

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**

: PRESENTED BY :

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

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

<b>ABSTRACT</b>	<b>4</b>
<b>1 .INTRODUCTION</b>	<b>5</b>
<b>2. PRODUCT OVERVIEW AND SUMMARY</b>	<b>6</b>
<b>2.1. PURPOSE</b>	<b>6</b>
<b>2.2. SCOPE</b>	<b>6</b>
<b>2.3. OVERVIEW</b>	<b>6</b>
A.TECHNOLOGIES USED	7
B. FEATURES PROVIDED	7
<b>2.4. FEASIBILITY STUDY</b>	<b>8</b>
A.TECHNICAL FEASIBILITY:	8
B. OPERATIONAL FEASIBILITY	9
C. ECONOMICAL FEASIBILITY	9
<b>3.REQUIREMENTS FULFILLED</b>	<b>10</b>
<b>3.1. FUNCTIONAL REQUIREMENTS</b>	<b>10</b>
<b>3.2. NON-FUNCTIONAL REQUIREMENTS</b>	<b>11</b>
<b>4.PROJECT DESIGN</b>	<b>12</b>
<b>4.1. DATA MODEL</b>	<b>12</b>
A. TABLES RELATED TO USER DETAILS	12
B. TABLES RELATED TO ORDERS	13
<b>4.2. FUNCTIONAL DECOMPOSITION DIAGRAM</b>	<b>16</b>
<b>4.3. USE CASE DIAGRAM</b>	<b>17</b>
<b>4.4. ACTIVITY DIAGRAM</b>	<b>20</b>
<b>4.6. ER DIAGRAM</b>	<b>20</b>
<b>5.PROJECT SCREENSHOTS</b>	<b>22</b>
<b>6.TESTING</b>	<b>26</b>
A. ADMIN FEATURES TEST	31
B. CUSTOMER FEATURES TEST	32
C. FARMER FEATURES TEST	32
D. DELIVERY PARTNER FEATURES TEST	32
<b>7.CONCLUSION</b>	<b>33</b>
<b>8.FUTURE SCOPE</b>	<b>34</b>

## **ABSTRACT**

The Business to Consumer Model has come a long way ever since its 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**

#### **i. FRONT END**

- HTML
- CSS
- JavaScript
- React
- Axios

#### **ii. BACK END**

- Spring Boot
- Spring Data JPA
- Hibernate

#### **iii.DATABASE MANAGEMENT SYSTEM**

- MySQL

### **B. FEATURES PROVIDED**

#### **i. FOR ADMININS**

- a. Login & Logout – Similar to customers and framers, admins can login & logout to access their account.
- b. Add/Update Farmers – Only admin is responsible for adding and updating the details of farmer.
- c. Delete Farmer – The admins can delete a farmer account if they need to for any purpose.
- d. Add New Category – Admins can add category.
- e. Delete Category – Admins can remove category.
- f. Add New Products – Admin can add new product with details as stock, price, name, quantity, image, category, etc.
- g. Manage Products – Admin can update the product details.
- h. View Users – Admin can view all registered users.
- i. Delete User – Admin can delete a user if need arises.
- j. View order details – Admin can view order details for all users.

#### **ii. FOR CUSTOMERS**

- a. Browse – Customers can browse the homepage to explore the entire products available.
- b. 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.
- c. View Profile – When logged in, customers can view their profile.
- d. 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.
- e. 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:

#### A. Tables Related to User Details

##### a. ADMIN Table:

```
mysql> DESC farmbazaardb.admin;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra       |
+-----+-----+-----+-----+-----+-----+
| user_id | int    | NO   | PRI | NULL    | auto_increment |
| address | varchar(255) | YES  |     | NULL    |             |
| fname   | varchar(255) | YES  |     | NULL    |             |
| is_active | bit(1) | YES  |     | NULL    |             |
| lname   | varchar(255) | YES  |     | NULL    |             |
| password | varchar(255) | YES  |     | NULL    |             |
| phno    | varchar(255) | YES  |     | NULL    |             |
| role    | varchar(255) | YES  |     | NULL    |             |
| username | varchar(255) | YES  |     | NULL    |             |
+-----+-----+-----+-----+-----+-----+
9 rows in set (0.00 sec)
```

##### b. FARMER Table

```
mysql> DESC farmbazaardb.farmer;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra       |
+-----+-----+-----+-----+-----+-----+
| user_id | int    | NO   | PRI | NULL    | auto_increment |
| address | varchar(255) | YES  |     | NULL    |             |
| fname   | varchar(255) | YES  |     | NULL    |             |
| is_active | bit(1) | YES  |     | NULL    |             |
| lname   | varchar(255) | YES  |     | NULL    |             |
| password | varchar(255) | YES  |     | NULL    |             |
| phno    | varchar(255) | YES  |     | NULL    |             |
| role    | varchar(255) | YES  |     | NULL    |             |
| username | varchar(255) | YES  |     | NULL    |             |
| total_profit | double | YES  |     | NULL    |             |
+-----+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

### c. CUSTOMER Table

