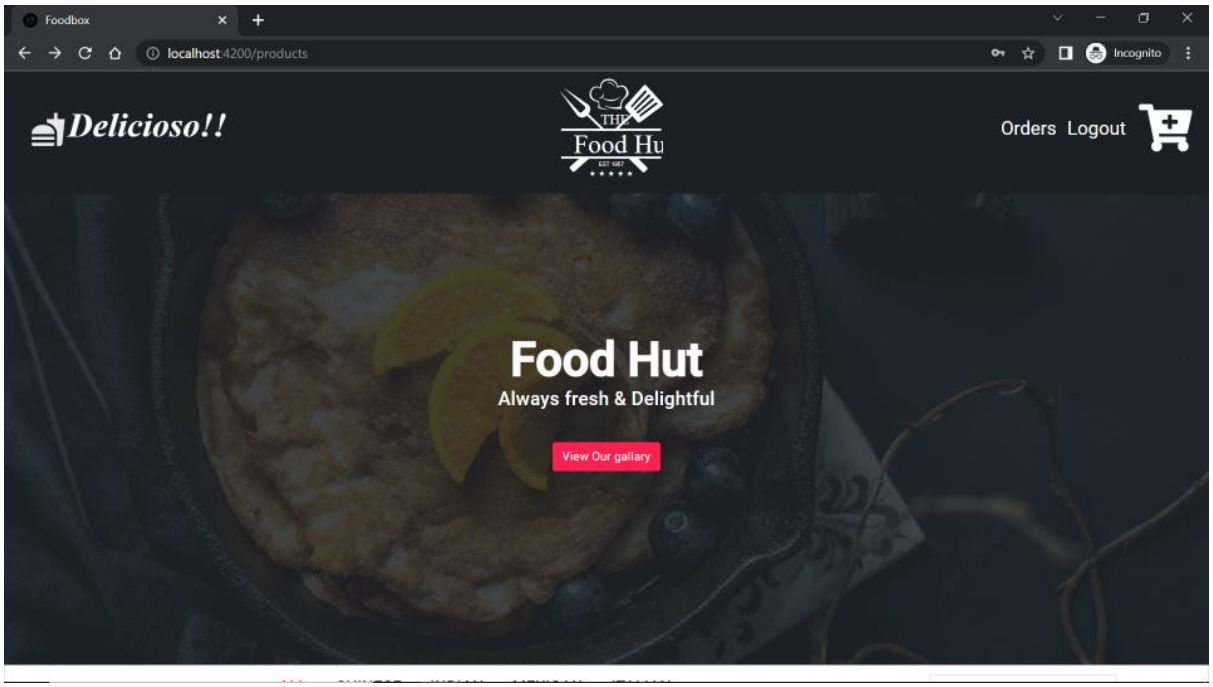
6. Register User



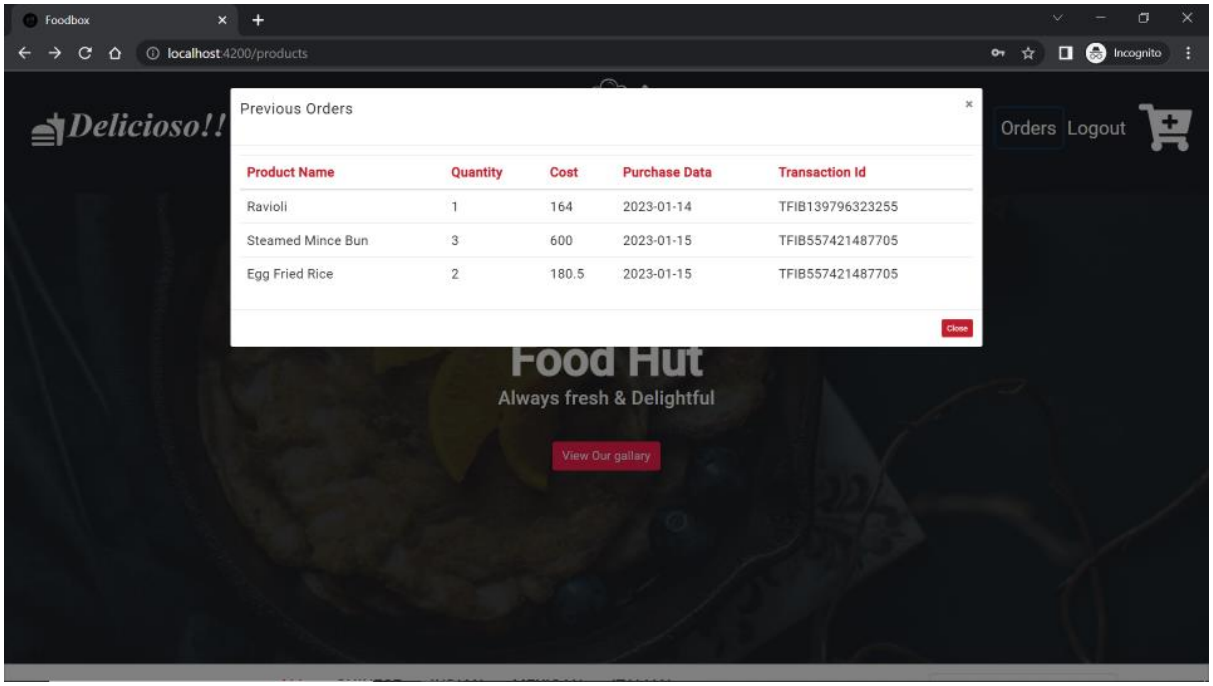
7. Admin Login



