A  
PROJECT REPORT ON

FARM BAZAAR

**An Online Food Merchandise Store**

SUBMITTED IN PARTIAL

FULLFILLMENT OF

DIPLOMA IN ADVANCED COOMPUTING (PG-DAC)



UNDER THE GUIDANCE OF

**Mrs. Janhavi Deo**

|  |  |
| --- | --- |
| Shubham Hemraj Samarth | 230943120081 |
| Chaudhary Mayur Arvind | 230943120018 |
| Gaurav Pramod Gaikwad | 230943120029 |
| Nishant Nitin Mahajan | 230943120047 |

: PRESENTED BY :

**AT  
INFOWAY TECHNOLOGIES, PUNE**

ACKNOWLEDGEMENT

The project “FRAM BAZAAR” was a great learning experience for us and we are submitting this work to Advanced Computing Training School (Infoway Technologies, Pune).

We are very glad to mention the name of Mrs. Janhavi Deo for her valuable guidance to work on this project.

We are highly grateful to Mrs. Ulka Joshi., Course Co-ordinator of ACTS Training Centre, Infoway, for her guidance and support whenever necessary during the course of our journey to acquire PG- Diploma in Advanced Computing (PG-DAC) through Infoway Technologies, Pune.

We would like to express our sincere gratitude towards Mrs. Mayuri Fakirpure, our faculty for J2SE and J2EE, who was always there for us. Her guidance and support throughout the course helped us to overcome various obstacles and intricacies during the course of our project work. Without her tremendous support, guidance, and efforts, this project would not have been possible.

**:FROM:**

|  |  |
| --- | --- |
| Shubham Hemraj Samarth | 230943120081 |
| Chaudhary Mayur Arvind | 230943120018 |
| Gaurav Pramod Gaikwad | 230943120029 |
| Nishant Nitin Mahajan | 230943120047 |

Table of Contents

[ABSTRACT 4](#_Toc159616354)

[1 .INTRODUCTION 5](#_Toc159616355)

[2. PRODUCT OVERVIEW AND SUMMARY 6](#_Toc159616356)

[2.1. PURPOSE 6](#_Toc159616357)

[2.2. SCOPE 6](#_Toc159616358)

[2.3. OVERVIEW 6](#_Toc159616359)

[A.TECHNOLOGIES USED 6](#_Toc159616360)

[B. FEATURES PROVIDED 7](#_Toc159616361)

[2.4. FEASIBILITY STUDY 8](#_Toc159616362)

[A.TECHNICAL FEASIBILITY: 8](#_Toc159616363)

[B. OPERATIONAL FEASIBILITY 9](#_Toc159616364)

[C. ECONOMICAL FEASIBILITY 9](#_Toc159616365)

[3.REQUIREMENTS FULFILLED 10](#_Toc159616366)

[3.1. FUNCTIONAL REQUIREMENTS 10](#_Toc159616367)

[3.2. NON-FUNCTIONAL REQUIREMENTS 10](#_Toc159616368)

[4.PROJECT DESIGN 12](#_Toc159616369)

[4.1. DATA MODEL 12](#_Toc159616370)

[A. Tables Related to User Details 12](#_Toc159616371)

[B. Tables Related to Orders 12](#_Toc159616372)

[4.2. FUNCTIONAL DECOMPOSITION DIAGRAM 13](#_Toc159616373)

[4.3. USE CASE DIAGRAM 13](#_Toc159616374)

[4.4. ACTIVITY DIAGRAM 13](#_Toc159616375)

[4.6. ER DIAGRAM 13](#_Toc159616376)

[5.PROJECT SCREENSHOTS 14](#_Toc159616377)

[6.TESTING 15](#_Toc159616378)

[A. ADMIN FEATURES TEST 15](#_Toc159616379)

[B. CUSTOMER FEATURES TEST 16](#_Toc159616380)

[C. FARMER FEATURES TEST 16](#_Toc159616381)

[D. DELIVERY PARTNER FEATURES TEST 16](#_Toc159616382)

[7.CONCLUSION 17](#_Toc159616383)

[8.FUTURE SCOPE 18](#_Toc159616384)

# ABSTRACT

The Business to Consumer Model has come a long way ever since it time of inception. While it has expanded into multiple types of goods, there is still a section of market that remains untapped: Fresh goods. As the current generation of consumers is becoming more and more health conscious, and with current trends of organic food, Fresh foods can become the next big thing in e-commerce.

This project deals with developing an e-commerce website for online fresh foods product sale. It provides list of farmers that offer fresh fruits and vegetables, and products page for each farmer’s offerings. It also provides a cart for ease of remembering the choices selected by user. The user can also view their order history to go back to the farmer from whom they purchased the last batch of products.

Two main technologies were used in this project: Java and React. Java was used for backend. React is used for client side rendering of the page, which offloads the load of rendering views to the client, and provides a fluid single page experience. MySQL has been used as database to store list of users, farmers and their products.

This project has been designed and implemented in multilevel architecture so as to have minimum coupling and maximum cohesion.

# 1 .INTRODUCTION

Fresh produce industries across the world are facing a roller-coaster ride of new developments and trends. Although there might be a few tight turns and steep slopes, the latest trends paint an inspirational picture of what lies ahead in the next five to 10 years.

In the fresh produce sector, technology and retail innovations abound. From futuristic hi-tech grocery stores, the rise of e-commerce opportunities, culinary innovation centres and revolutionary robotics technology to vertical farming and plant-based food innovations like cauliflower pizza and vegetable steaks.

Online Shopping of Fresh Food opens up a new world of options. Users won’t have to go from store to store to hunt for fresh food. They won’t have to worry about wondering whether their food is organic or inorganic. They will be able to refill their fridges in just one click, all while sitting at home.

Our system offers one stop solution to all fresh food needs. Users can log into their accounts and then they will be taken to produces offered by the farmer.

Customer can pick what foods they want to order and add to the cart. Once they are done selecting what they require, after reviewing cart summary they can simply click on check out button to pay bill and they will get an order details pdf on their registered email for the same. Their cart will be delivered to their houses.

This can be done from any place, at any time all from the internet, thus making it easy to get your daily need of fresh foods.

# 2. PRODUCT OVERVIEW AND SUMMARY

## 2.1. PURPOSE

The Farm Bazaar, as the name suggests is about farmers and their showcased merchandise. It is about connecting farmers directly to the customers, thereby cutting the middle man. This ensure that customers get fresh foods at a very cheap price. This also ensure that all the farmers get a fair chance at gaining customers so that they don’t have to rely on any middle man.

## 2.2. SCOPE

“Farmers Market Place” aims to deliver a web-based application that hosts a wide collection of the food-items that users can browse through. Users can place orders and make payment. They can update their profile, add delivery address .They can view their order history as well.

Admins can manage various product details like stock, price, adding new products, and categories etc. Only admin can add farmers. Admins can even delete users and/or farmers, if the need arises.

This project does not support the actual logistics and delivery of items and actual payment logic. We are assuming that the organization that implements it will be using third-party payment API which can easily be integrated in our application if needed. Farmers Market Place is only an interface for both customers (for browsing and shopping for food items) and admins (for managing products, farmers, users listing).

## 2.3. OVERVIEW

### A.TECHNOLOGIES USED

1. FRONT END

* HTML
* CSS
* JavaScript
* React
* Axios

1. BACK END

* Spring Boot
* Spring Data JPA
* Hibernate

1. DATABASE MANAGEMENT SYSTEM

* MySQL

### B. FEATURES PROVIDED

1. **FOR ADMINS**
2. Login & Logout – Similar to customers and framers, admins can login & logout to access their account.
3. Add/Update Farmers – Only admin is responsible for adding and updating the details of farmer.
4. Delete Farmer – The admins can delete a farmer account if they need to for any purpose.
5. Add New Category – Admins can add category.
6. Delete Category – Admins can remove category.
7. Add New Products – Admin can add new product with details as stock, price, name, quantity, image, category, etc.
8. Manage Products – Admin can update the product details.
9. View Users – Admin can view all registered users.
10. Delete User – Admin can delete a user if need arises.
11. View order details – Admin can view order details for all users.
12. **FOR CUSTOMERS**
13. Browse – Customers can browse the homepage to explore the entire products available.
14. Register, Login & Logout – New customers can register on the site. Existing customers can then login to access their account information and logout when the account is not in use.
15. View Profile – When logged in, customers can view their profile.
16. Add to Cart & Place Orders – If customers finds the food item of their choice they can save the item in the cart until they decide to purchase it. If at any point they want to cancel certain item they can simply remove it from the cart on one click. When they wish to purchase it, they can place orders for those items by selecting a delivery address on their account and pay the bill.
17. View Order History – Every customer can view their order history in order to get an idea about their past spending. Also the customer will get email notification for respective order details.

## 2.4. FEASIBILITY STUDY

Feasibility is the determination of whether a project is worth undertaking or not. Before actually recommending the new system, it is important to investigate if it is feasible to develop it.

Before developing and implementing a system, we have to make sure that the system is feasible in the following ways:

### A.TECHNICAL FEASIBILITY:

In this type of feasibility study, the system analyst has to check whether it is possible or not to develop the requested system with the available manpower, software, hardware, etc.

This project makes use of cross-platform software and solutions like Java, and hence can run on any operating system. React, used in front-end, is swift and light weight framework when it comes to delivering the requested page as it doesn’t reload the entire page for every HTTP request. It only re-renders the components that need to fetch new data. Also, as React is modular in nature, it is easy to develop new components and scale up existing components in order to add new features to the system. The combination of Spring Boot, Spring Data JPA and Hibernate for backend make for a fast, easy to set-up and reliable system to interact with the database, as they are secure and transactional in nature. Since the sensitive data of customers and admins need to be stored in a robust and secure database, MySQL database management system was chosen as it is an industry standard.

### B. OPERATIONAL FEASIBILITY

In this type of feasibility study, the operation of the system is considered. An analysis is performed on whether it is feasible for the user department to use the application. Thus, the proposed system is said to be operationally feasible only if clients are able to understand the system clearly and correctly, and can use it with ease.

In the design of this project, we always kept user experience in mind. We made an effort to have a good user interface with consistent theme and alluring design to keep the users interested and engaged. In our project, the use of universally known icons and instructions that are easy to understand makes sure that the user will not need any special technical know-how to use the application. We made sure that the information available throughout the application is arranged in a logically coherent and consistent manner, guaranteeing that the users will have a smooth and effortless experience and even enjoy using the application.

### C. ECONOMICAL FEASIBILITY

In this type of feasibility study, the benefits of the system to the organization are considered by taking into consideration the cost-benefit analysis. All the software and technologies used in our project free, open-source, and widely available, with each of the technologies having an extensive community support. This makes “**Farm Bazaar**” an economically feasible solution to the organizations that wish to implement it.

# 3.REQUIREMENTS FULFILLED

## 3.1. FUNCTIONAL REQUIREMENTS

Following are the functional requirements fulfilled by our project:

* Similar to customers, admins can login & logout to access their account.
* Only admin is responsible for adding and updating the details of farmer.
* The admins can delete a farmer account if they need to, for any purpose.
* Admins can add and remove category.
* Admin can add new product with details as stock, price, name, quantity, image, category and update and remove them.
* Admin can view all registered users, delete a user if need arises
* Admin can view order details for all users.
* Customers can browse the homepage to explore the entire products available.
* When logged in, customers can view their profile and update their details.
* If customers finds the food item of their choice they can save the item in the cart until they decide to purchase it. If at any point they want to cancel certain item they can simply remove it from the cart on one click. When they wish to purchase it, they can place orders for those items by selecting a delivery address on their account and pay the bill.
* Every customer can view their order history in order to get an idea about their past spending. Also the customer will get email notification for respective order details.

## 3.2. NON-FUNCTIONAL REQUIREMENTS

Following are the non-functional requirements fulfilled by our project:

 Since the application uses lightweight and established software components that are also cross-platform, it is remarkably performant and has good support for every operating system.

 The use of React for front end and Spring Boot, Spring Data JPA and Hibernate for back end delivers quick response times to admins and customers alike.

 Card-style UI and well-known icons and symbols used throughout the application provides a consistent theme and user-friendly interface that anyone can grasp easily, even without a technical background.