```
mysql> DESC farmbazaardb.customer;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| user_id | int    | NO   | PRI | NULL    | auto_increment |
| address | varchar(255) | YES  |     | NULL    |
| fname   | varchar(255) | YES  |     | NULL    |
| is_active | bit(1) | YES  |     | NULL    |
| lname   | varchar(255) | YES  |     | NULL    |
| password | varchar(255) | YES  |     | NULL    |
| phno    | varchar(255) | YES  |     | NULL    |
| role    | varchar(255) | YES  |     | NULL    |
| username | varchar(255) | YES  |     | NULL    |
+-----+-----+-----+-----+-----+-----+
9 rows in set (0.02 sec)
```

### d. DELIVERY PARTNER Table

```
mysql> DESC farmbazaardb.delivery_partner;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| user_id | int    | NO   | PRI | NULL    | auto_increment |
| address | varchar(255) | YES  |     | NULL    |
| fname   | varchar(255) | YES  |     | NULL    |
| is_active | bit(1) | YES  |     | NULL    |
| lname   | varchar(255) | YES  |     | NULL    |
| password | varchar(255) | YES  |     | NULL    |
| phno    | varchar(255) | YES  |     | NULL    |
| role    | varchar(255) | YES  |     | NULL    |
| username | varchar(255) | YES  |     | NULL    |
| workload | int    | NO   |     | NULL    |
+-----+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

## B. Tables Related to Orders

### a. Product Details Table

```
mysql> DESC farmbazaardb.product;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id      | int    | NO   | PRI | NULL    | auto_increment |
| image   | longblob | YES  |     | NULL    |
| name    | varchar(255) | YES  |     | NULL    |
| pre_order_quantity | double | NO   |     | NULL    |
| price   | double  | NO   |     | NULL    |
| quantity | double | NO   |     | NULL    |
| category_id | int   | YES  | MUL | NULL    |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.01 sec)
```

## b. Order Table

```
mysql> DESC farmbazaardb.orders;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| order_id | int | NO | PRI | NULL | auto_increment |
| delivery_address | varchar(255) | YES | NULL |
| delivery_date | datetime(6) | YES | NULL |
| delivery_status | varchar(255) | YES | NULL |
| expected_delivery_date | datetime(6) | YES | NULL |
| order_status | varchar(255) | YES | NULL |
| placed_date | datetime(6) | YES | NULL |
| total_amount | double | NO | NULL |
| customer_id | int | YES | MUL | NULL |
| delivery_partner_id | int | YES | MUL | NULL |
+-----+-----+-----+-----+-----+
10 rows in set (0.01 sec)
```

## c. Order Details Table

```
mysql> DESC farmbazaardb.order_item;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| order_item_id | int | NO | PRI | NULL | auto_increment |
| price | double | NO | NULL |
| quantity | int | NO | NULL |
| updated_at | datetime(6) | YES | NULL |
| order_id | int | YES | MUL | NULL |
| product_id | int | YES | MUL | NULL |
+-----+-----+-----+-----+-----+
6 rows in set (0.01 sec)
```

## d. Category Table

```
mysql> DESC farmbazaardb.category;
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| id | int | NO | PRI | NULL | auto_increment |
| name | varchar(255) | YES | NULL |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

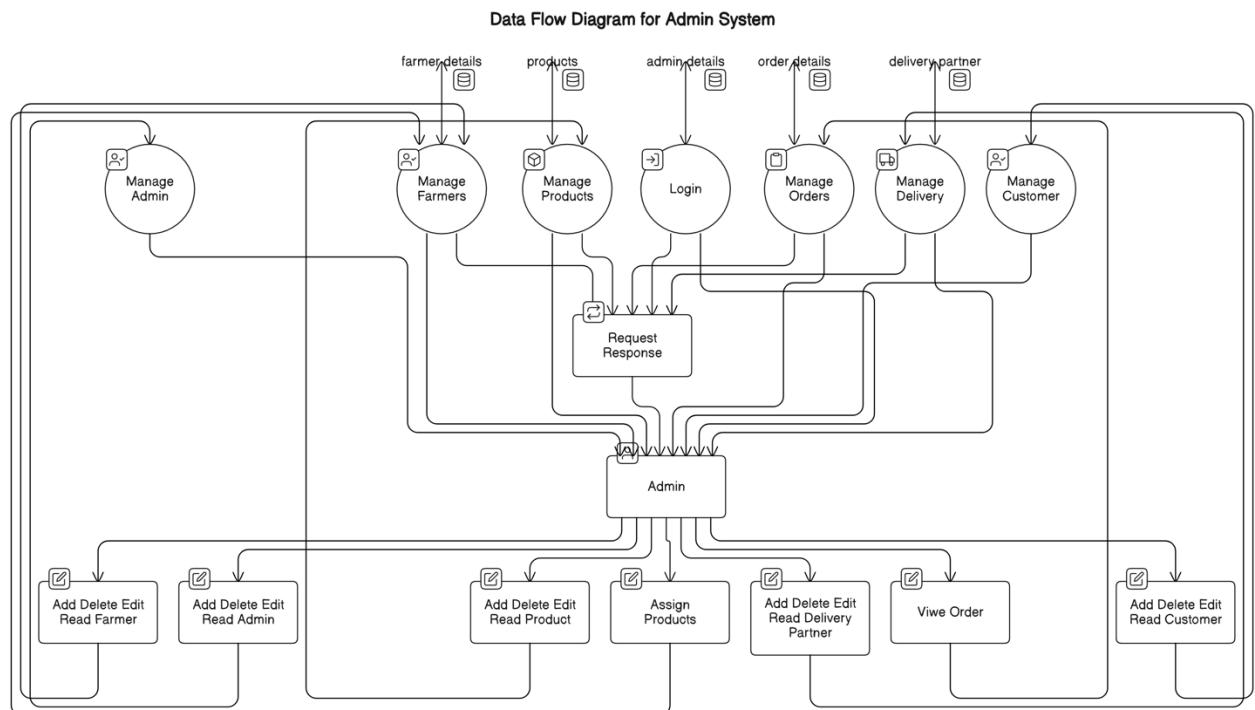
## e. Cart

