**Foodbox**

1. **Foodbox**

**This document contains sections for:**

* Project Description
* [Core concepts used in project](file:///C:\Users\Asus\Downloads\LockedMe%20-%20Virtual%20Key%20for%20Repositories.docx#Core_concepts)
* [Flow of the Application](file:///C:\Users\Asus\Downloads\LockedMe%20-%20Virtual%20Key%20for%20Repositories.docx#Flow).
* Project Users Stories : ( Agile and Scrum )
* Git Repositories
* How to run project
* Outputs
* [Unique Selling Points of the Application](file:///C:\Users\Asus\Downloads\LockedMe%20-%20Virtual%20Key%20for%20Repositories.docx#USP)
* [Conclusions](file:///C:\Users\Asus\Downloads\LockedMe%20-%20Virtual%20Key%20for%20Repositories.docx#Conclusions)

The code for this project is hosted at : https://github.com/deepak-pvsn/capstone-project-foodbox-webapp

The project is developed by P V S N Deepak.

* 1. **Project Description:**

DESCRIPTION

Create a dynamic and responsive online food delivery web application for ordering food items of different cuisines from a restaurant.

**Background of the problem statement:**

Foodbox is a restaurant chain that delivers food items of different cuisines at affordable prices. It was established in 2014 in Bengaluru, India. It had been serving fine all these years, however, the business analysts noticed a decline in sales since 2016. They found out that the online ordering of food items with companies, such as Swiggy and Foodpanda were gaining more profit by eliminating middlemen from the equation. As a result, the team decided to hire a Full Stack developer to develop an online food delivery web application with a rich and user-friendly interface.  
You are hired as the Full Stack Java developer and are asked to develop the web application. The management team has provided you with the requirements and their business model so that you can easily arrange different components of the application.

**Features of the application:**

1. Registration
2. Login
3. Payment gateway
4. Searching
5. Filtering
6. Sorting
7. Dynamic data
8. Responsive and compatible with different devices

**Recommended technologies:**

1. Database management: MySQL and Oracle
2. Backend logic: Java programming, NodeJS
3. Frontend development: JSP, Angular, Bootstrap, HTML/CSS, and Javascript
4. Automation and testing technologies: Selenium, Jasmine, and TestNG
5. DevOps and production technologies: Git, GitHub, Jenkins, Docker, Kubernetes, and AWS

**Project development guidelines:**

* The project will be delivered within four sprints with every sprint delivering a minimal viable product.
* It is mandatory to perform proper sprint planning with user stories to develop all the components of the project.
* The learner can use any technology from the above-mentioned technologies for different layers of the project.
* The web application should be responsive and should fetch or send data dynamically without hardcoded values.
* The learner must maintain the version of the application over GitHub and every new change should be sent to the repository.
* The learner must implement a CI/CD pipeline using Jenkins.
* The learner should also deploy and host the application on an AWS EC2 instance.
* The learner should also implement automation testing before the application enters the CI/CD pipeline.
* The learner should use Git branching to do basic automation testing of the application in it separately.
* The learner should make a rich frontend of the application, which is user- friendly and easy for the user to navigate through the application.
* There will be two portals in the application, namely admin and user portal.

**Admin Portal:**  
The admin portal deals with all the backend data generation and product information. The admin user should be able to:

* Add or remove different cuisines to or from the application to build a rich product line
* Edit food item details like name, price, cuisine, description, and offers to keep it aligned to the current prices
* Enable or disable the food items

**User Portal:**  
It deals with the user activities. The end-user should be able to:

* Sign-in to the application to maintain a record of activities
* Search for food items based on the search keyword
* Apply filters and sort results based on different cuisines to get the best deals
* Add all the selected food items to a cart and customize the purchase at the end
* Perform a seamless payment process
* Get an order summary details page once the payment is complete
  1. **Core concepts used in the project:**
  2. **Front End:**

HTML.

CSS3 and Bootstrap4. TypeScript.

Angular, Angular Material for View.

* 1. **VS Code:** As an IDE to design frontend of the application.
  2. **Git:** To connect and push files from the local system to GitHub.
  3. **GitHub:** To store the application code and track its versions
  4. **Scrum:** An efficient agile framework to deliver the product incrementally.
  5. **Back End:**

Eclipse IDE.

Java Programming.

Searching and Sorting Spring Boot DevTools.