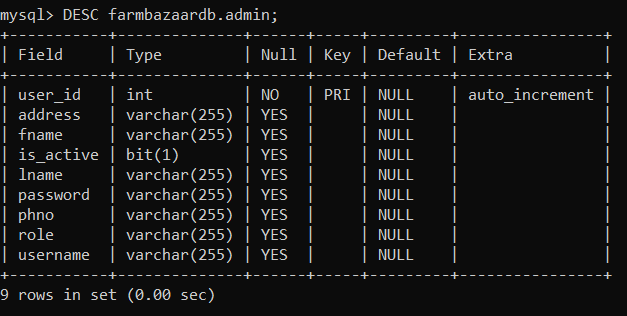
# 4.PROJECT DESIGN

## 4.1. DATA MODEL

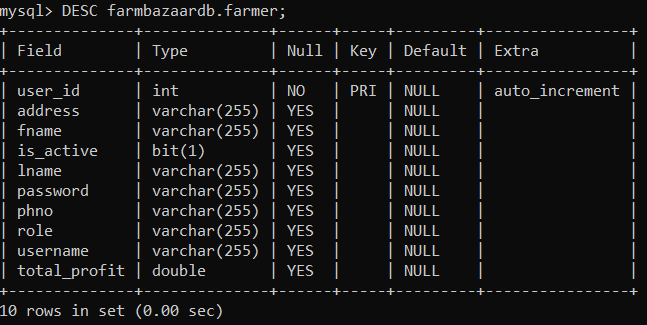
The following tables depict the database design used for “Wordsworth” application:

### Tables Related to User Details

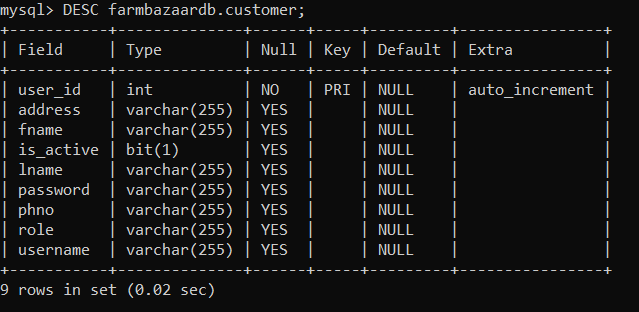
1. ADMIN Table:



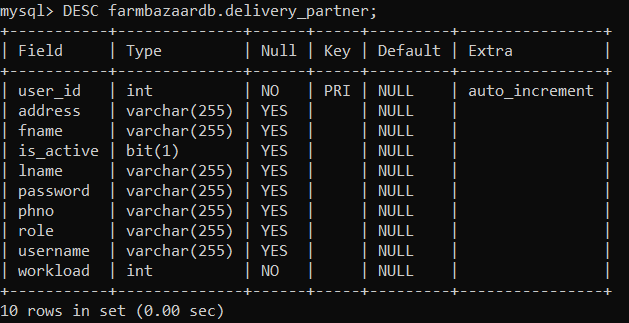
1. FARMER Table



1. CUSTOMER Table

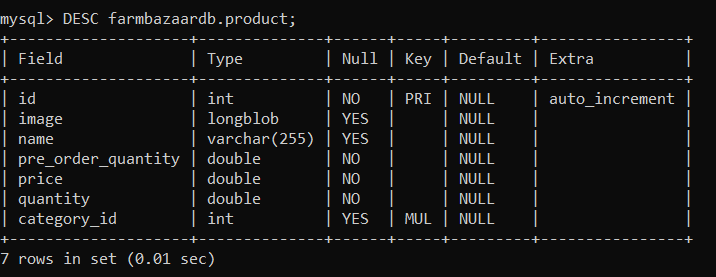


1. DELIVERY PARTNER Table

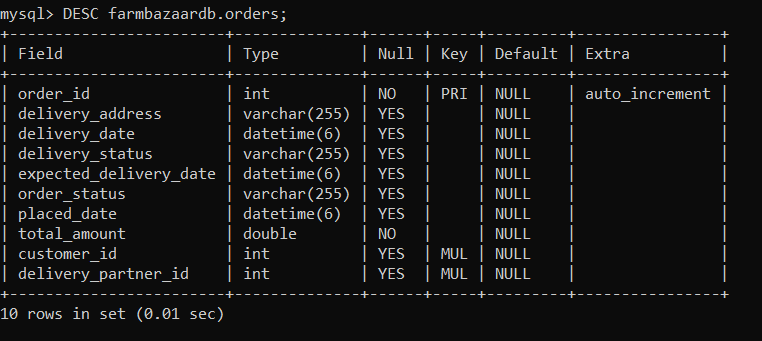


### Tables Related to Orders

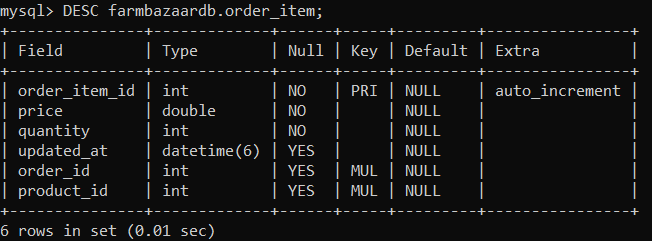
1. Product Details Table



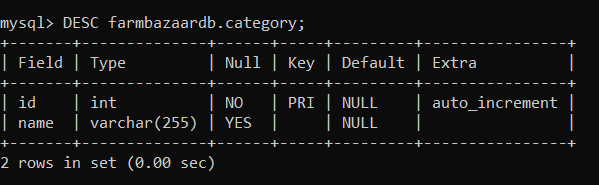
1. Order Table



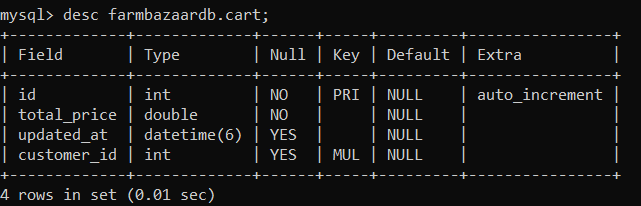
1. Order Details Table



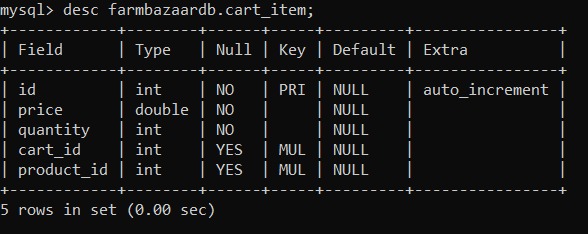
1. Category Table



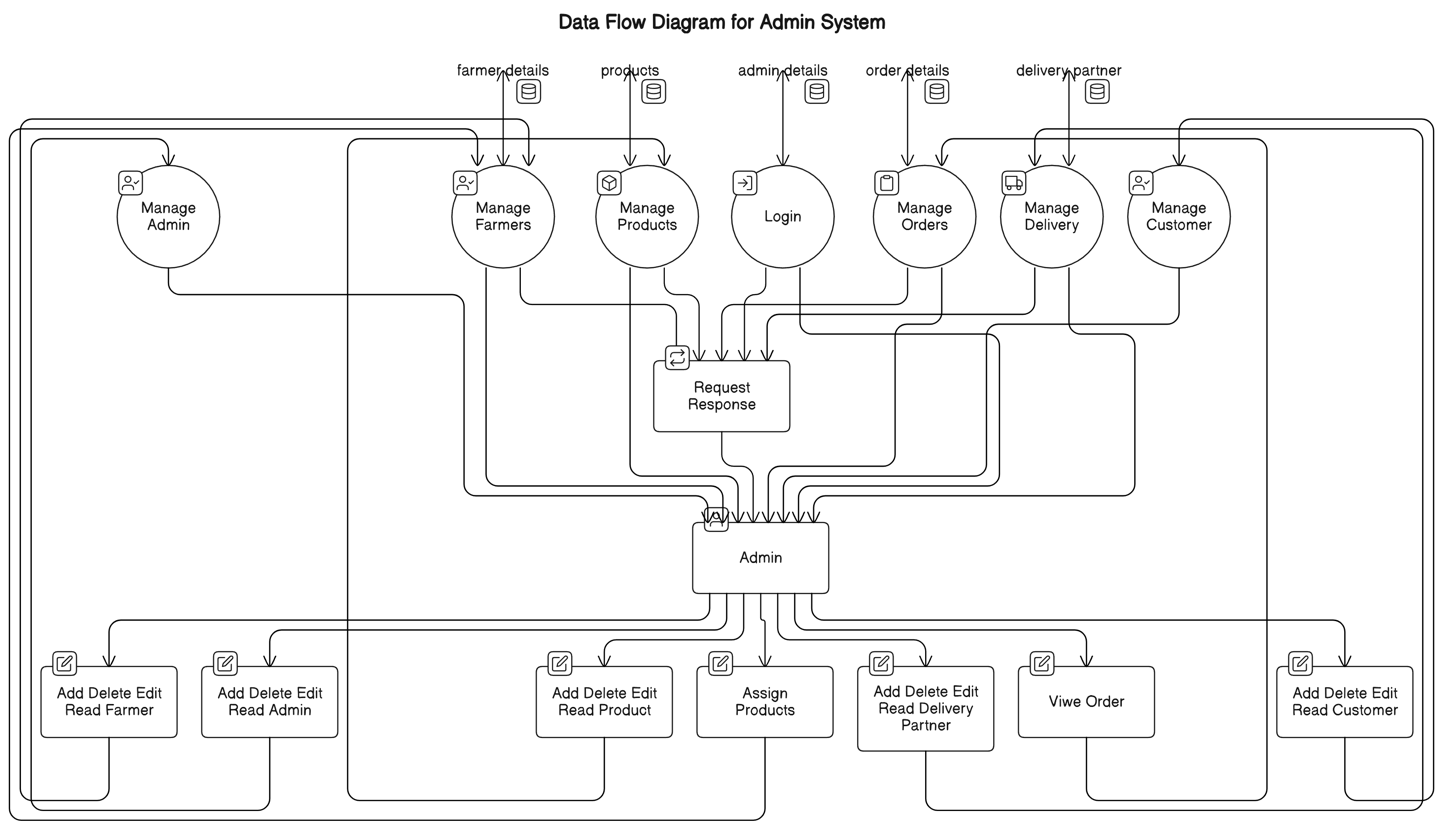
1. Cart

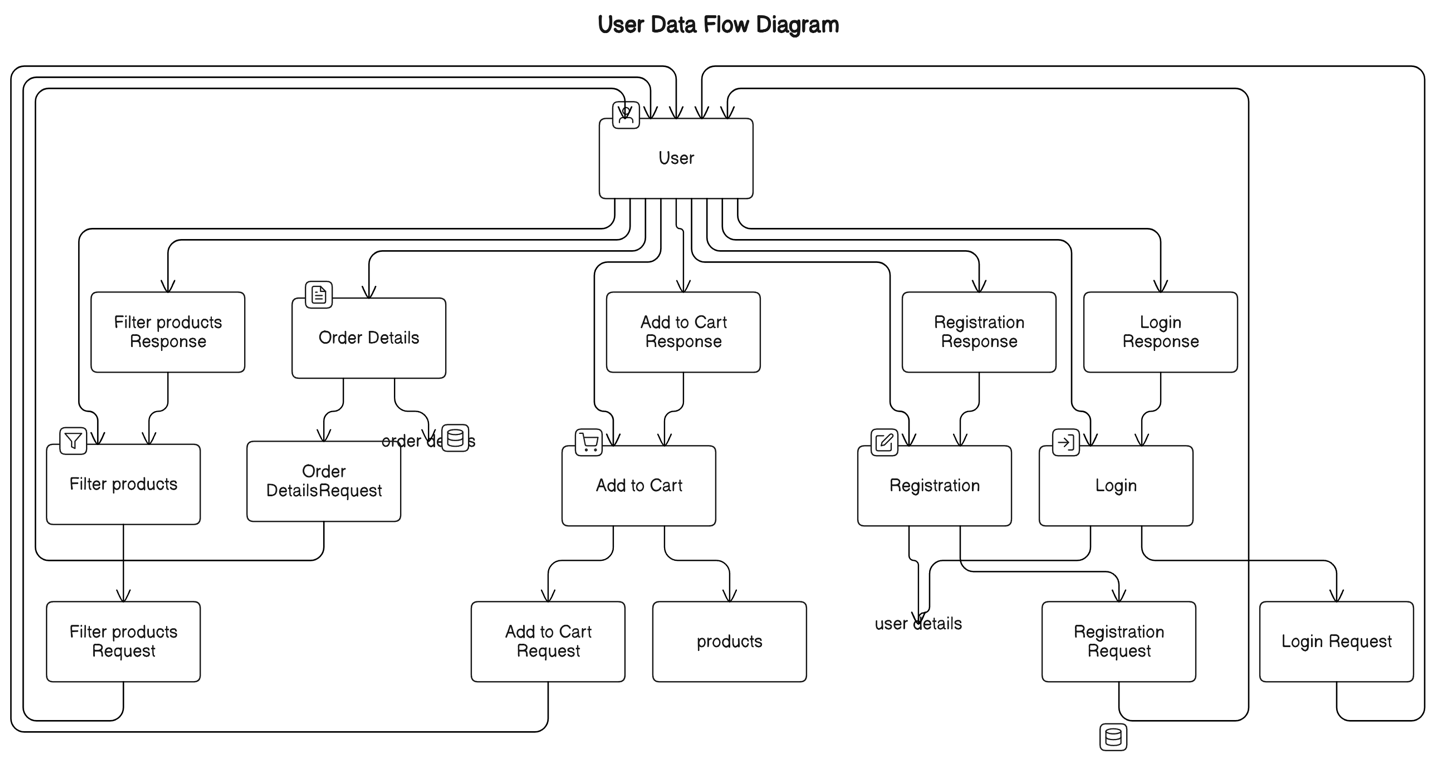


1. Cart Items:

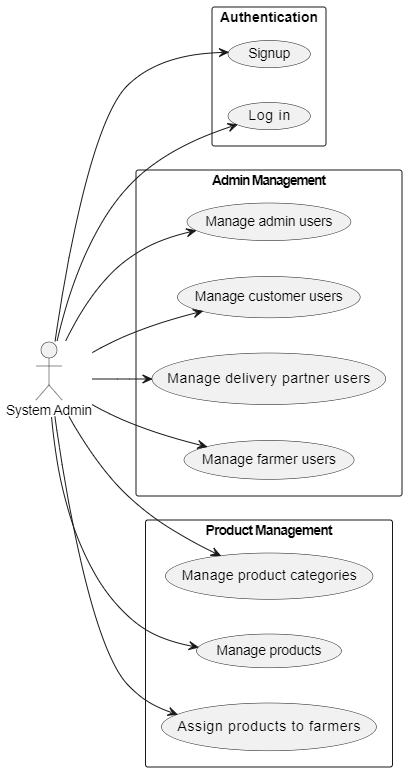