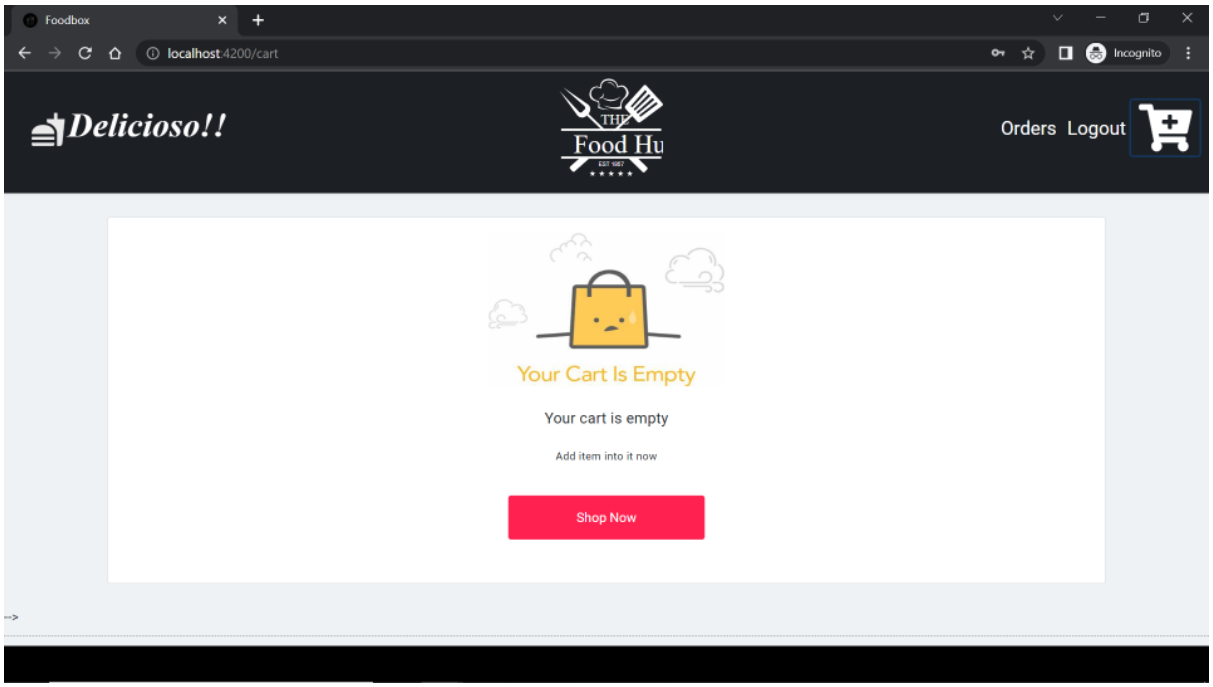
8. After Login



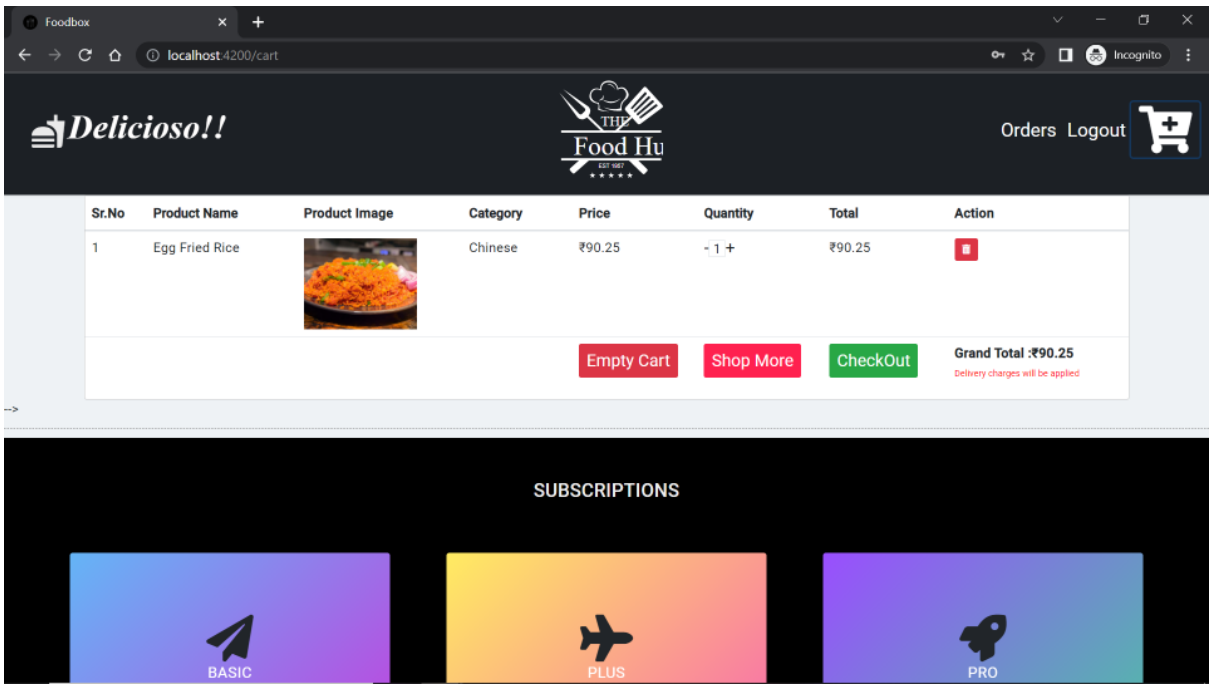
9. To view Previous Orders