```
mysql> desc farmbazaardb.cart;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | int  | NO   | PRI | NULL   | auto_increment |
| total_price | double | NO   |     | NULL   |               |
| updated_at | datetime(6) | YES  |     | NULL   |               |
| customer_id | int  | YES  | MUL | NULL   |               |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)
```

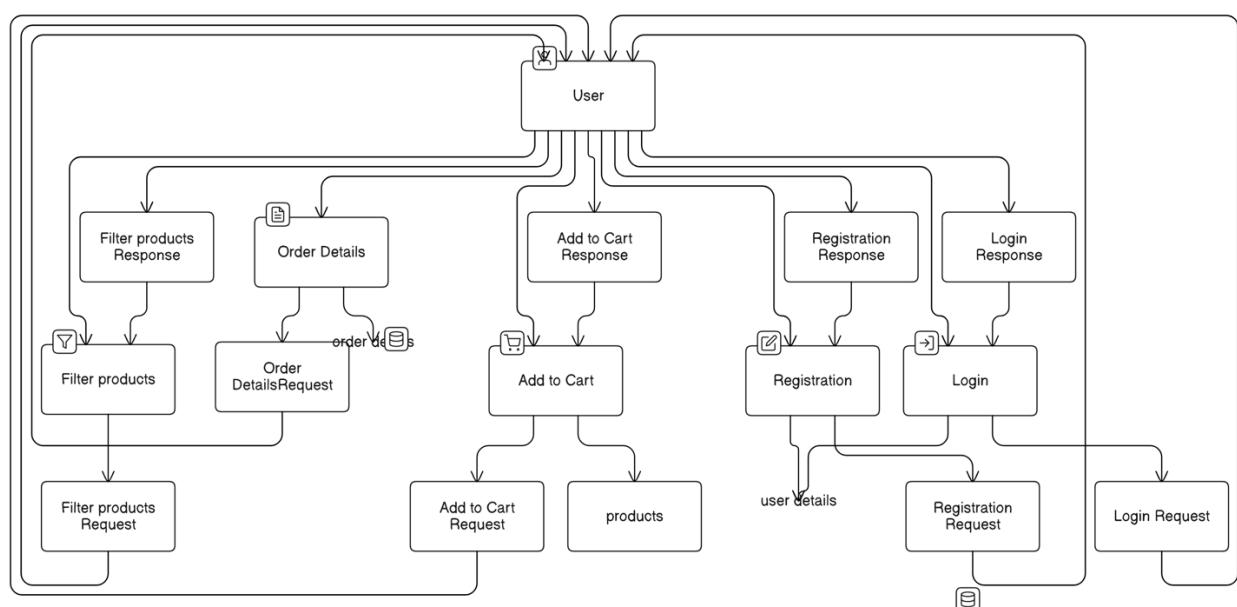
## f. Cart Items:

```
mysql> desc farmbazaardb.cart_item;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | int  | NO   | PRI | NULL   | auto_increment |
| price | double | NO   |     | NULL   |               |
| quantity | int  | NO   |     | NULL   |               |
| cart_id | int  | YES  | MUL | NULL   |               |
| product_id | int  | YES  | MUL | NULL   |               |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