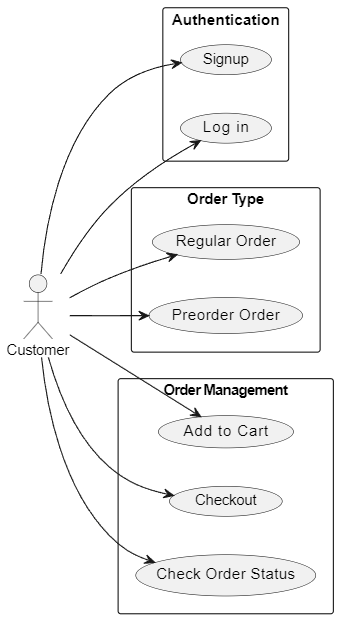
## 4.2. FUNCTIONAL DECOMPOSITION DIAGRAM

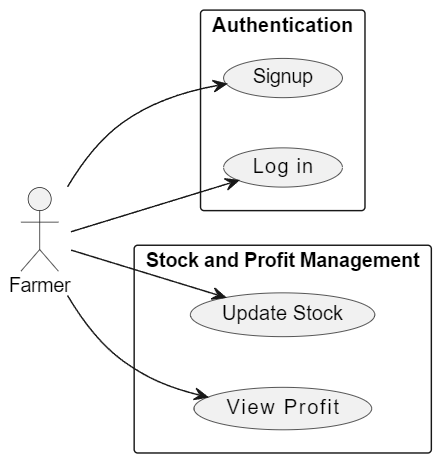


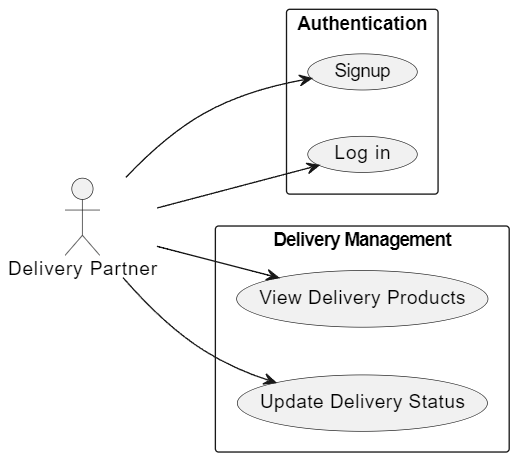


## 4.3. USE CASE DIAGRAM







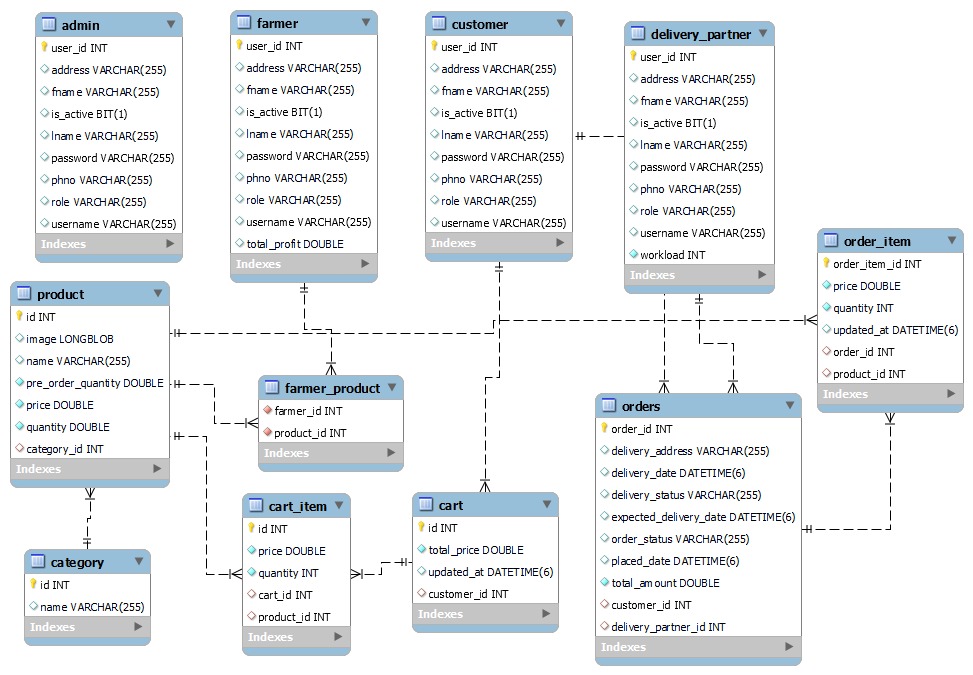


## 4.4. SYSTEM ARCHITECTURE DIAGRAM



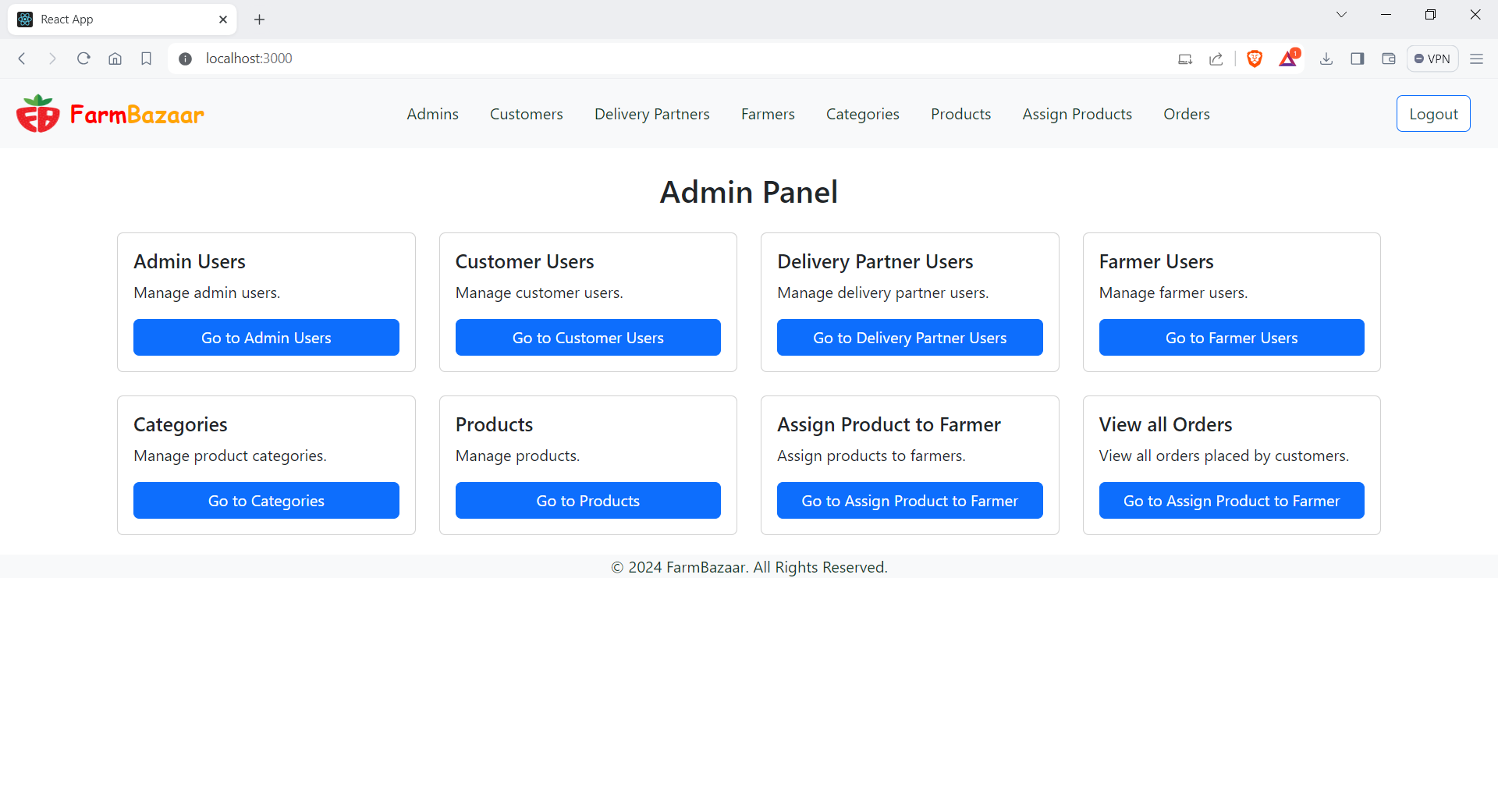
## 

## 4.6. ER DIAGRAM

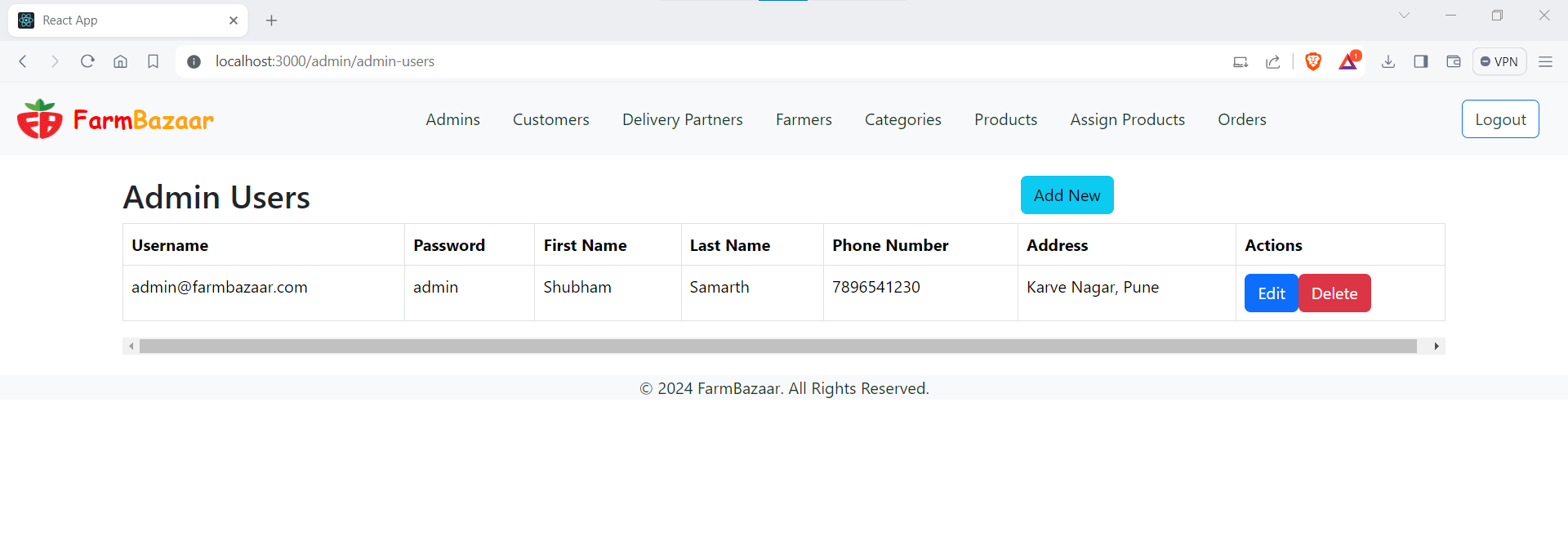


# 5.PROJECT SCREENSHOTS

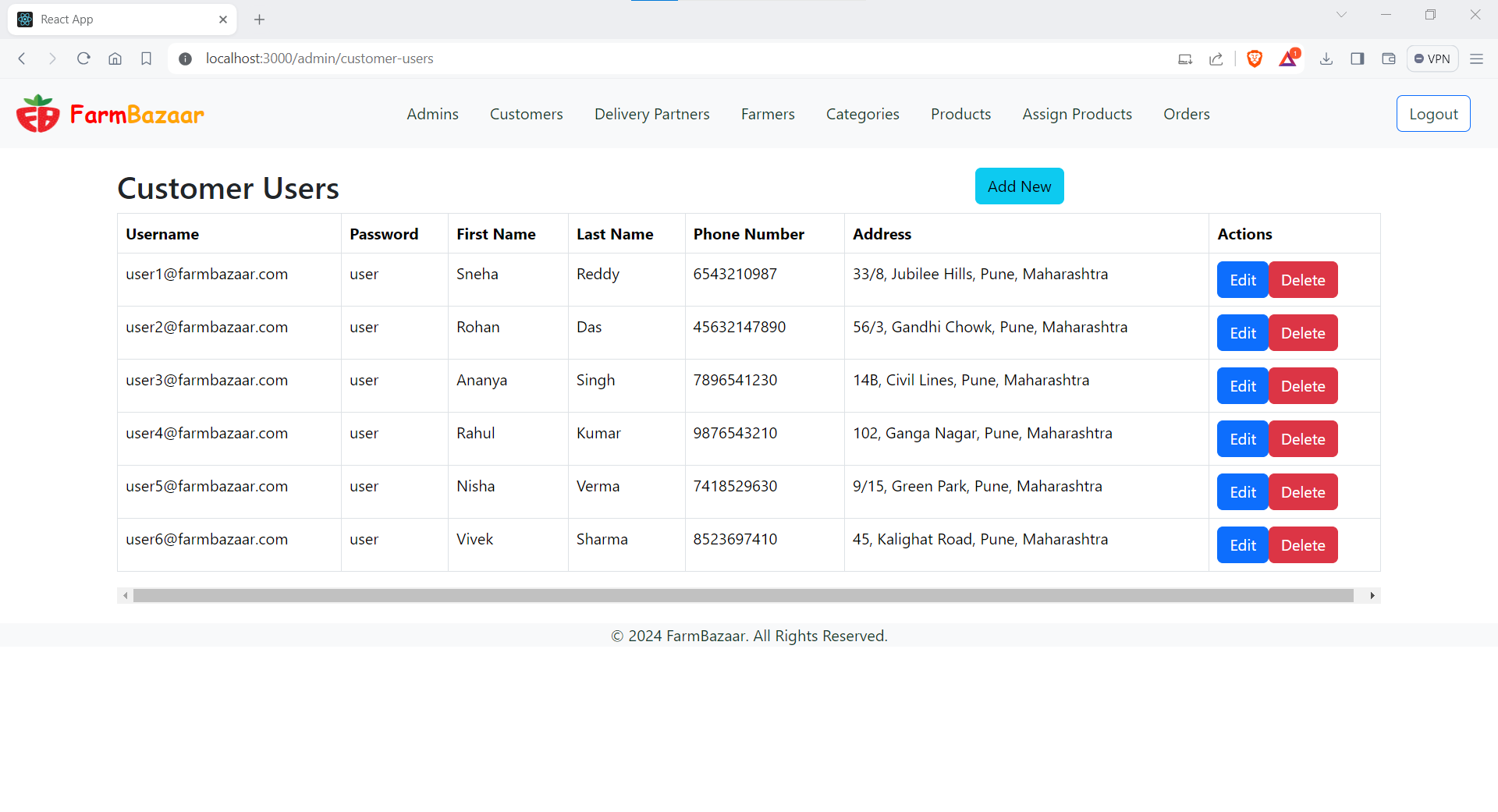
Admin Homepage



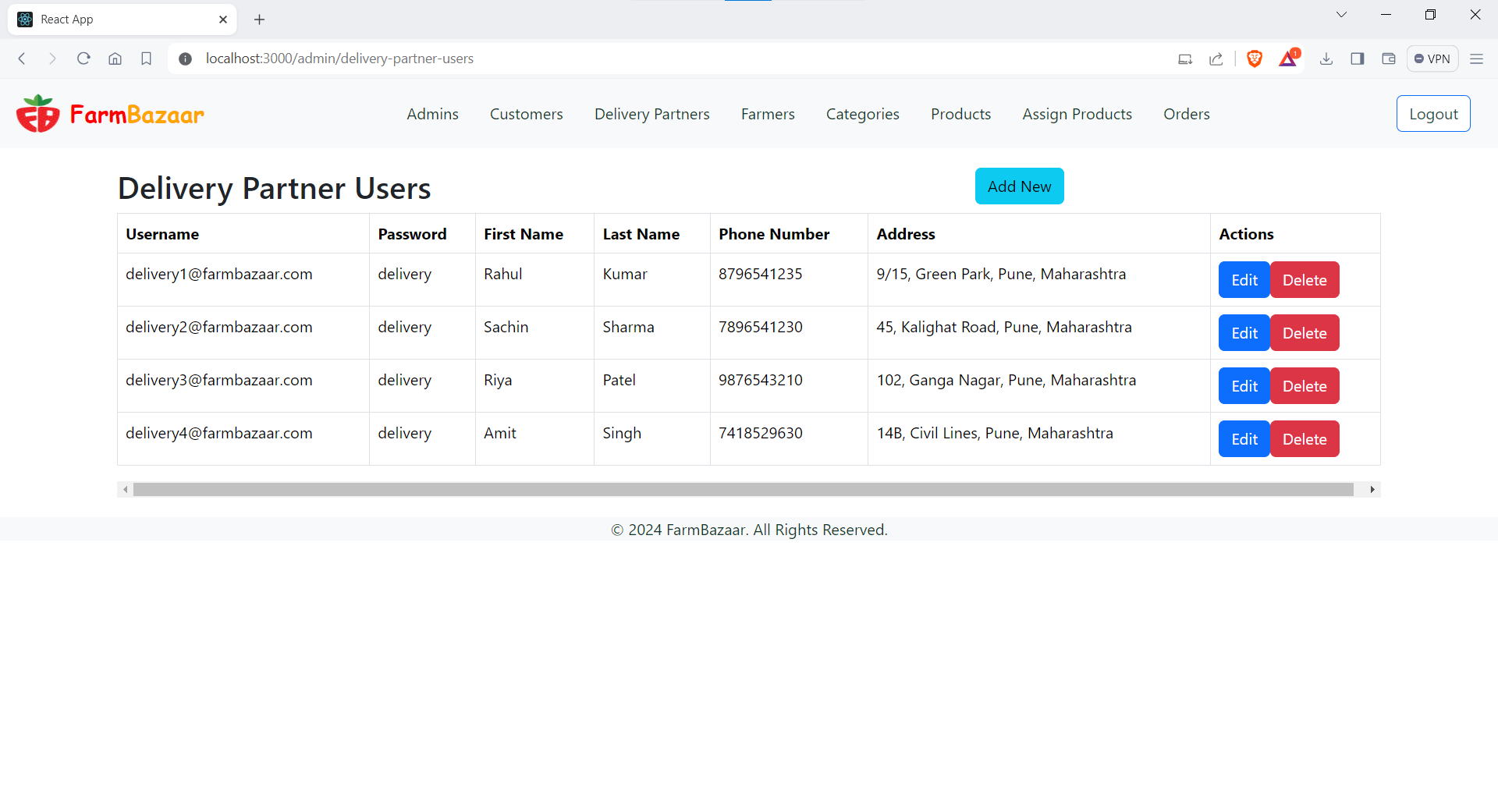
Admin Users (CRUD)



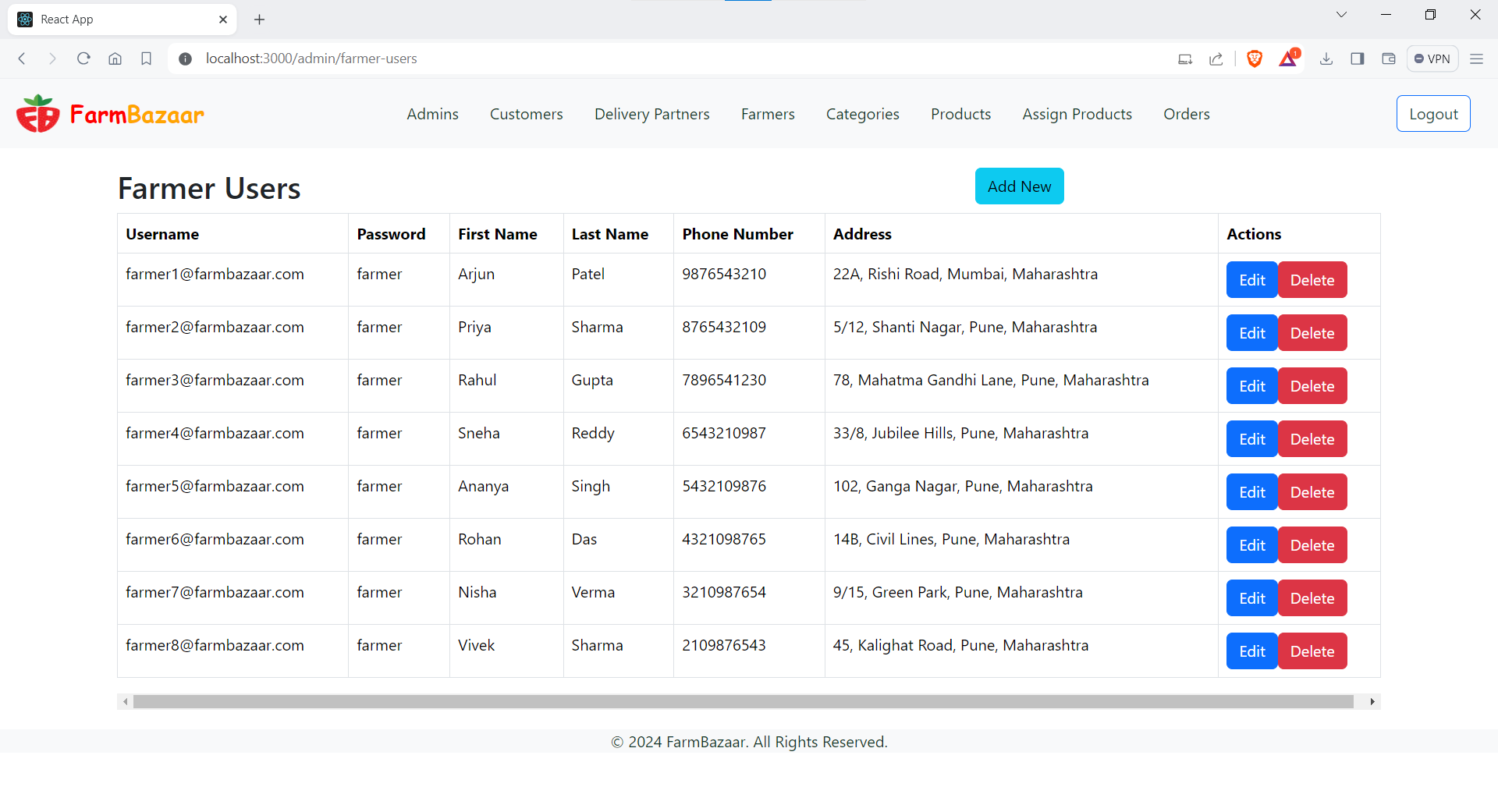
Customer Users (CRUD)