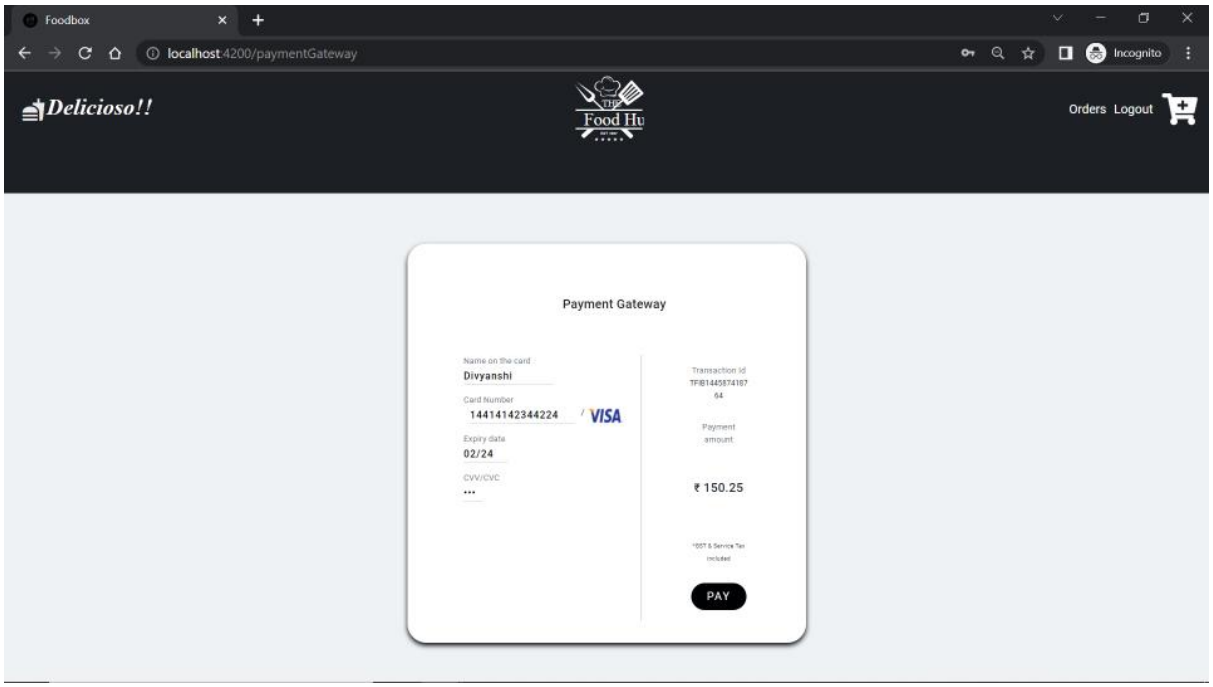
10. Click on Cart



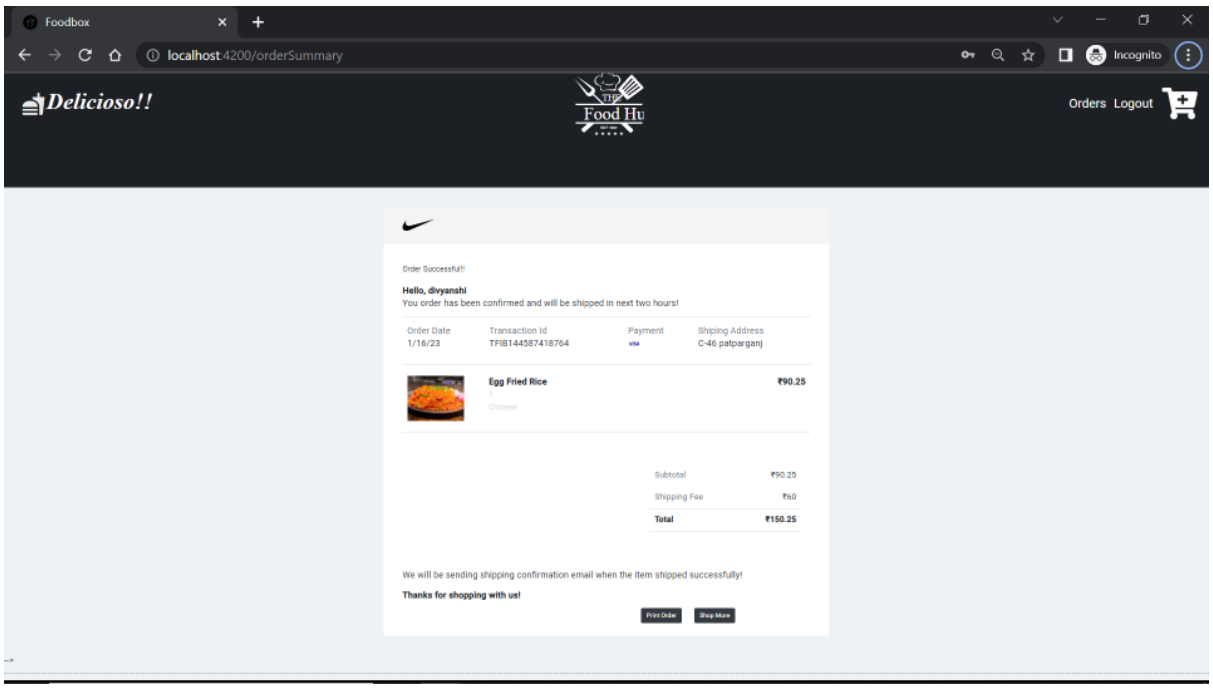
11. After adding a product to the cart