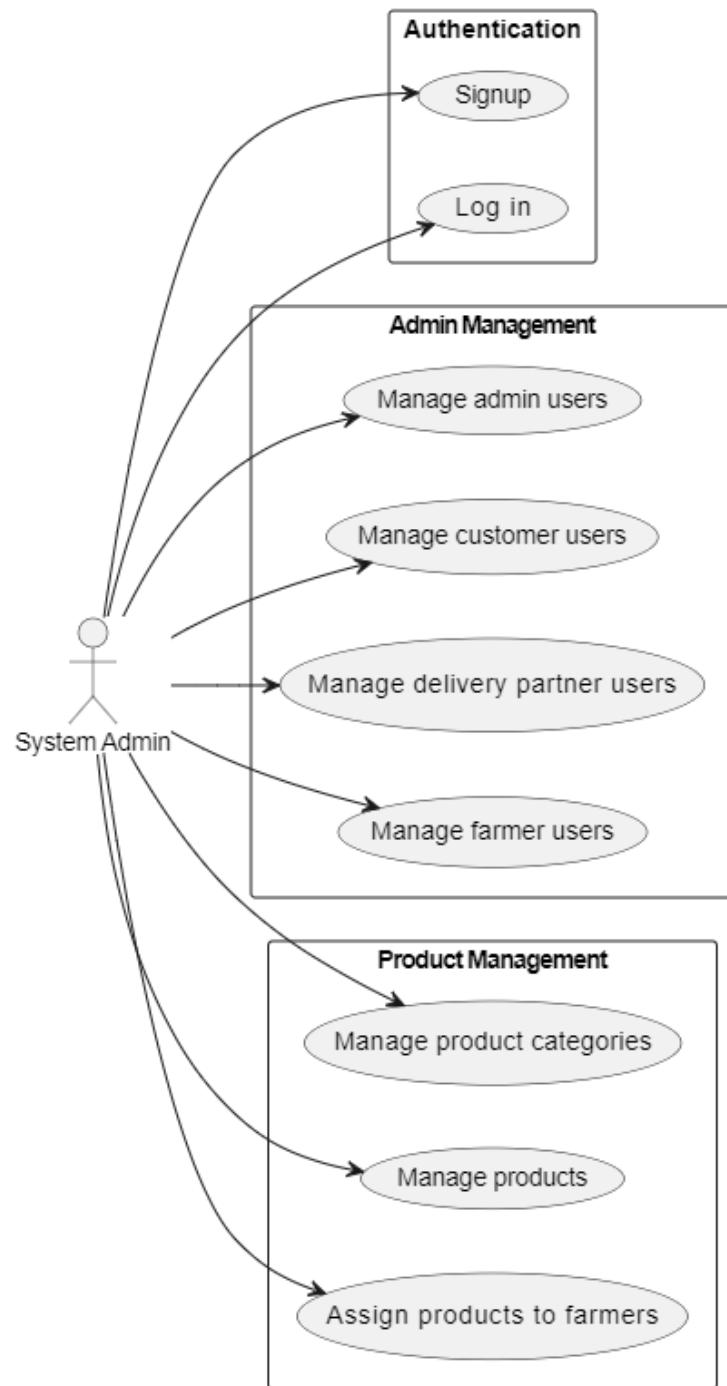
## 4.2. FUNCTIONAL DECOMPOSITION DIAGRAM

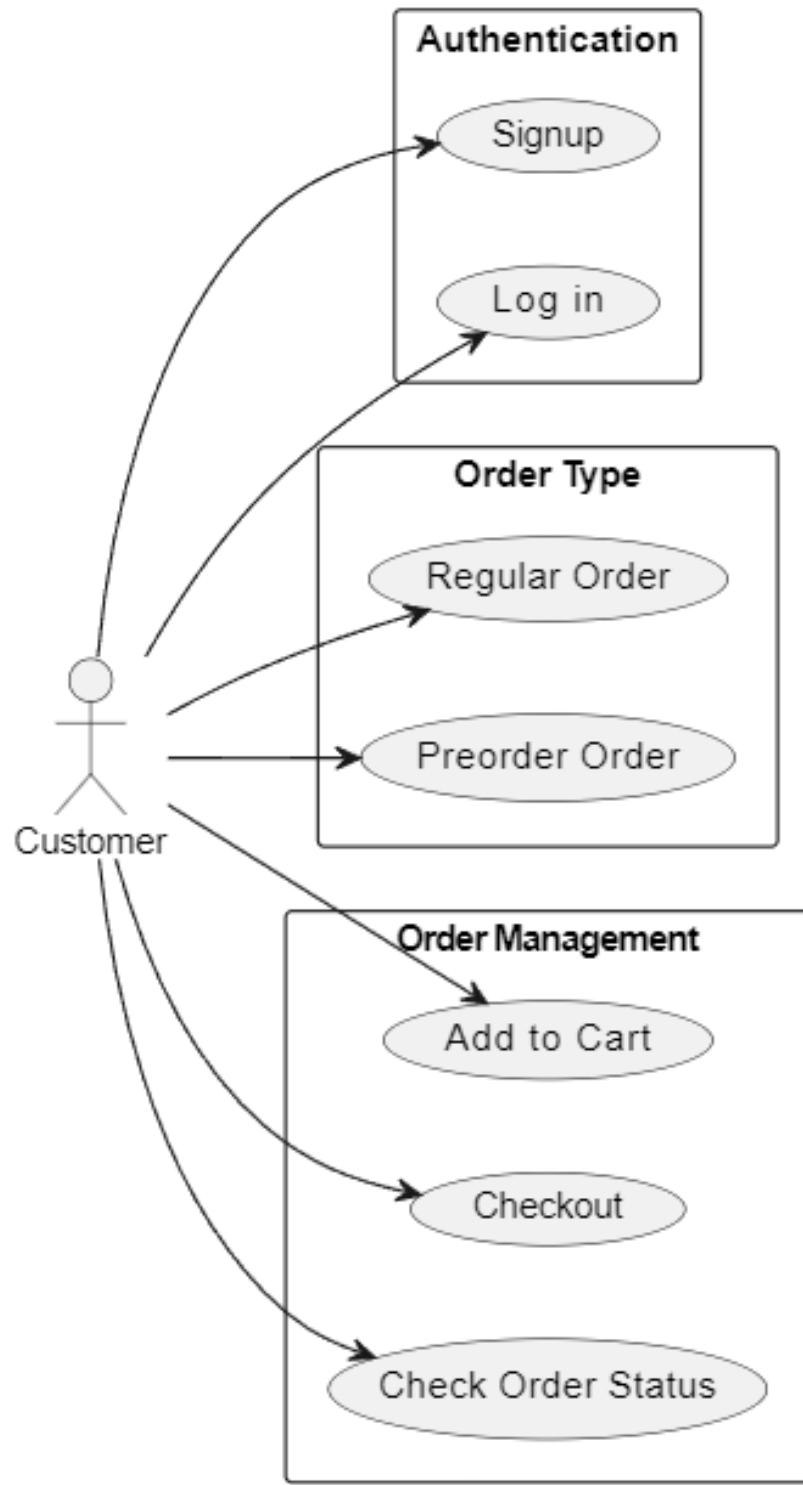