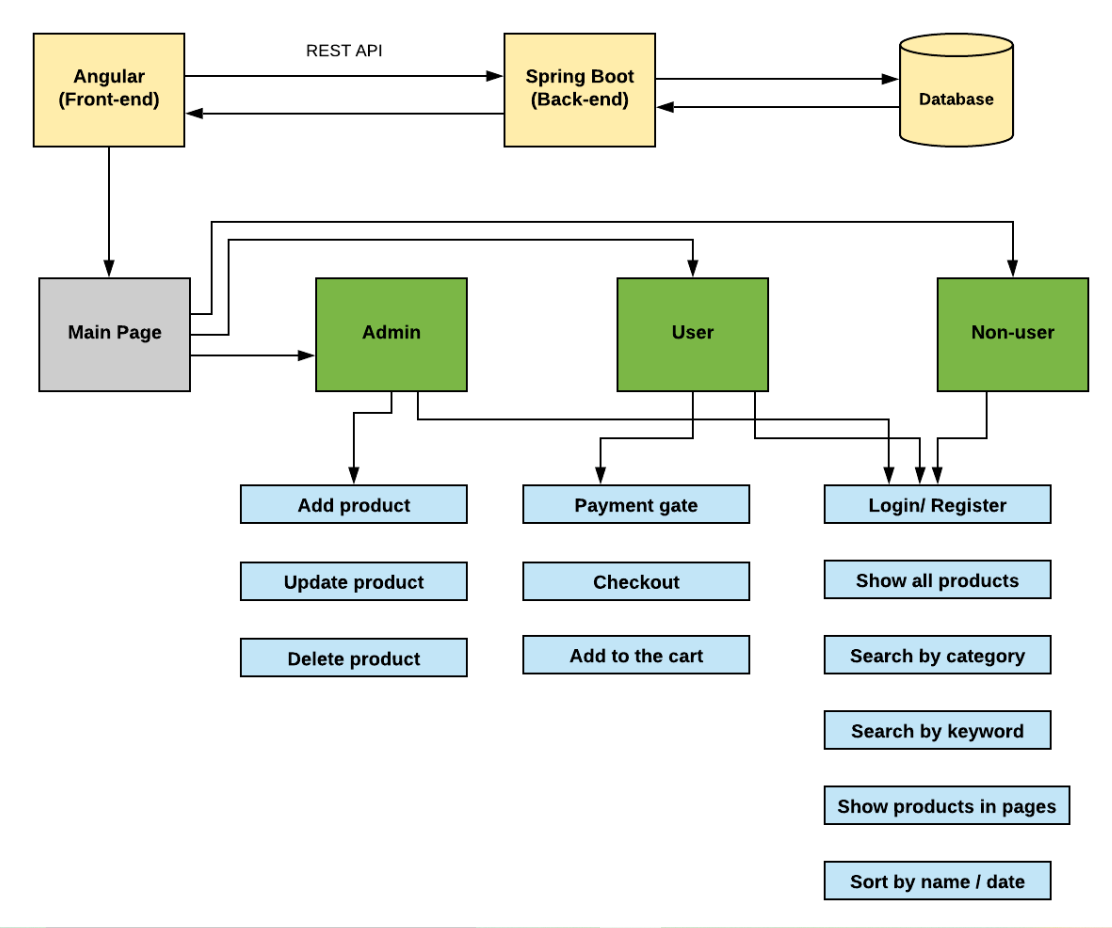
Spring Web and Spring Data JPA.

* 1. **Testing and DevOps:**

Selenium. Jenkins.

Docker and TestNG..

1. **Architecture diagram / flow chart**



* 1. **Project Users Stories : ( Agile and Scrum )**

The project is planned to be completed in 3 sprints. Tasks assumed to be completed in the sprint are:

* Creating the flow of the application
* Initializing git repository to track changes as development progresses.
* Writing the Java program to fulfill the requirements of the project.
* Testing the Java program with different kinds of User input
* Pushing code to GitHub.

1. As a user I want to order my food by logging into portal so that I can store my credentials.

2. As a user I want registration page so that I can access Foodbox app.

3. As a user I want login page to authenticate a user to access Foodbox app.

4. As a user I want to add any number of foods.

5. As a user I want to search my food, delete my food.

6. As a user I want to pay amount using payment gateway.

7. As a user I want view the cart and add product to the cart.

8. As a Admin I want to build registration page to enter user in system.

9. As a Admin I want to build login page so that we can authenticate user.

10. As a Admin create user interaction with console input.

11. As a Admin I need to display all the foods.

12. As a Admin I need to add or update the product.

13. As a admin I need to Authenticate the payment gateway system.

14. As a admin I need to track the details of users.

Sprint 1

1. As a user I want to order my food by logging into portal so that I can store my credentials.

2. As a user I want registration page so that I can access Foodbox app.

3. As a user I want login page to authenticate a user to access Foodbox app.

4. As a user I want to add any number of foods.

Sprint 2

5. As a user I want to search my food, delete my food.

6. As a user I want to pay amount using payment gateway.

7. As a user I want view the cart and add product to the cart.

8. As a Admin I want to build registration page to enter user in system.

Sprint 3

9. As a Admin I want to build login page so that we can authenticate user.

10. As a Admin create user interaction with console input.

11. As a Admin I need to display all the foods.

12. As a Admin I need to add or update the product.

13. As a admin I need to Authenticate the payment gateway system.

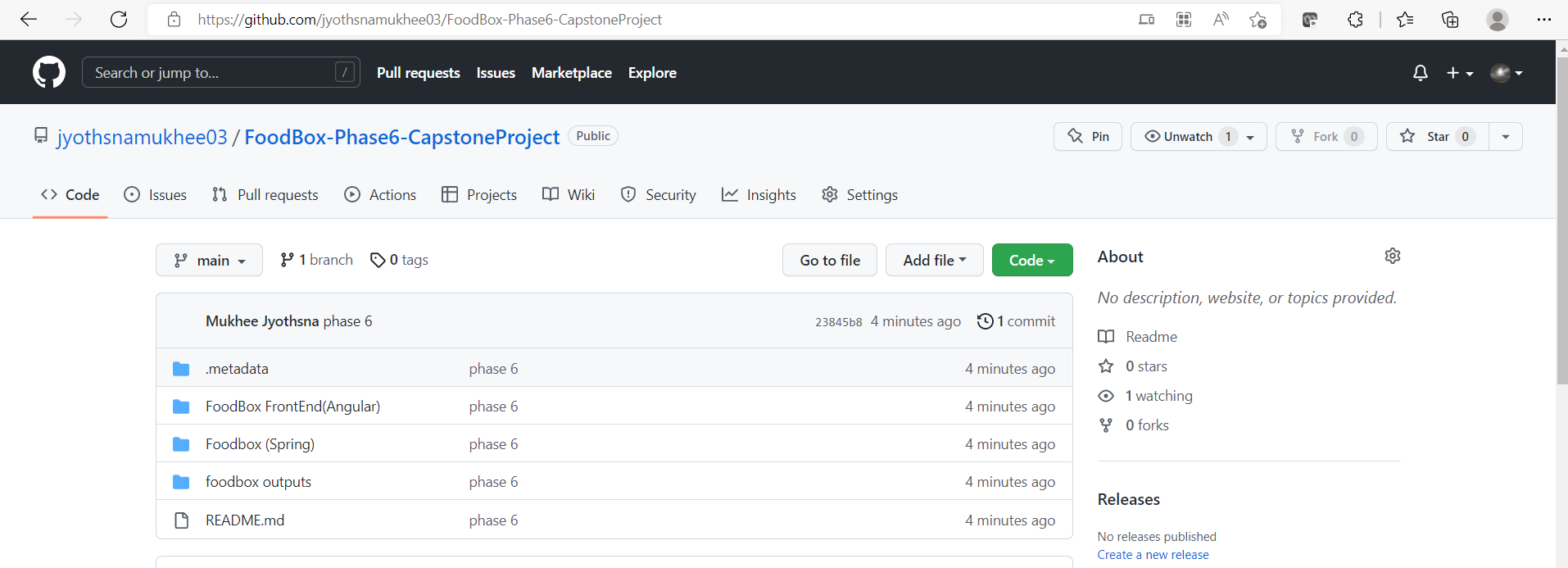
14. As a admin I need to track the details of users.