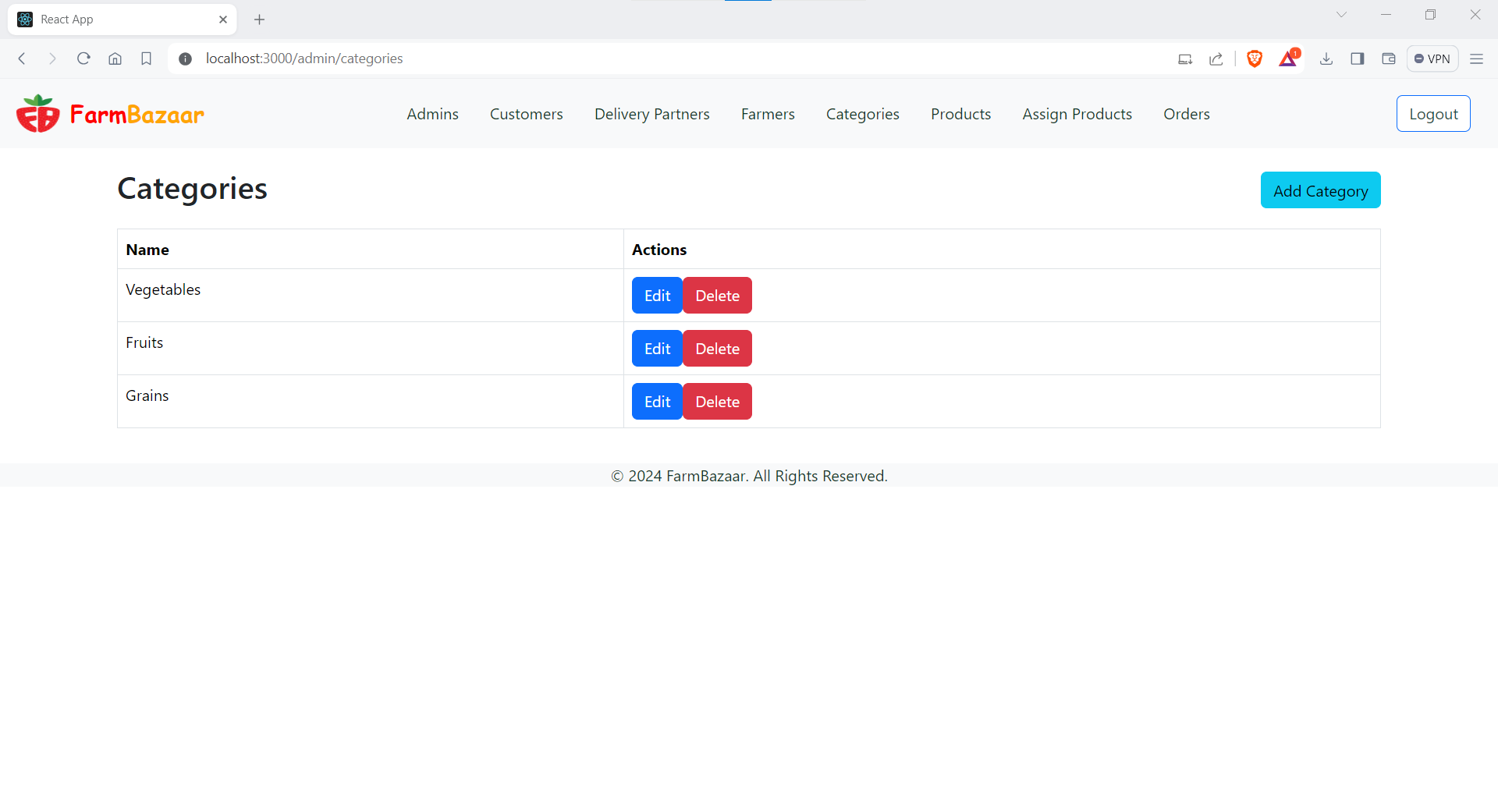
Delivery Partners Users (CRUD)



Farmer Users (CRUD)



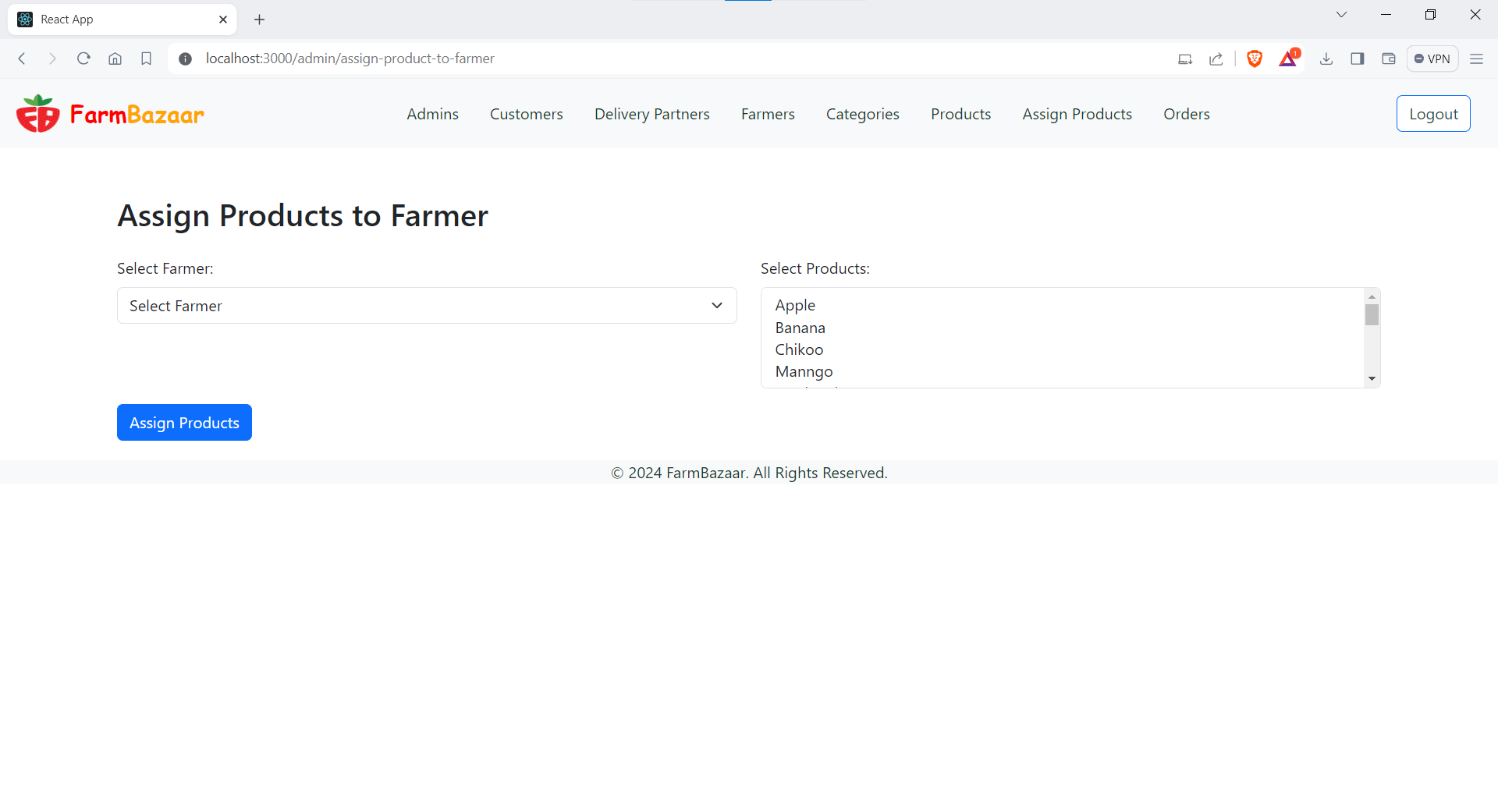
Categories (CRUD)