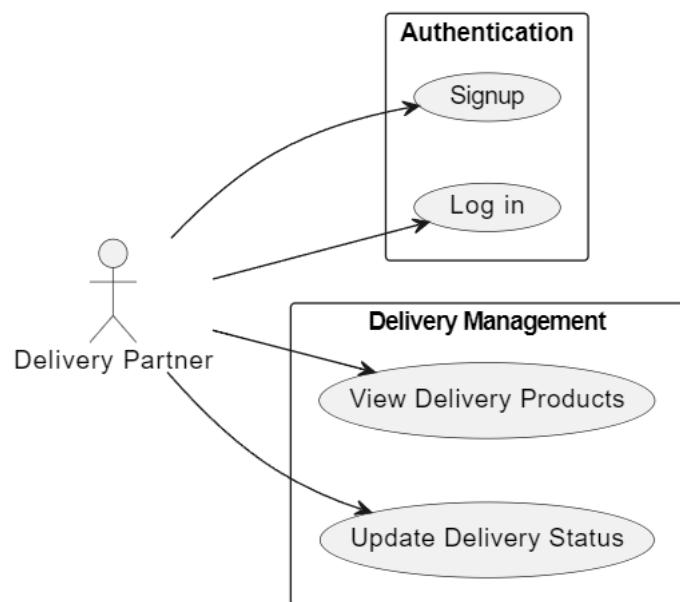
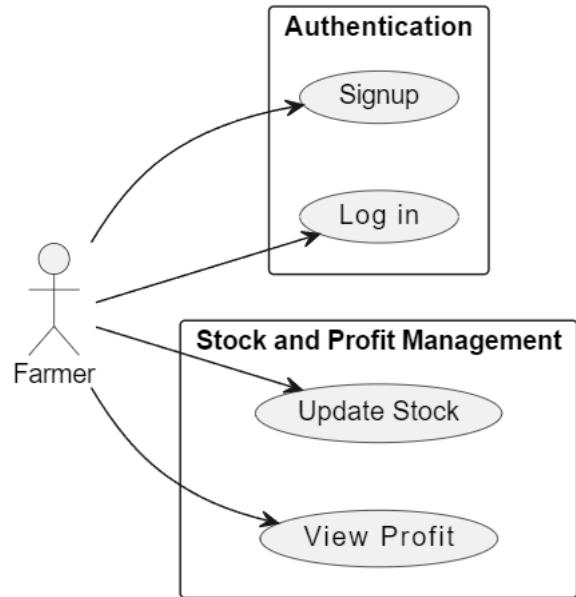
User Data Flow Diagram