**3. Project git Repositories**

1. link : https://github.com/deepak-pvsn/capstone-project-foodbox-webapp

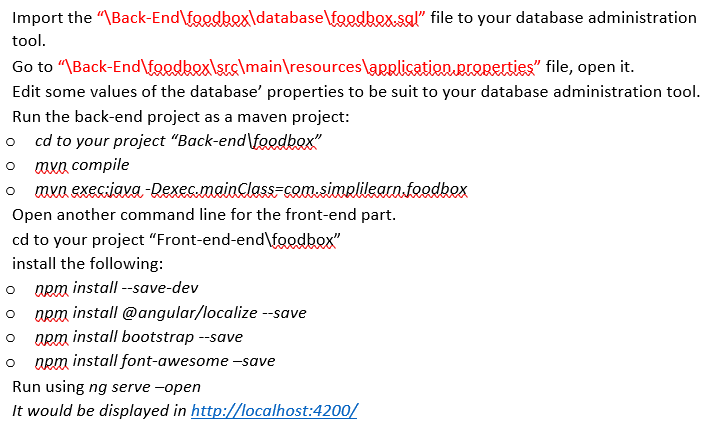
2.clonegit: git clone https://github.com/deepak-pvsn/capstone-project-foodbox-webapp

3. Screen shot :

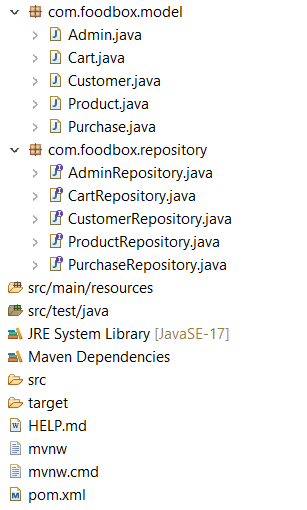
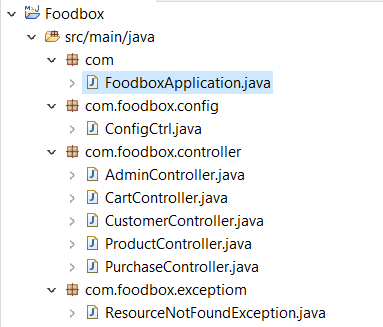


**4. How to run poject:**

4.1. clone project

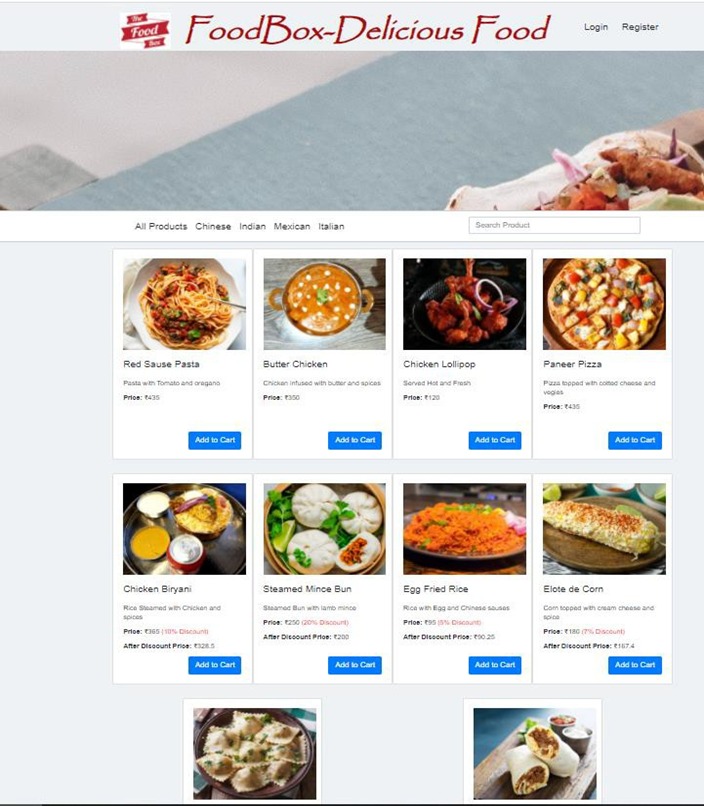
clone git : git clone https://github.com/deepak-pvsn/capstone-project-foodbox-webapp ****