12. Once you click on Checkout



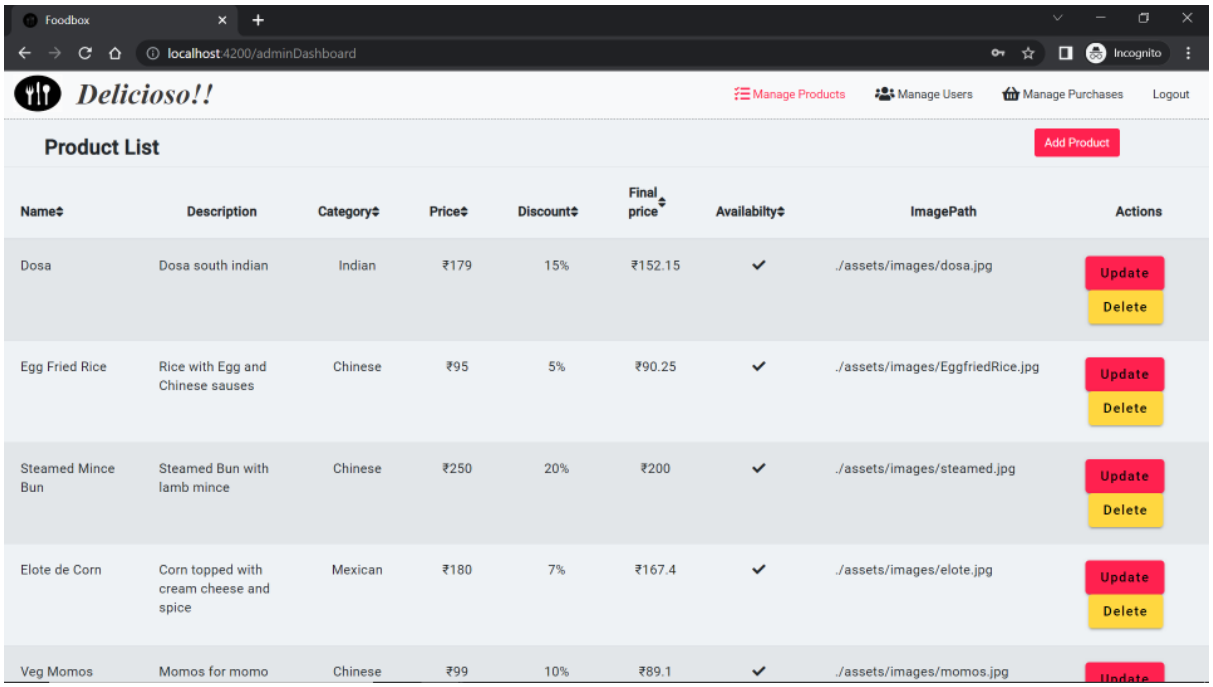
13. After Successful Payment.