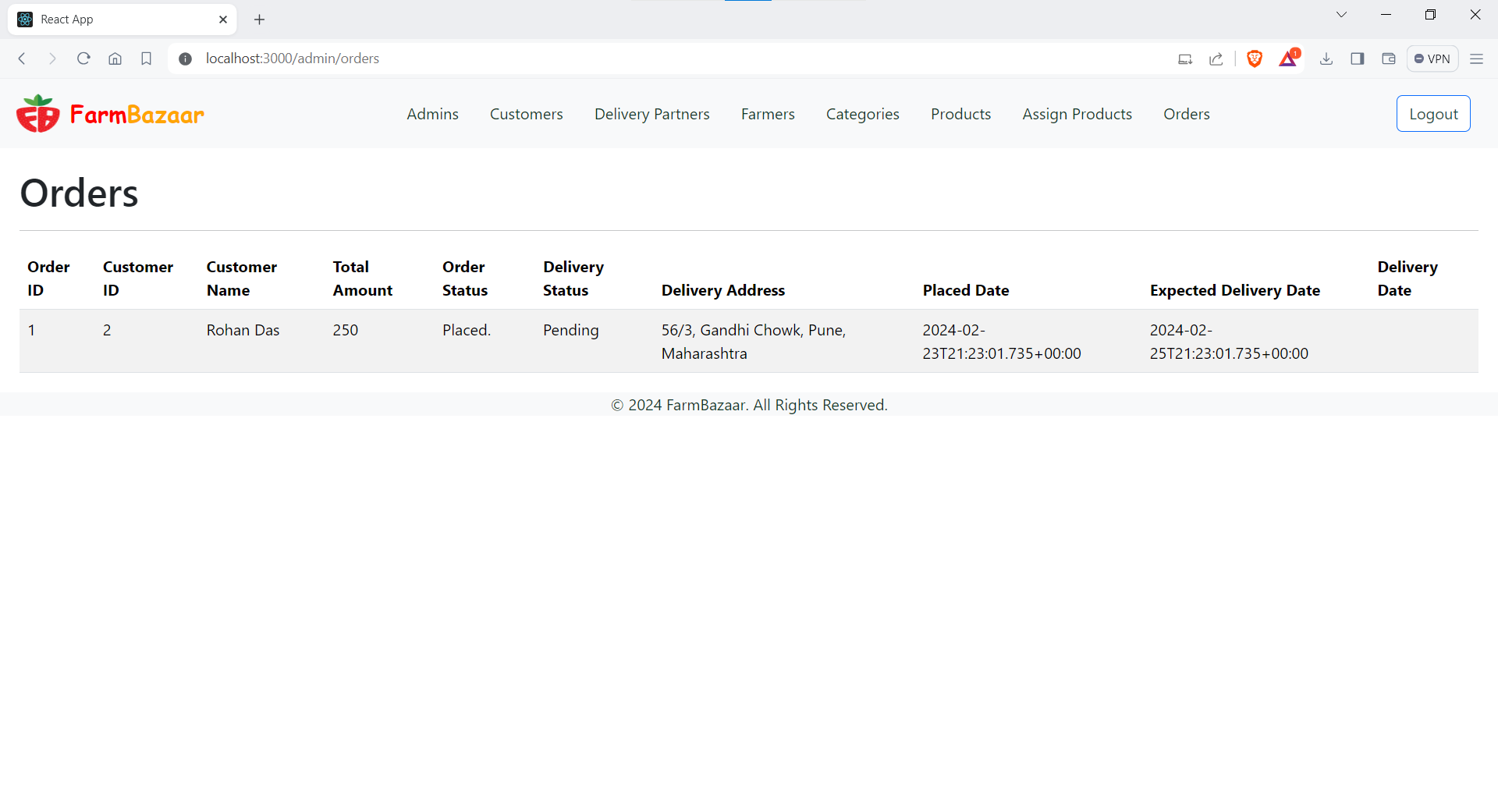
Products



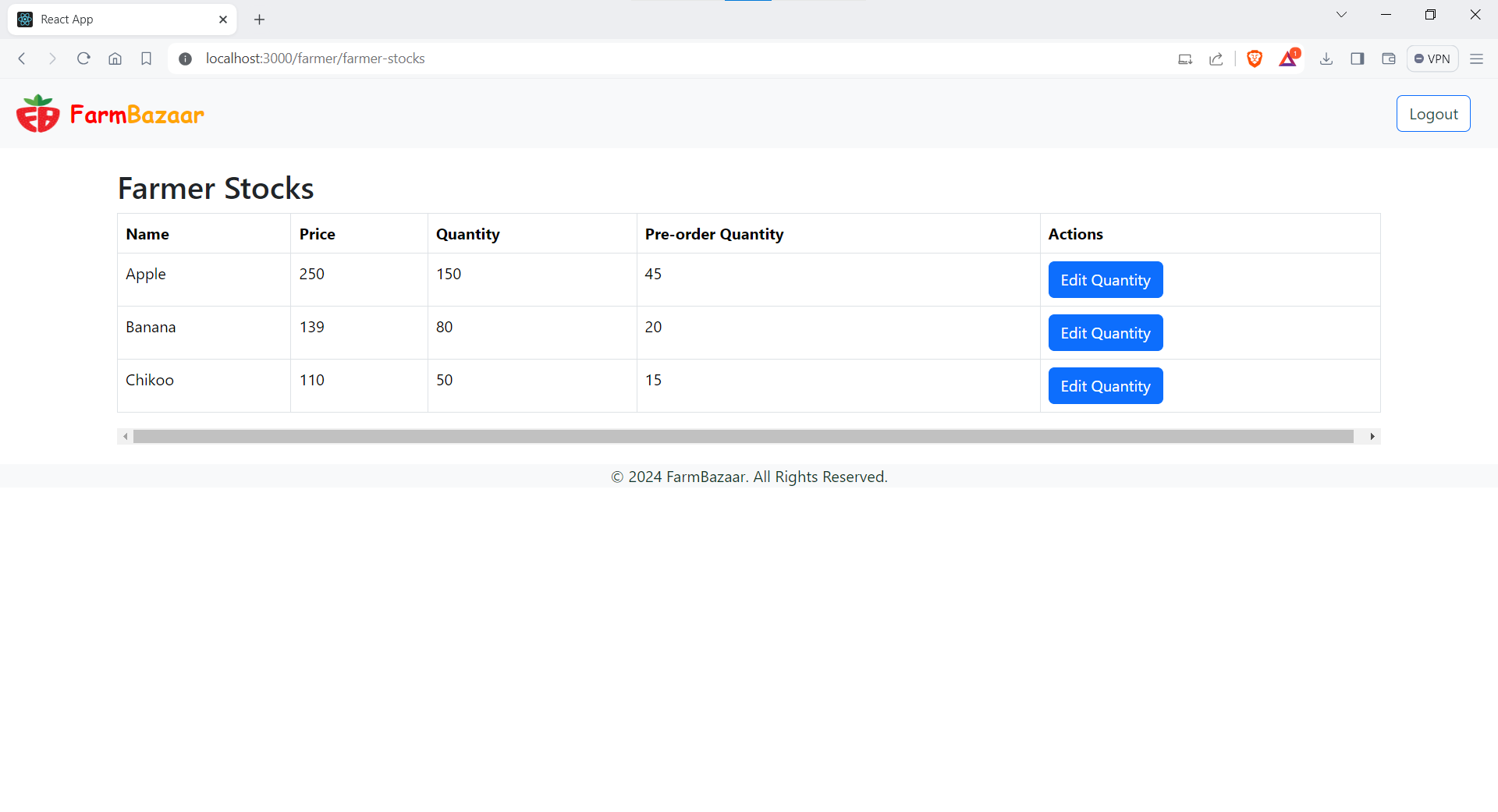
Assign Products to Farmer:

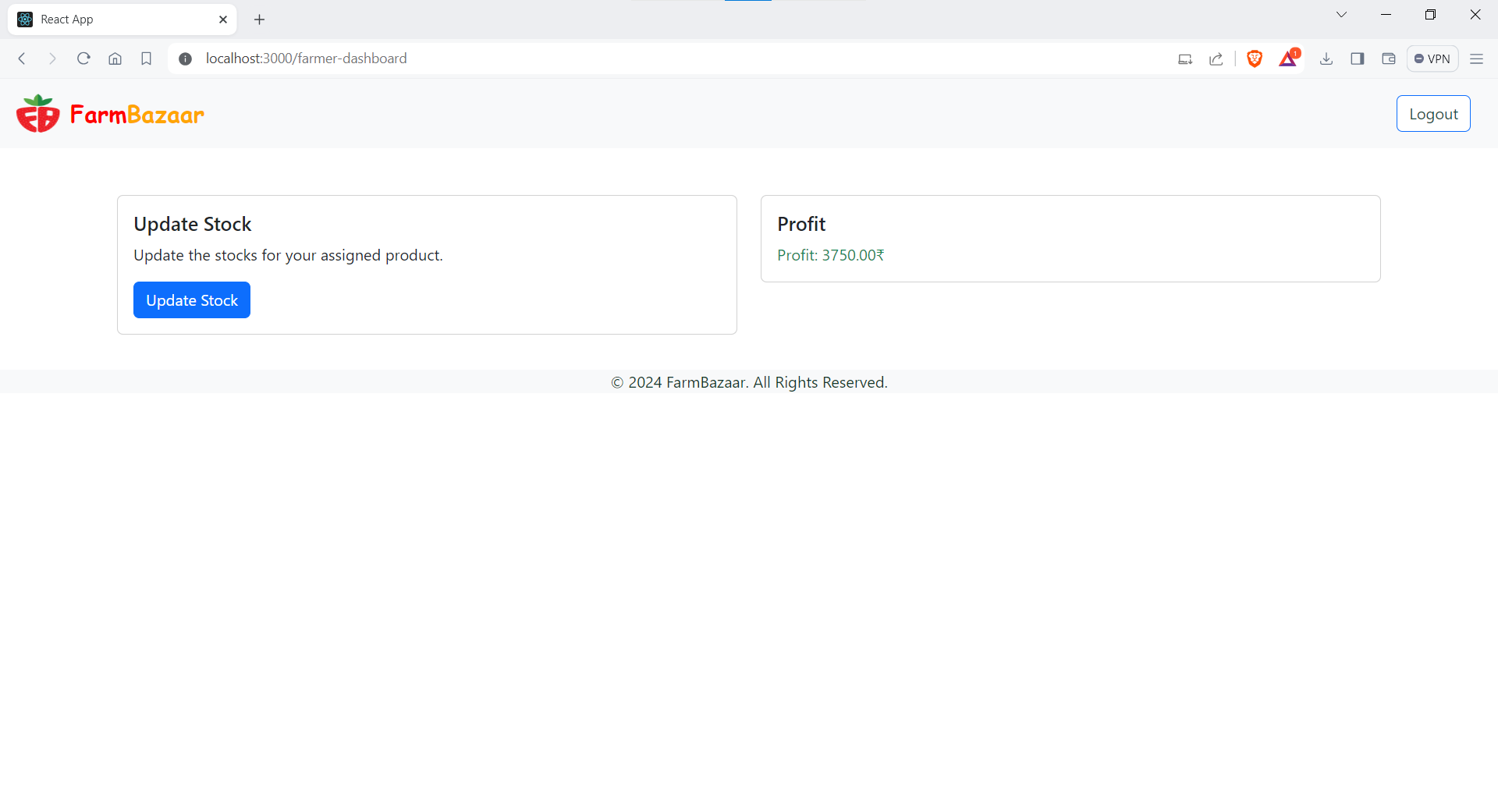


Orders:

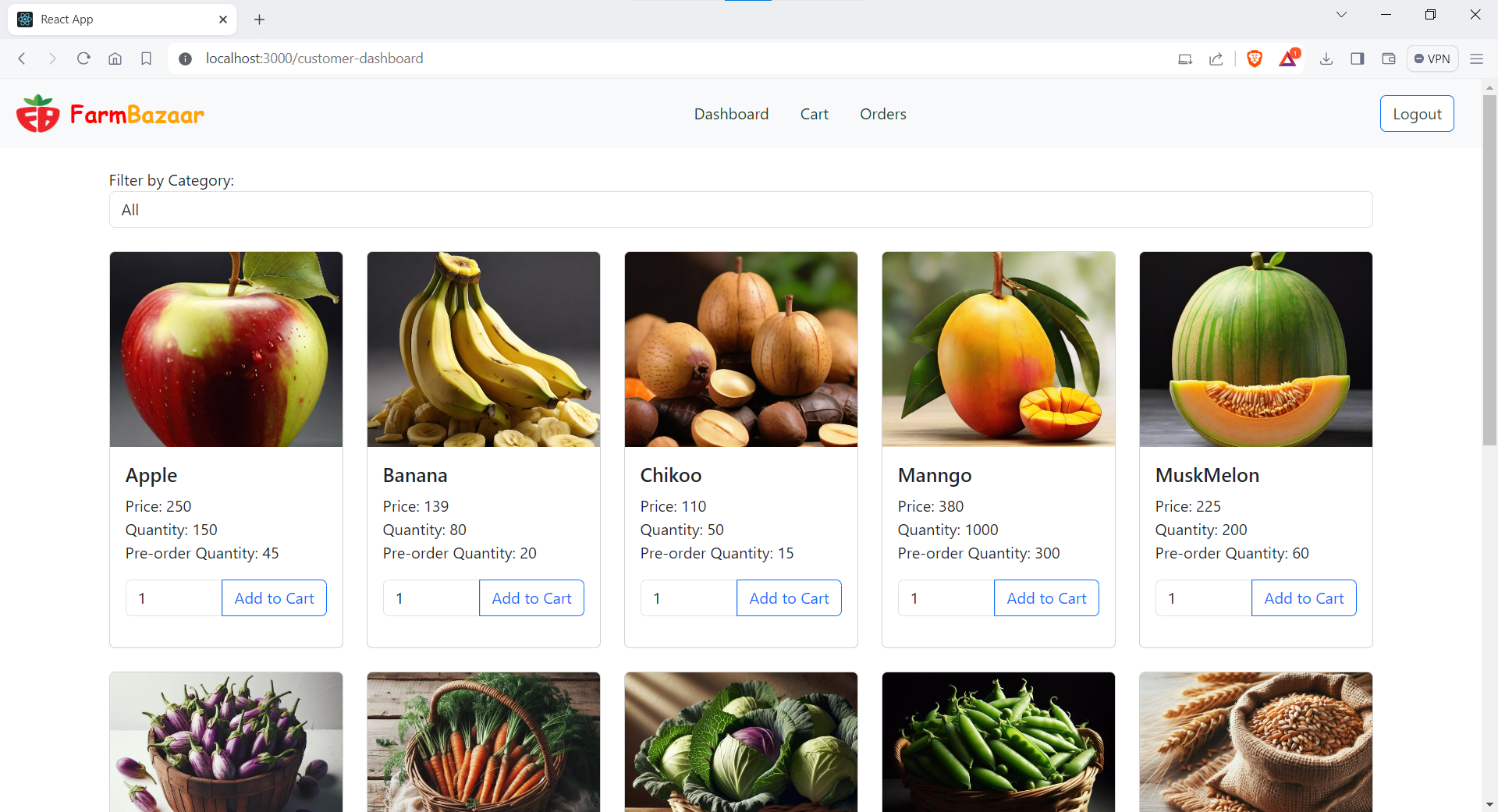


Farmer:

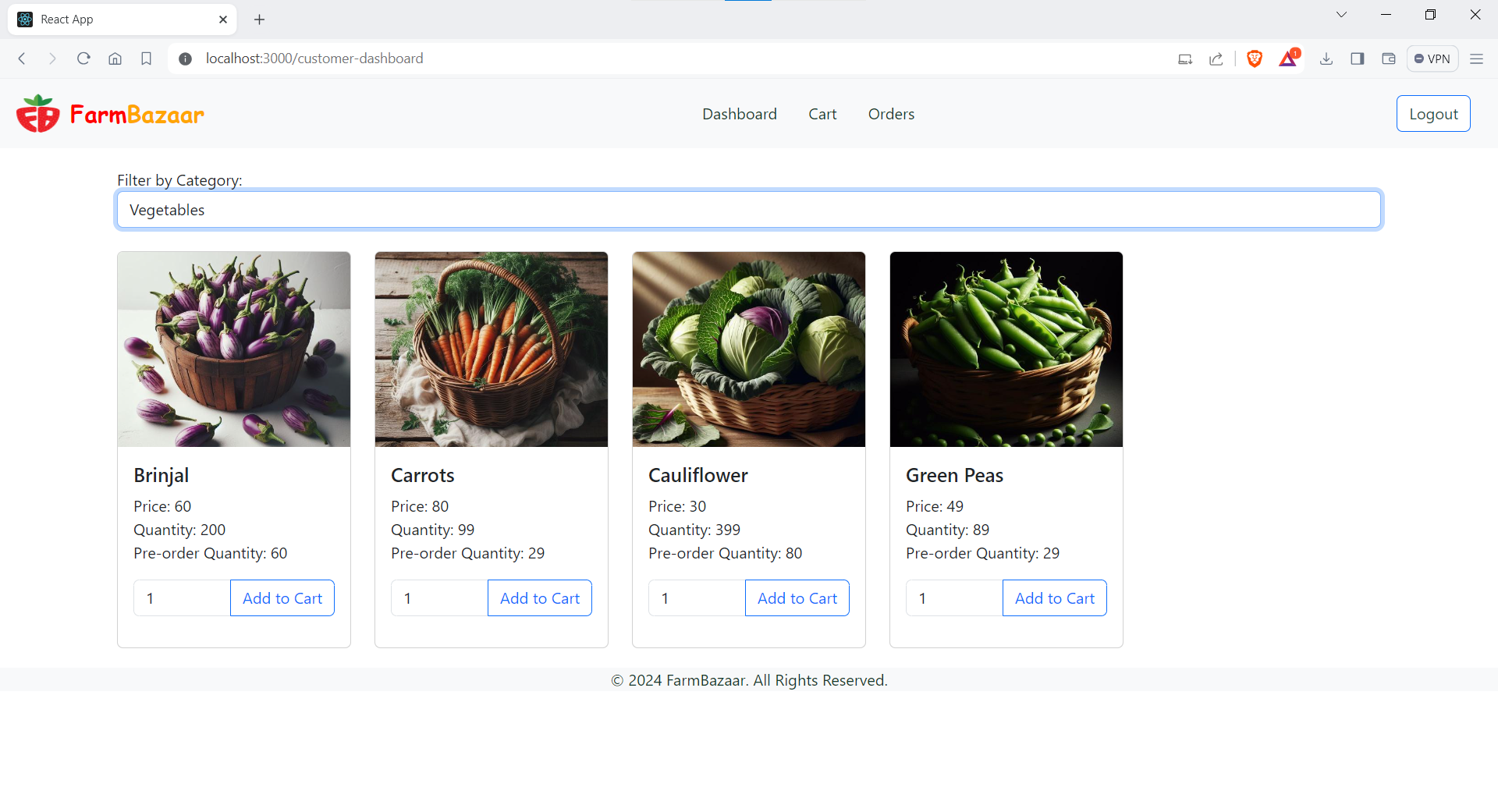




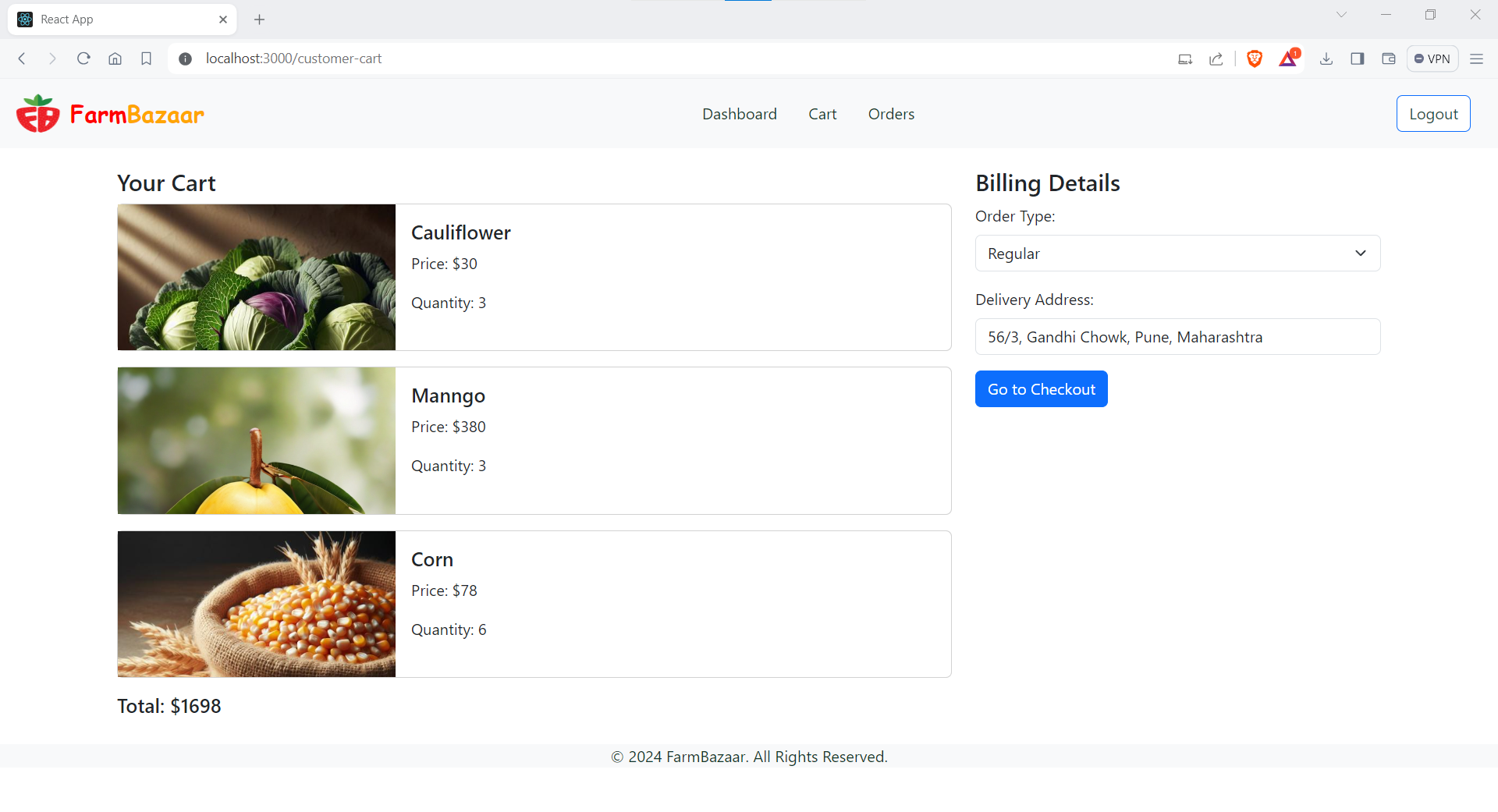
Users:



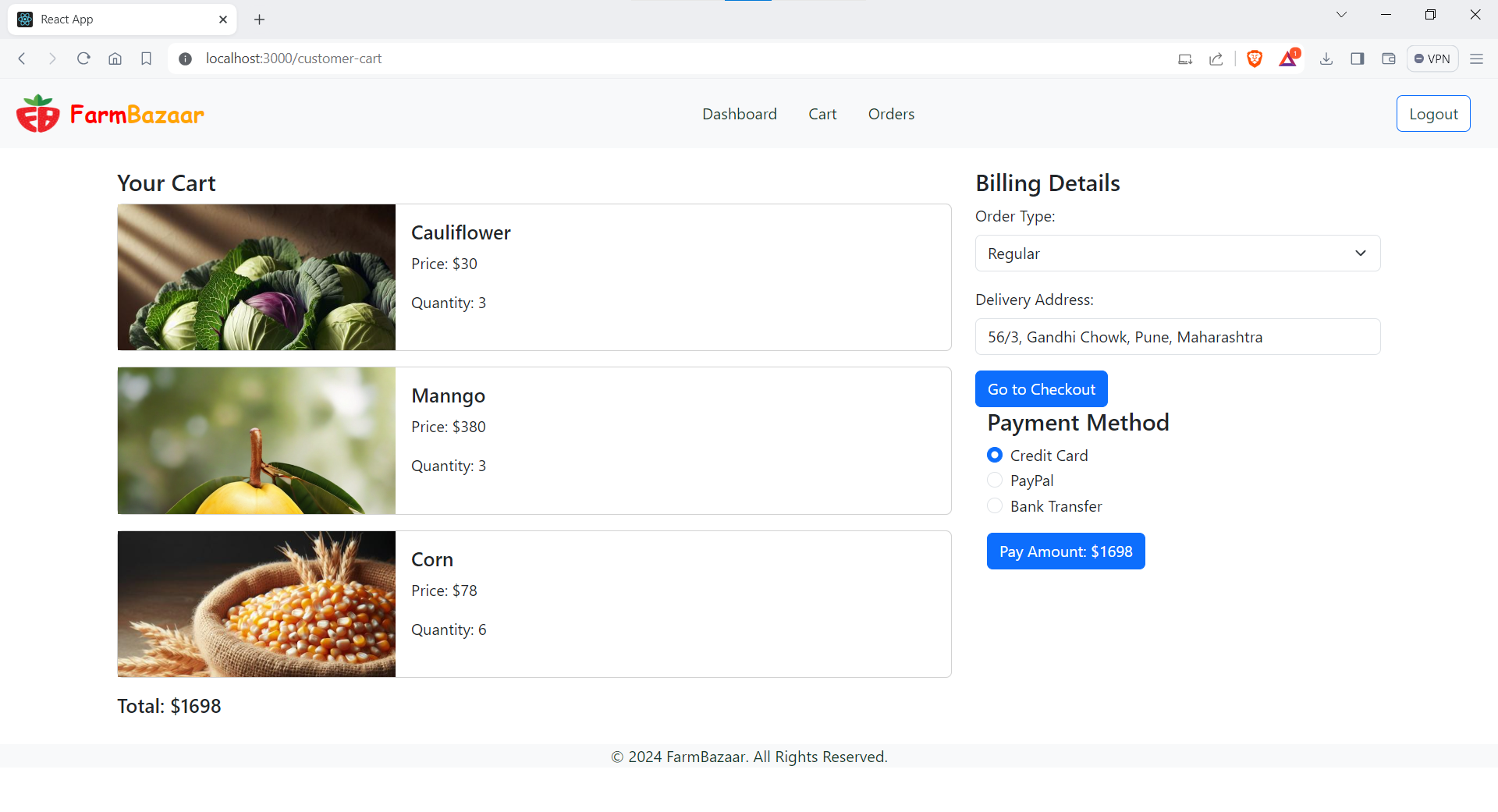
# Filter by Category:

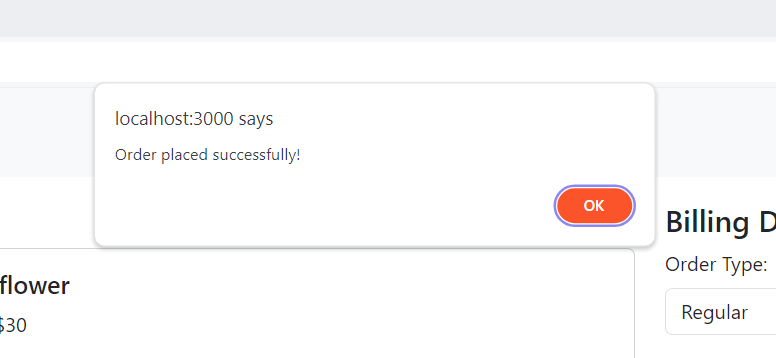


# Cart:

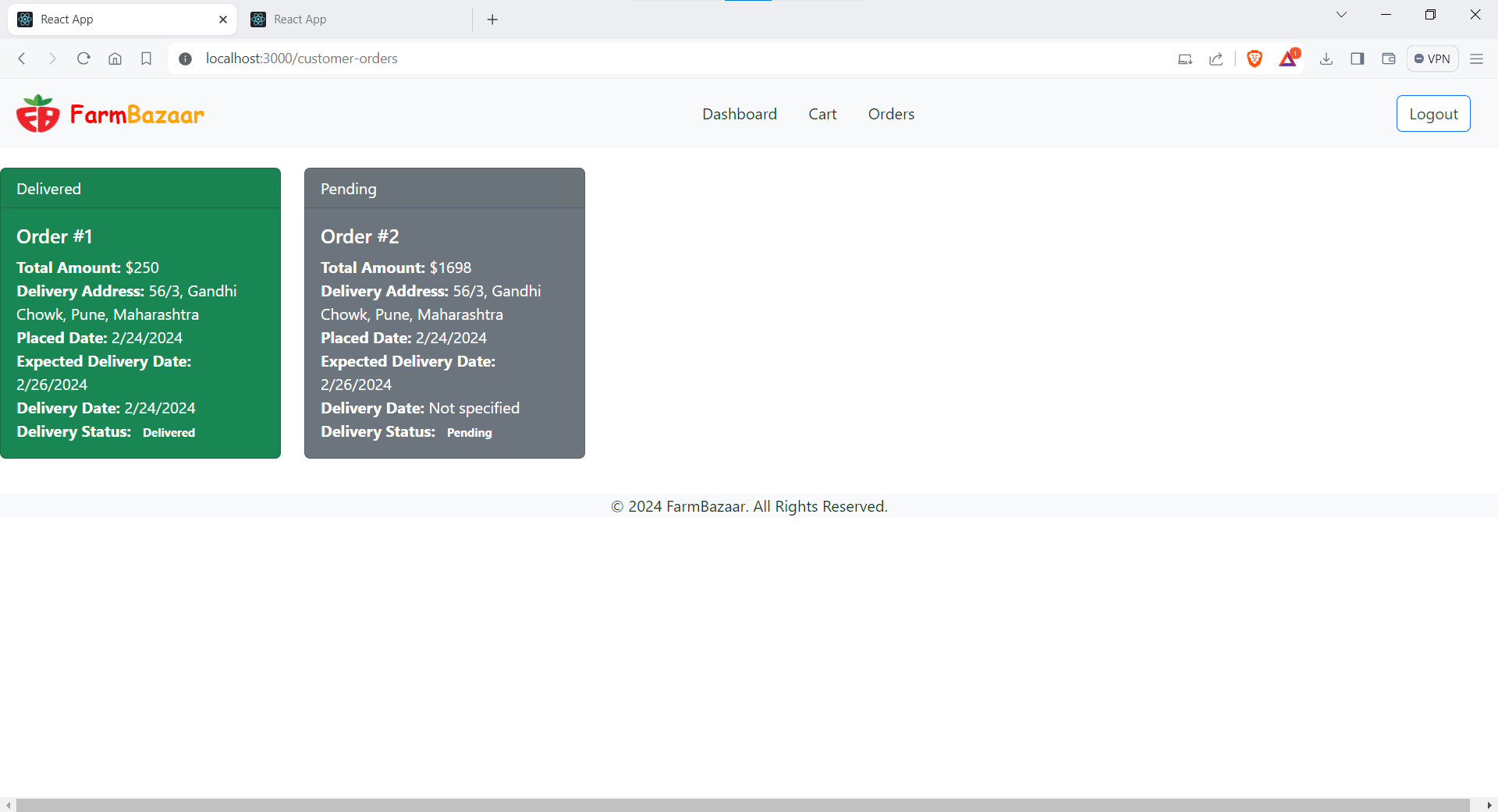


Checkout:

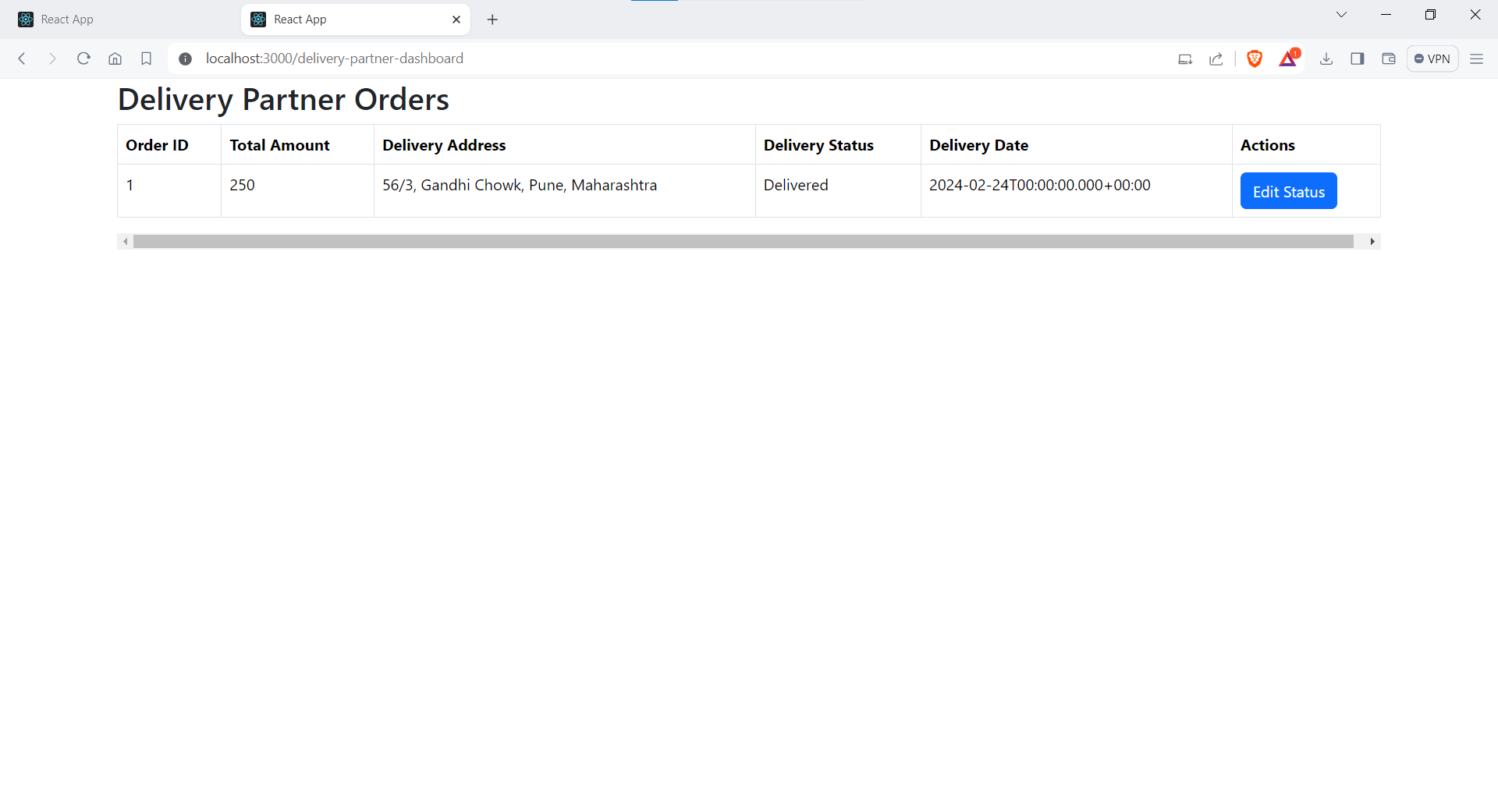




Orders



Delivery Partner:



# 6.TESTING

One of the main purposes of testing is to validate and verify that the system works as intended. No program or system design is perfect. However, if we implement the system without proper testing, then it may cause problems and lead to a bad user experience.

Testing and checking outcomes of each test gives us the best chance to detect and correct errors before the system is implemented in a production environment.

In the course of our project, we made an effort to manually test each component. In all cases, we obtained the desired results as demonstrated below.

### ADMIN FEATURES TEST

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Outcome | Result |
| 1. | Login as Admin | Fetched authenticated admin details saved in database | Passed |
| 2. | Add New Framer | Framer details along with all related details were added to database. | Passed |
| 3. | Update/Delete Farmer | Framer details along with all related details were added to database. | Passed |
| 4. | Add New Product Category | New product category gets added to database | Passed |
| 5. | Add New Product Item | New items and all its respective details saved in database | Passed |
| 6. | Update/Delete Product Details | The details of an existing products were update/deleted in the database | Passed |
| 7. | View Customers | All the Customer details are fetched from database | Passed |
| 8. | Update/Delete Customers | The details of an existing Customer is updated/deleted in database | Passed |
| 9. | Assign Products to Framers | Specific product to be assigned to specific framer and details get updated in database successfully | Passed |
| 10. | Logout | The session was cleared | Passed |

### B. CUSTOMER FEATURES TEST

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Outcome | Result |
| 1. | Register as Customer | New customer details saved in the database. | Passed |
| 2. | Login as Customer | Fetched authenticated user details saved in database. | Passed |
| 3. | Browse Products | Fetched list of all products from the database | Passed |
| 4. | Add Food items to Cart | The product along with necessary details were saved | Passed |
| 5. | Place Order | The cart items associated with the customer were saved in the form of placed order in the database | Passed |
| 6. | View Order History | The past orders made by the customers were fetched from the database. | Passed |
| 7. | Update User Profile | The profile information updates/modifications get reflected in database. | Passed |
| 8. | Logout | The Session was cleared | Passed |

### C. FARMER FEATURES TEST

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Outcome | Result |
| 1. | Register as Farmer | New Farmer details saved in the database. | Passed |
| 2. | Login as Farmer | Fetched authenticated user details saved in database. | Passed |
| 3. | Update stocks of products | The stocks details of the product assigned to farmer were updated in database | Passed |
| 4. | Logout | The Session was cleared | Passed |

### DELIVERY PARTNER FEATURES TEST

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Outcome | Result |
| 1. | Register as Delivery Partner | New Delivery Partner details saved in the database. | Passed |
| 2. | Login as Delivery Partner | Fetched authenticated user details saved in database. | Passed |
| 3. | Update the delivery status of the order | The status of the of the assigned orders to the delivery were updated in the database. | Passed |
| 4. | Logout | The Session was cleared | Passed |

# 7.CONCLUSION

“Farm Bazaar”, an online store application, was developed by our project team to simplify the online sale and purchase of Fresh and organic merchandise.

We tried using the latest technologies that are cross-platform and robust. Each and every software we used was open-source in nature, which keeps the cost of production at a minimum.

We were also meticulous about the user experience aspect of our application so that navigating our website is an easy and seamless experience.

In conclusion, “Farm Bazaar” is an application would definitely be a good choice for any fresh-food merchandise trading Farmers that wishes to enter the online market. At the same time, it provides one-stop platform for Customers to purchase their daily need of merchandise directly from authenticated Farmers.

We are confident that the numerous features and visually appealing look of application will certainly give a big boost to the Farmers.

# 8.FUTURE SCOPE

Using whatever we have learnt over the duration of this course, we tried to make our project as user-friendly and gave it as many features as possible in the limited time allotted for the project work. That said, there are certainly more features that can be added to our application. Some of those are mentioned below:

1. The most purchased and/or sponsored products can be highlighted as customer favourites to promote merchandise further.
2. Rating chart for Farmers and Products.
3. ProductDisplaybasedonCategories,distributingFarmersandrespective ratings.
4. Discounts can be given on a per-user basis depending on the customer’s purchase history as well as how many products they buy at the same time.
5. Customers can up-vote/down-vote/report feedbacks.
6. Additional payment means can be added.
7. In case the user forgets the password, a ‘reset password’ functionality can be added.
8. CAPTCHA can be added to login page.