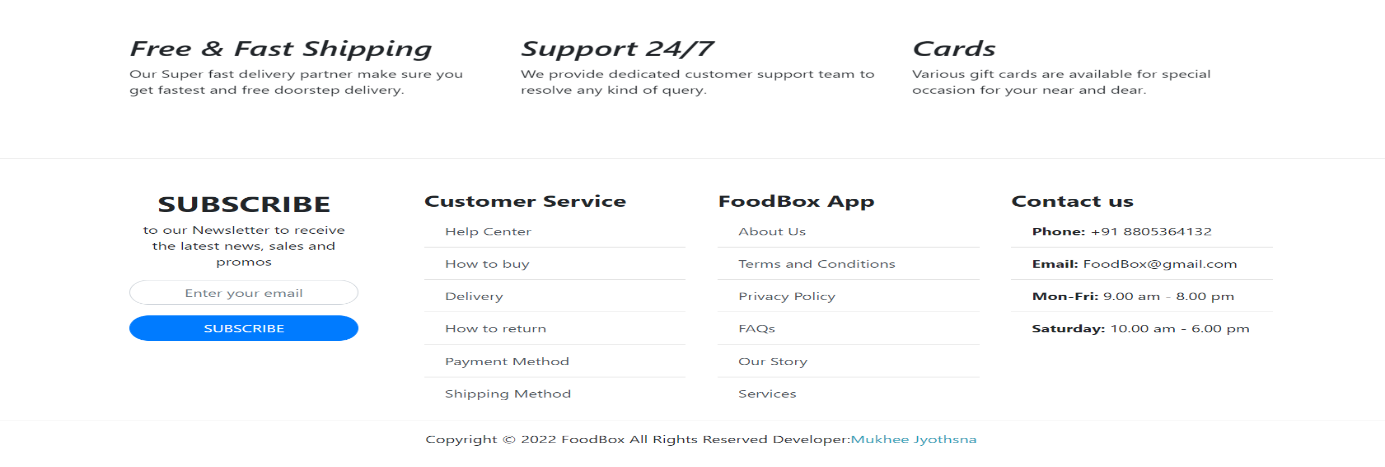
**Directory Structure / package:**

****

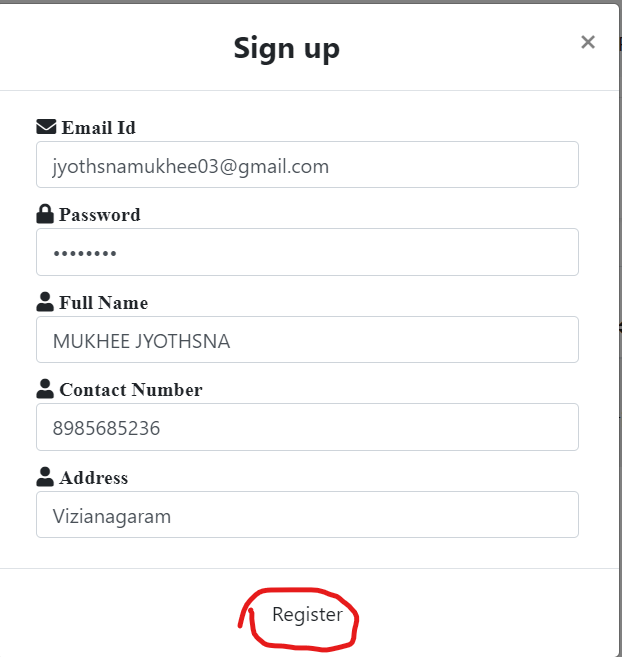
**Outputs:**

**Home Page**

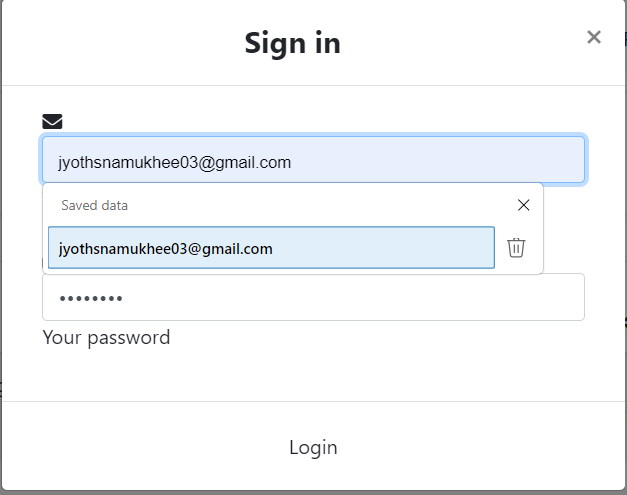




**Sign up:**



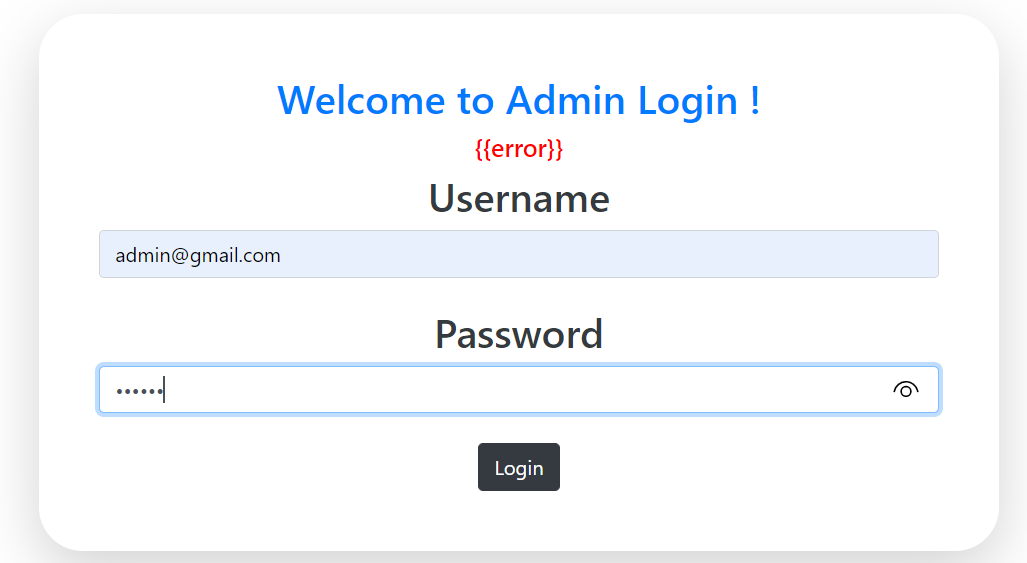
**Login:**



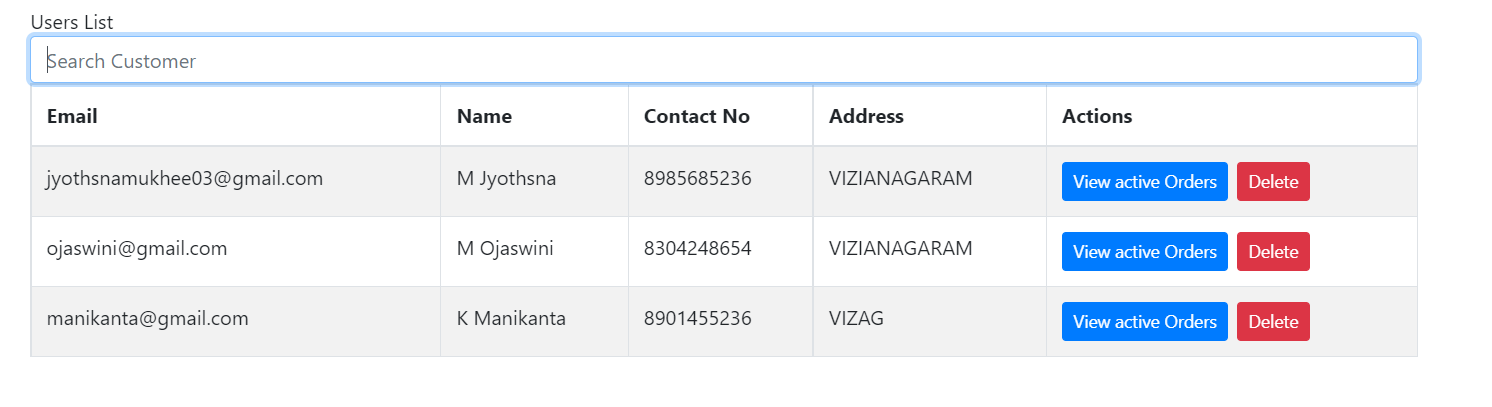
## **Login save:**



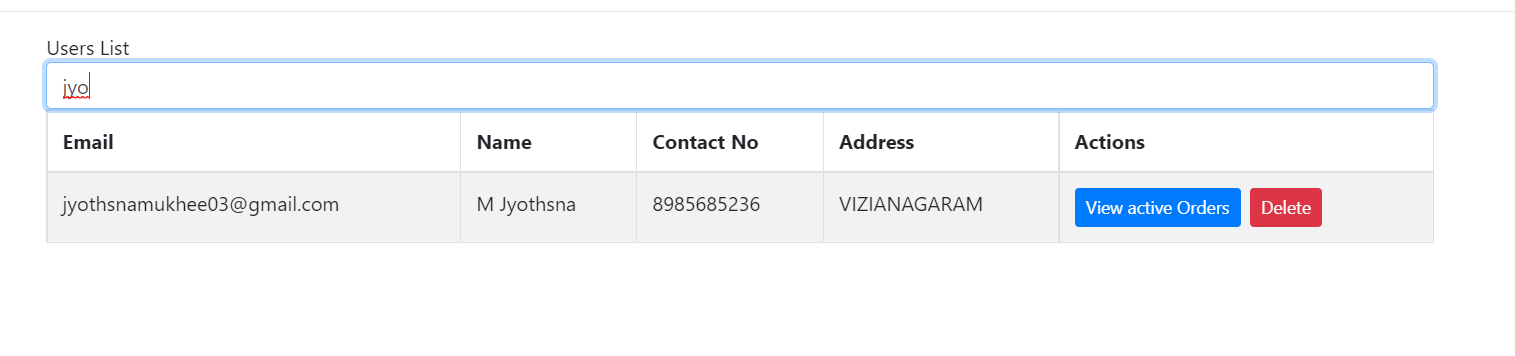
**Admin login:**



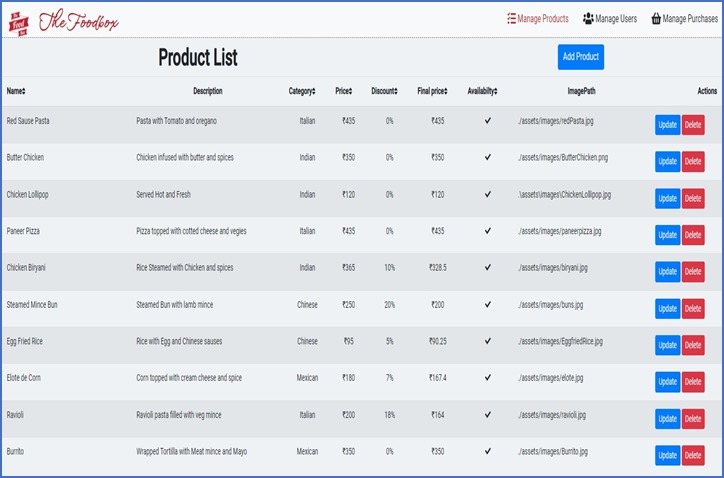
**User list:**



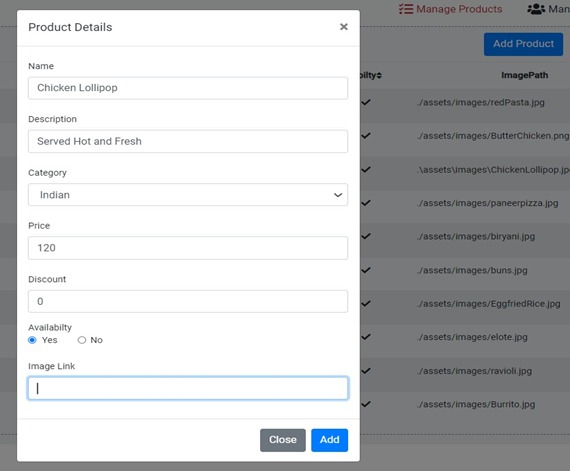