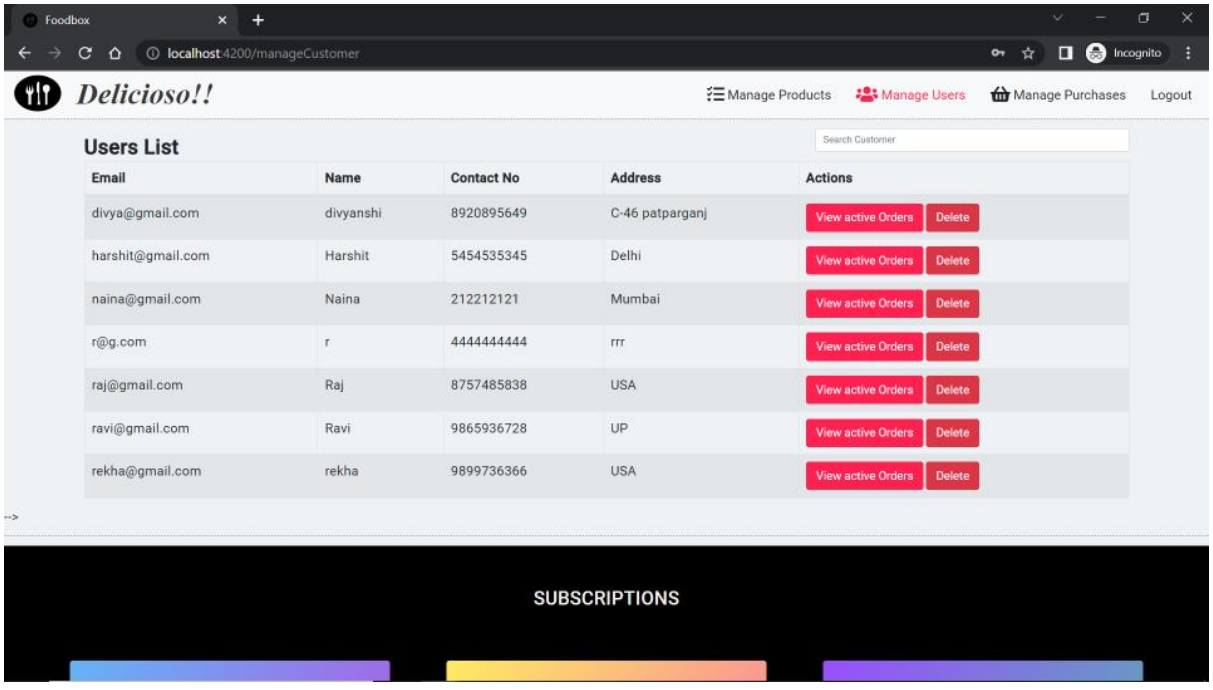
14. After logging in as an admin

Username – root

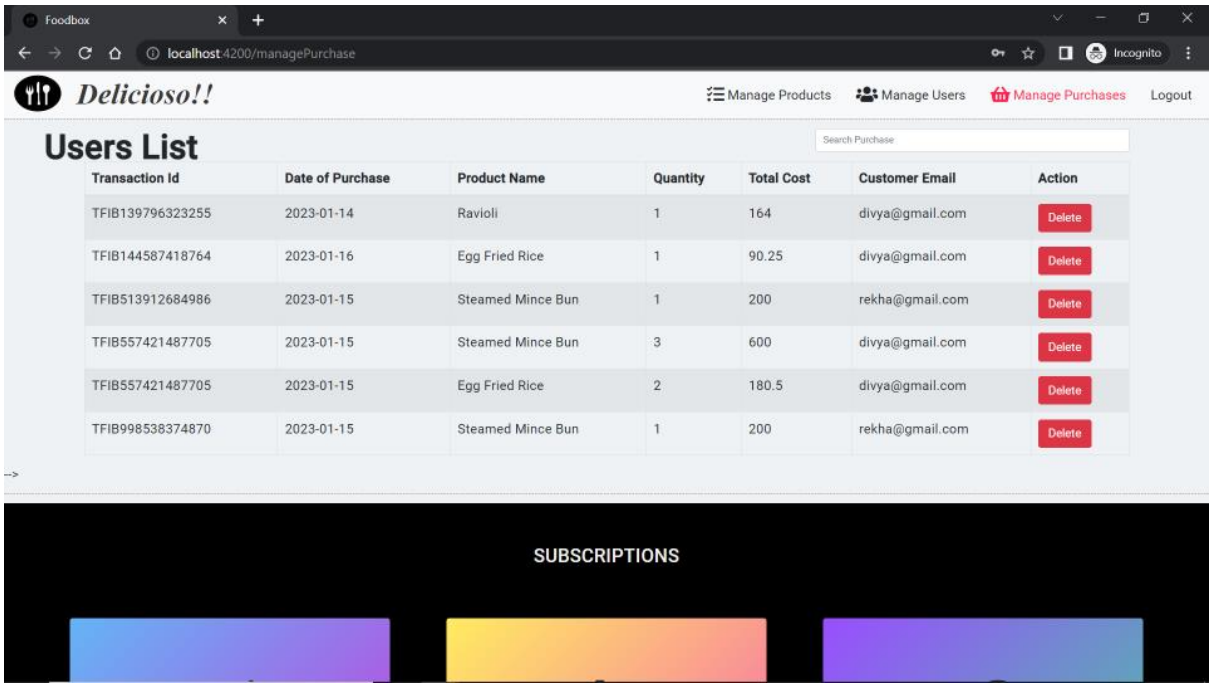
Password - root



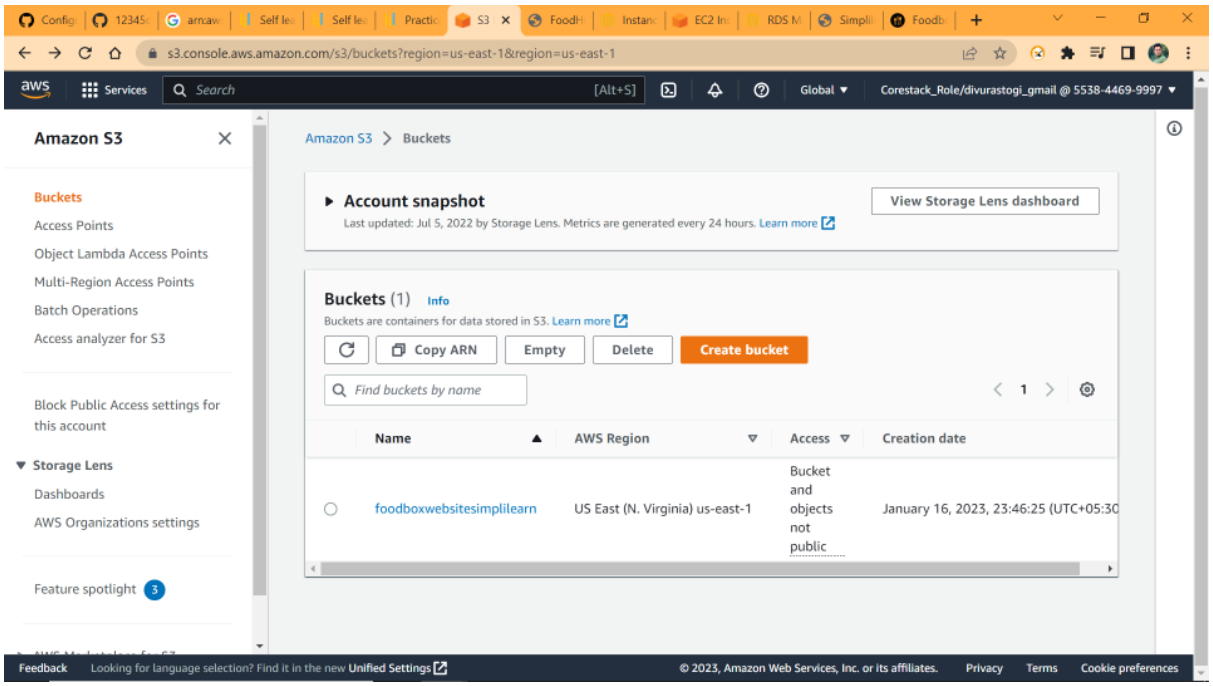