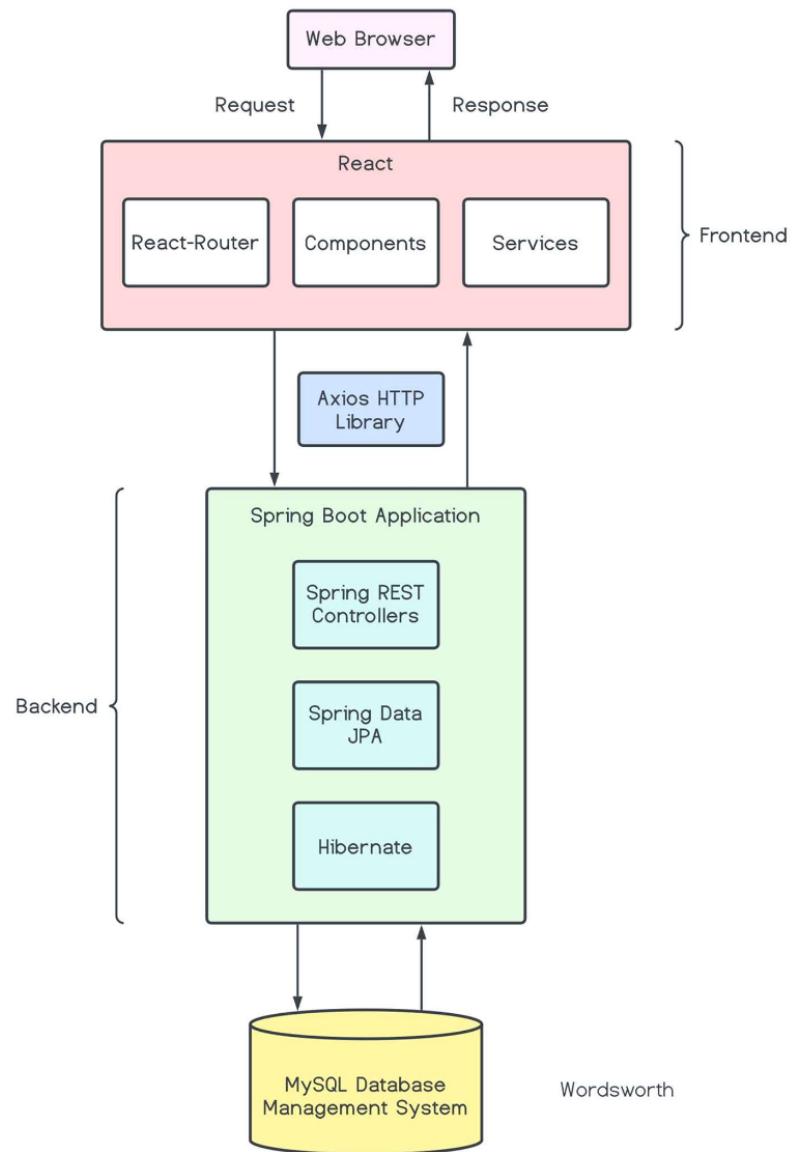
### 4.3. USE CASE DIAGRAM



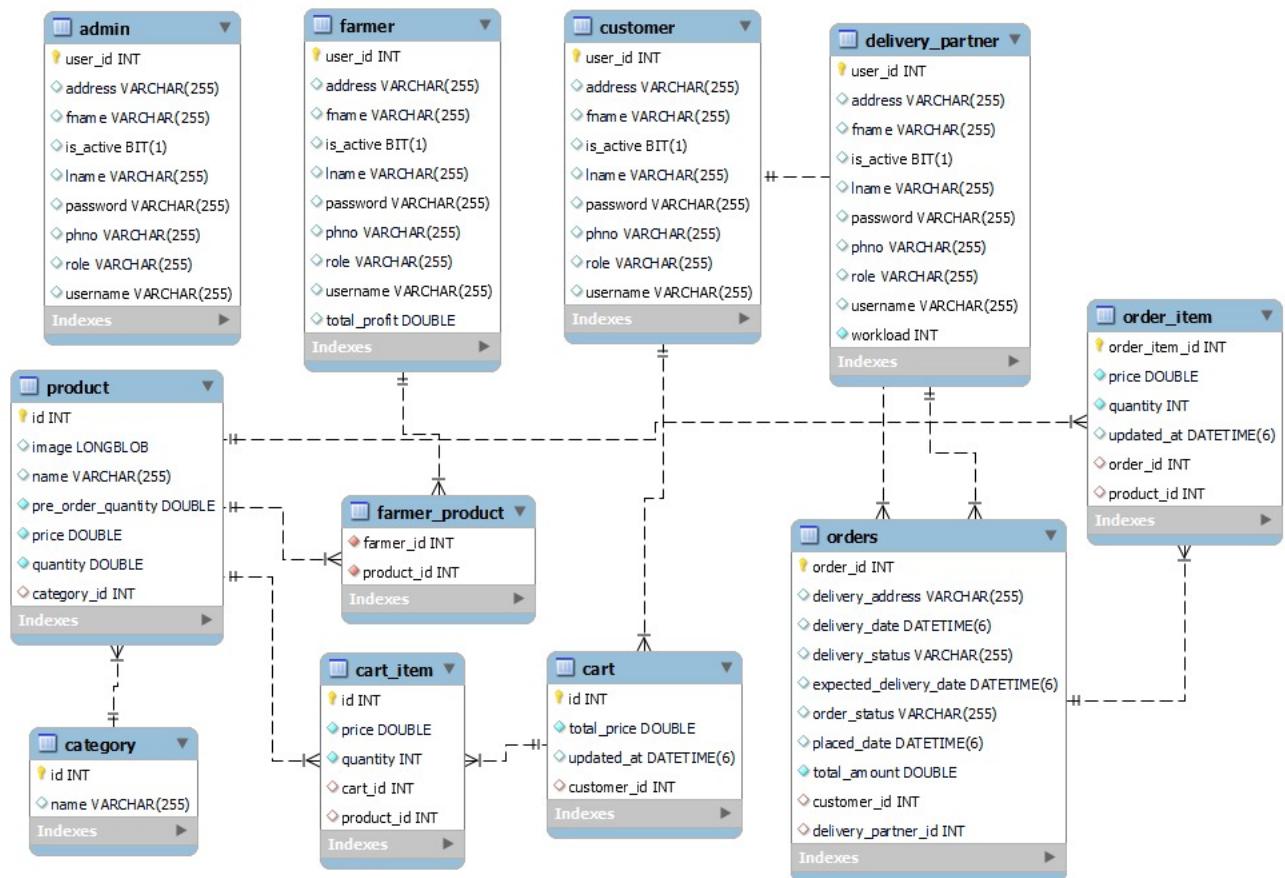




## 4.4. SYSTEM ARCHITECTURE DIAGRAM



## 4.6. ER DIAGRAM



## 5.PROJECT SCREENSHOTS

### Admin Homepage

The screenshot shows the FarmBazaar Admin Panel. At the top, there's a header bar with the FarmBazaar logo, a search bar, and navigation links for Admins, Customers, Delivery Partners, Farmers, Categories, Products, Assign Products, Orders, and Logout. Below the header is a section titled "Admin Panel" containing eight cards:

- Admin Users**: Manage admin users. [Go to Admin Users](#)
- Customer Users**: Manage customer users. [Go to Customer Users](#)
- Delivery Partner Users**: Manage delivery partner users. [Go to Delivery Partner Users](#)
- Farmer Users**: Manage farmer users. [Go to Farmer Users](#)
- Categories**: Manage product categories. [Go to Categories](#)
- Products**: Manage products. [Go to Products](#)
- Assign Product to Farmer**: Assign products to farmers. [Go to Assign Product to Farmer](#)
- View all Orders**: View all orders placed by customers. [Go to Assign Product to Farmer](#)

At the bottom, a footer bar displays the copyright notice: © 2024 FarmBazaar. All Rights Reserved.

### Admin Users (CRUD)

The screenshot shows the "Admin Users" page under the FarmBazaar Admin Panel. The URL is localhost:3000/admin/admin-users. The page features a table with columns: Username, Password, First Name, Last Name, Phone Number, Address, and Actions (with Edit and Delete buttons). A single row is visible for the user "admin@farmbazaar.com". At the top right, there's a blue "Add New" button. The footer contains the copyright notice: © 2024 FarmBazaar. All Rights Reserved.

Username	Password	First Name	Last Name	Phone Number	Address	Actions
admin@farmbazaar.com	admin	Shubham	Samarth	7896541230	Karve Nagar, Pune	<a href="#">Edit</a> <a href="#">Delete</a>

## Customer Users (CRUD)

The screenshot shows a web browser window titled "React App" with the URL "localhost:3000/admin/customer-users". The header includes the FarmBazaar logo and navigation links for Admins, Customers, Delivery Partners, Farmers, Categories, Products, Assign Products, and Orders, along with a Logout button. The main content area is titled "Customer Users" and features a table with the following data:

Username	Password	First Name	Last Name	Phone Number	Address	Actions	
user1@farmbazaar.com	user	Sneha	Reddy	6543210987	33/8, Jubilee Hills, Pune, Maharashtra	<a href="#">Edit</a>	<a href="#">Delete</a>
user2@farmbazaar.com	user	Rohan	Das	45632147890	56/3, Gandhi Chowk, Pune, Maharashtra	<a href="#">Edit</a>	<a href="#">Delete</a>
user3@farmbazaar.com	user	Ananya	Singh	7896541230	14B, Civil Lines, Pune, Maharashtra	<a href="#">Edit</a>	<a href="#">Delete</a>
user4@farmbazaar.com	user	Rahul	Kumar	9876543210	102, Ganga Nagar, Pune, Maharashtra	<a href="#">Edit</a>	<a href="#">Delete</a>
user5@farmbazaar.com	user	Nisha	Verma	7418529630	9/15, Green Park, Pune, Maharashtra	<a href="#">Edit</a>	<a href="#">Delete</a>
user6@farmbazaar.com	user	Vivek	Sharma	8523697410	45, Kalighat Road, Pune, Maharashtra	<a href="#">Edit</a>	<a href="#">Delete</a>

At the bottom, there is a copyright notice: "© 2024 FarmBazaar. All Rights Reserved."

## Delivery Partners Users (CRUD)

The screenshot shows a web browser window titled "React App" with the URL "localhost:3000/admin/delivery-partner-users". The header includes the FarmBazaar logo and navigation links for Admins, Customers, Delivery Partners, Farmers, Categories, Products, Assign Products, and Orders, along with a Logout button. The main content area is titled "Delivery Partner Users" and features a table with the following data:

Username	Password	First Name	Last Name	Phone Number	Address	Actions	
delivery1@farmbazaar.com	delivery	Rahul	Kumar	8796541235	9/15, Green Park, Pune, Maharashtra	<a href="#">Edit</a>	<a href="#">Delete</a>
delivery2@farmbazaar.com	delivery	Sachin	Sharma	7896541230	45, Kalighat Road, Pune, Maharashtra	<a href="#">Edit</a>	<a href="#">Delete</a>
delivery3@farmbazaar.com	delivery	Riya	Patel	9876543210	102, Ganga Nagar, Pune, Maharashtra	<a href="#">Edit</a>	<a href="#">Delete</a>
delivery4@farmbazaar.com	delivery	Amit	Singh	7418529630	14B, Civil Lines, Pune, Maharashtra	<a href="#">Edit</a>	<a href="#">Delete</a>

At the bottom, there is a copyright notice: "© 2024 FarmBazaar. All Rights Reserved."

## Farmer Users (CRUD)

The screenshot shows a browser window titled "React App" with the URL "localhost:3000/admin/farmer-users". The page header includes the FarmBazaar logo and navigation links for Admins, Customers, Delivery Partners, Farmers, Categories, Products, Assign Products, and Orders, along with a Logout button. A blue "Add New" button is located at the top right of the table. The main content is a table titled "Farmer Users" with columns: Username, Password, First Name, Last Name, Phone Number, Address, and Actions (Edit and Delete buttons). The table contains 8 rows of data, each representing a farmer user with a unique email and address.