**User list search:**



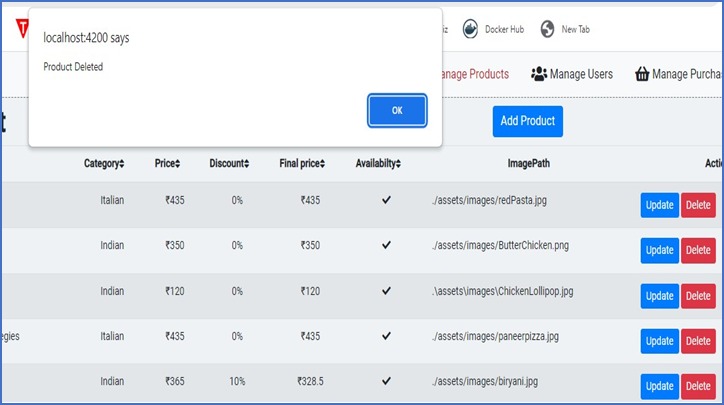
**Manage product:**



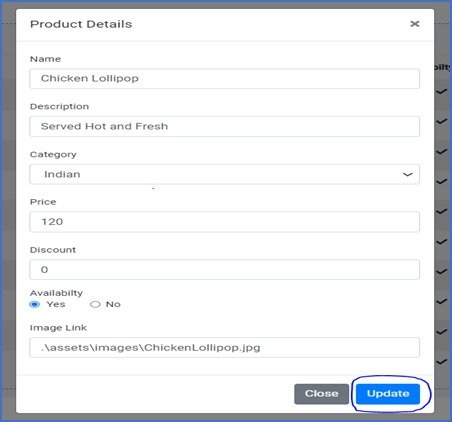
**Add Product:**



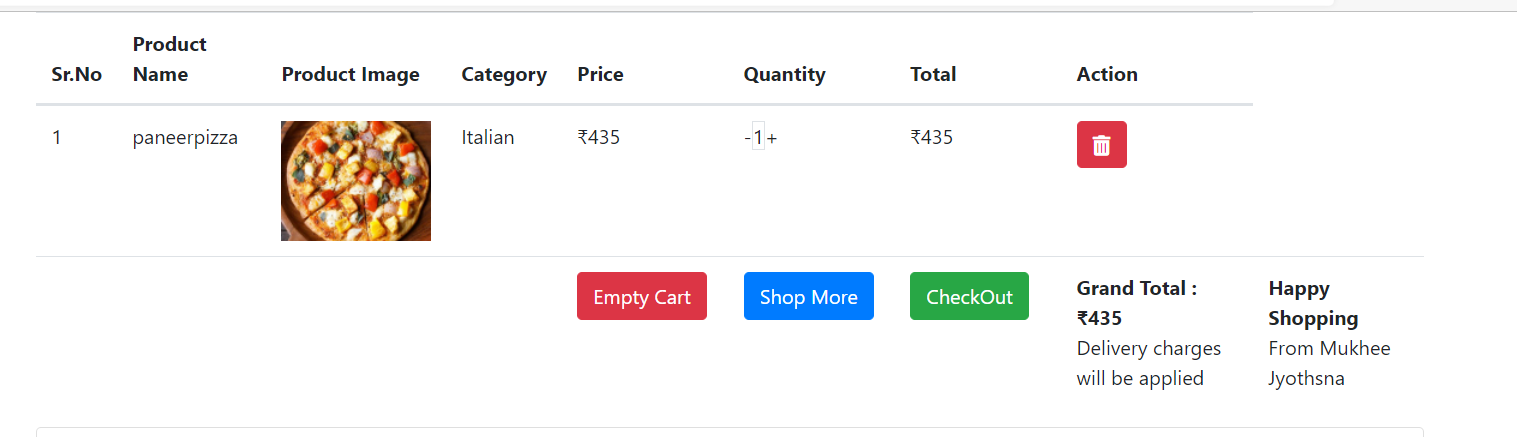
**Delete Product:**



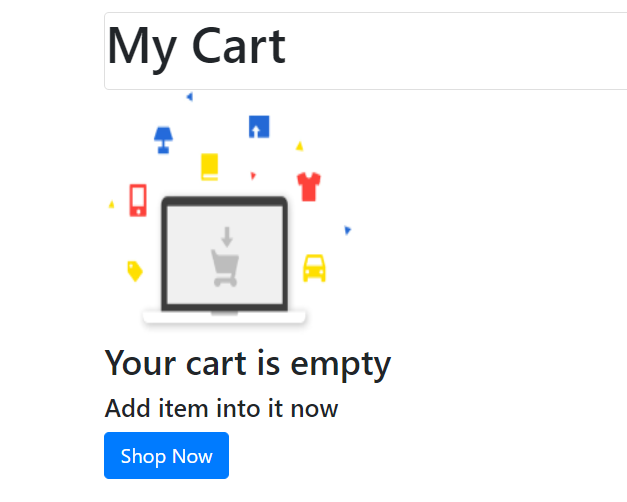
**Update Product:**



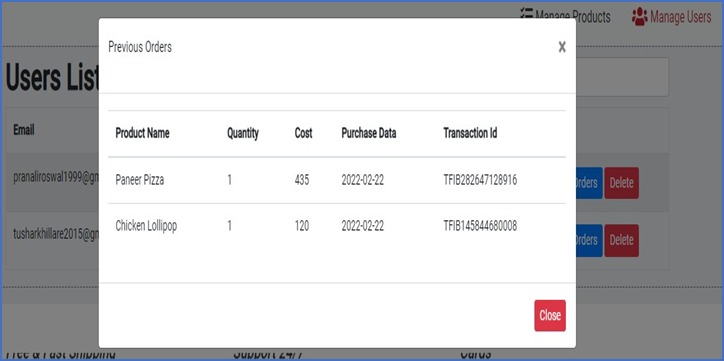
**Add cart:**



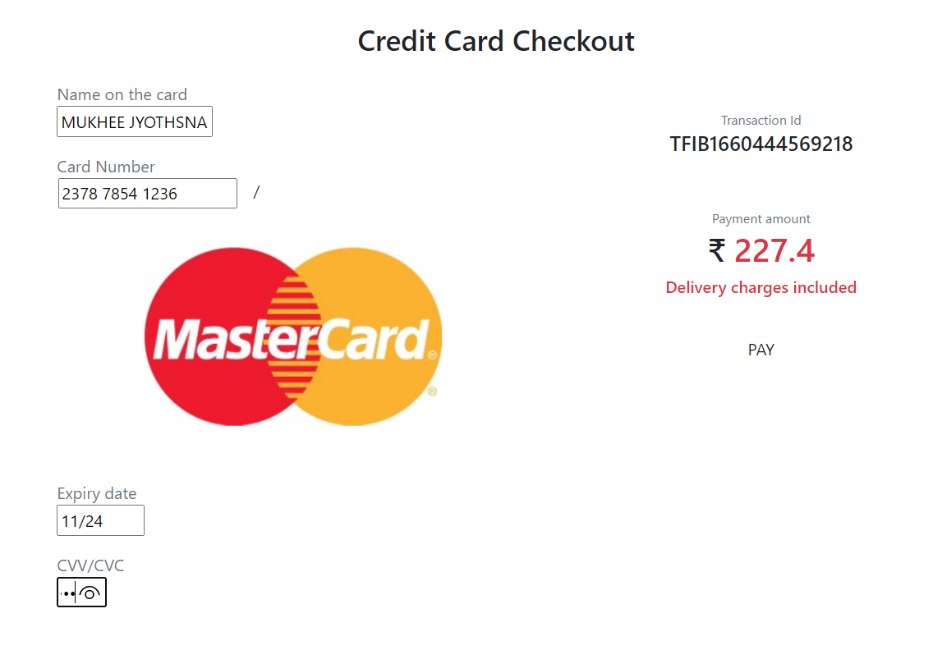
**My cart:**



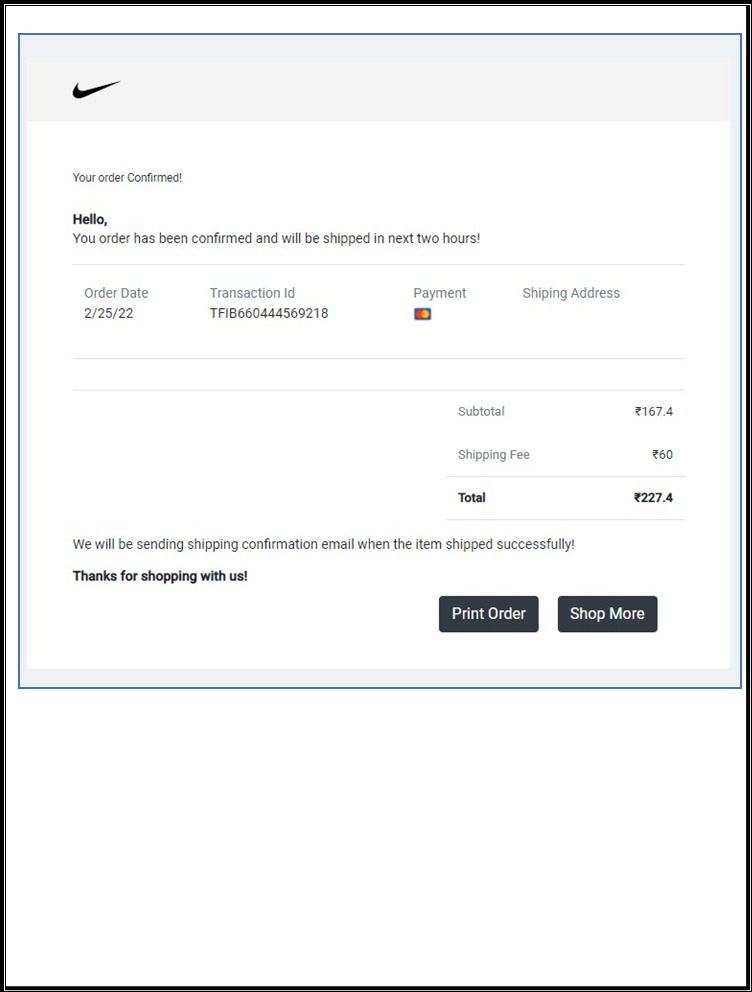