Manage Users

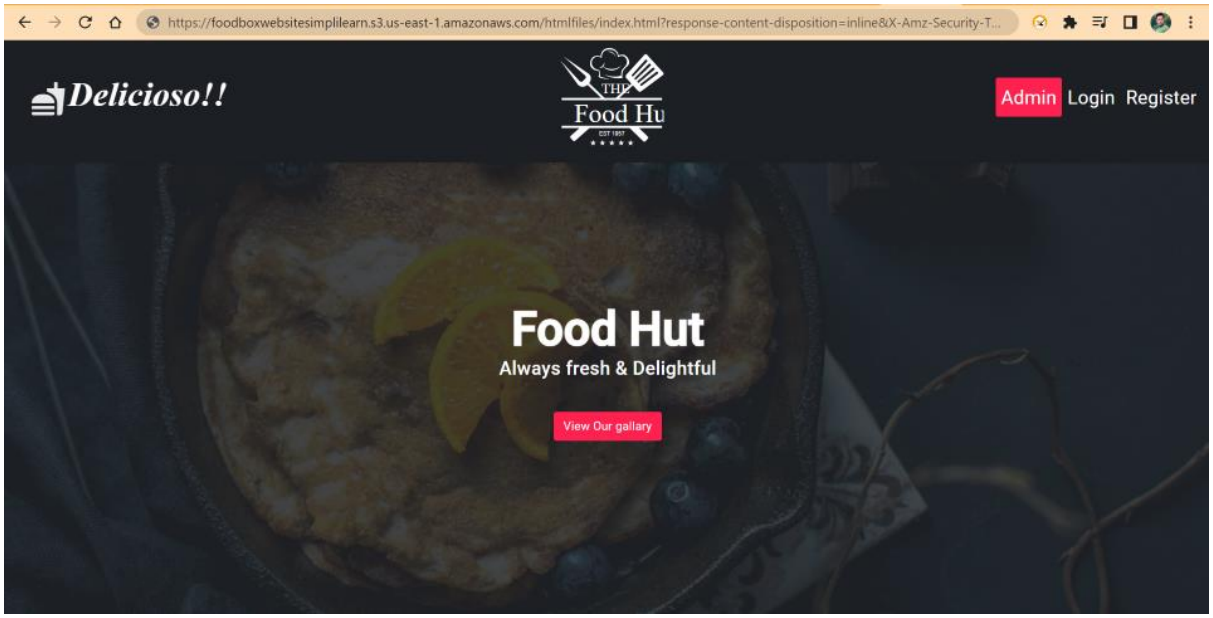


Manage Purchases



7. Hosting the website on Amazon S3 instance





9. Conclusions:

Further enchantment can be made as files can be stored in database for better retrieve and use of the user interactions.

The Data stored into the database can be secured and While retrieval we can use database query to select the information.