**Active orders:**



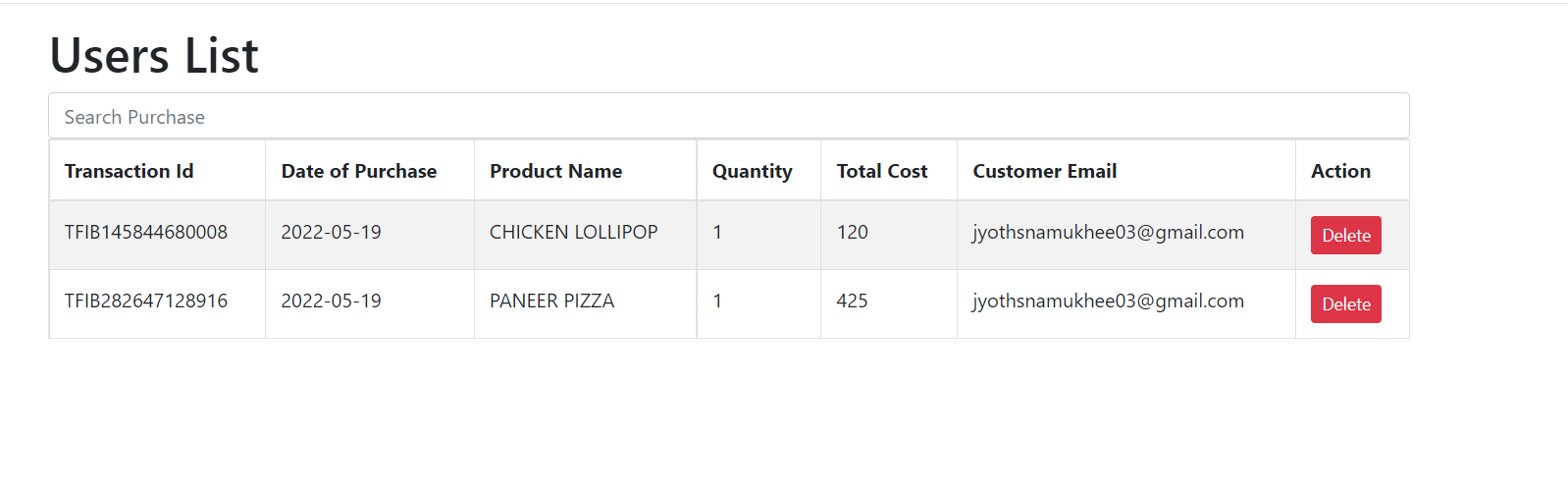
**Payment gateway:**



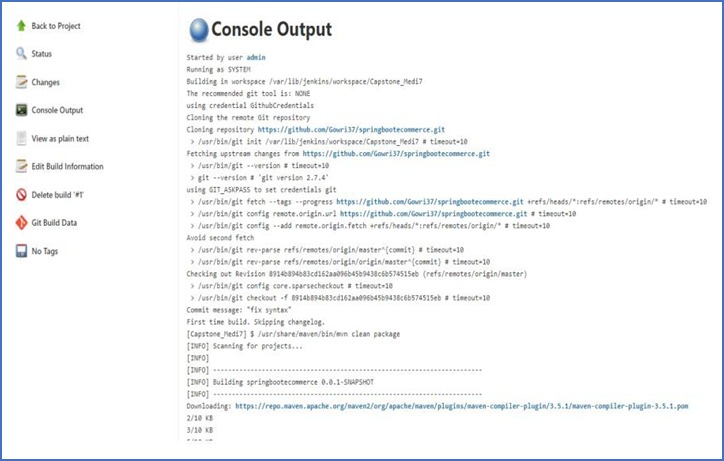
**Payment status:**



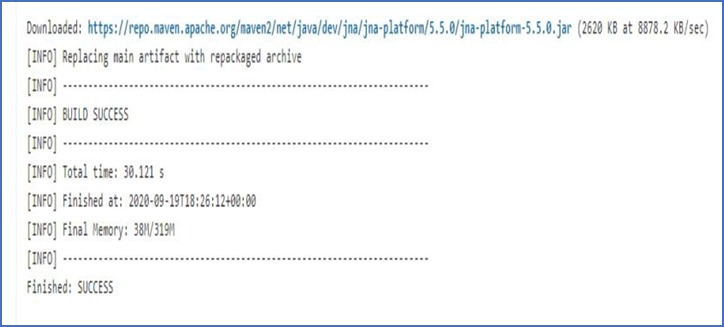
**User purchase list:**



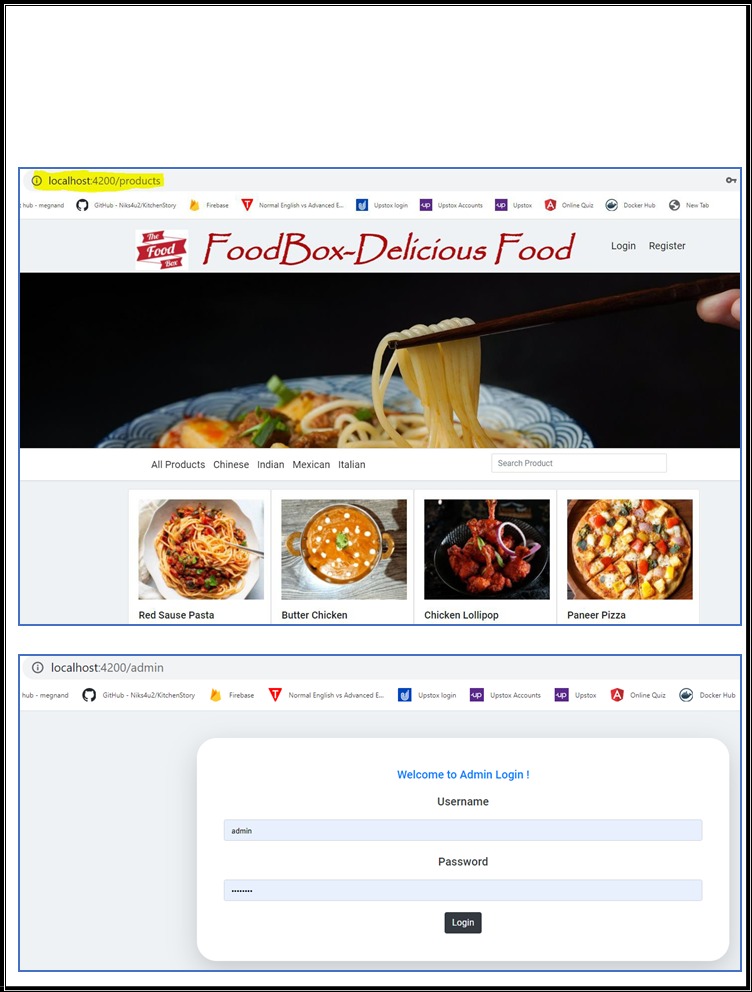
**Jenkin test:**



**Spring test:**



**Localhost:**



## **Step 6:** Pushing the code to GitHub repository

* Open your command prompt and navigate to the folder where you have created your files.

*cd <folder path>*

* Initialize repository using the following command:

*git init*

* Add all the files to your git repository using the following command:

*git add .*

* Commit the changes using the following command:

*git commit . -m <commit message>*

* Push the files to the folder you initially created using the following command:

*git push -u origin master*

## **Unique Selling Points of the Application**

1. The application is designed to keep on running and taking user inputs even after exceptions occur. To terminate the application, appropriate option needs to be selected.
2. With this application, one can easily access the food items as their own by adding cart.
3. Total price is also determined by its own.
4. Any user can signed up and no restrictions.
5. Online payment is also accepted.

## **Conclusions**

At the end of the project we had acquired some of the materials and parts, some bought and some from the food. Very easy to use the application and easy to make payment.