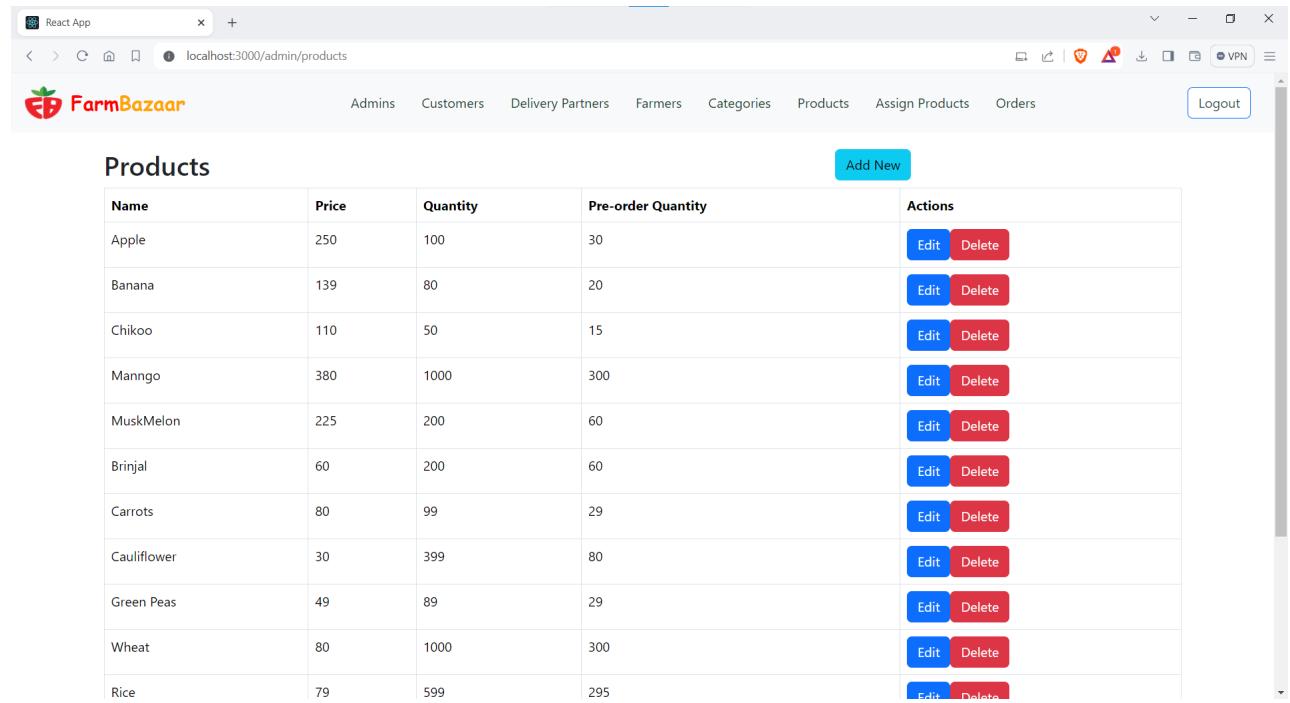
Username	Password	First Name	Last Name	Phone Number	Address	Actions
farmer1@farmbazaar.com	farmer	Arjun	Patel	9876543210	22A, Rishi Road, Mumbai, Maharashtra	<button>Edit</button> <button>Delete</button>
farmer2@farmbazaar.com	farmer	Priya	Sharma	8765432109	5/12, Shanti Nagar, Pune, Maharashtra	<button>Edit</button> <button>Delete</button>
farmer3@farmbazaar.com	farmer	Rahul	Gupta	7896541230	78, Mahatma Gandhi Lane, Pune, Maharashtra	<button>Edit</button> <button>Delete</button>
farmer4@farmbazaar.com	farmer	Sneha	Reddy	6543210987	33/8, Jubilee Hills, Pune, Maharashtra	<button>Edit</button> <button>Delete</button>
farmer5@farmbazaar.com	farmer	Ananya	Singh	5432109876	102, Ganga Nagar, Pune, Maharashtra	<button>Edit</button> <button>Delete</button>
farmer6@farmbazaar.com	farmer	Rohan	Das	4321098765	14B, Civil Lines, Pune, Maharashtra	<button>Edit</button> <button>Delete</button>
farmer7@farmbazaar.com	farmer	Nisha	Verma	3210987654	9/15, Green Park, Pune, Maharashtra	<button>Edit</button> <button>Delete</button>
farmer8@farmbazaar.com	farmer	Vivek	Sharma	2109876543	45, Kalighat Road, Pune, Maharashtra	<button>Edit</button> <button>Delete</button>

## Categories (CRUD)

The screenshot shows a browser window titled "React App" with the URL "localhost:3000/admin/categories". The page header includes the FarmBazaar logo and navigation links for Admins, Customers, Delivery Partners, Farmers, Categories, Products, Assign Products, and Orders, along with a Logout button. A blue "Add Category" button is located at the top right of the table. The main content is a table titled "Categories" with columns: Name and Actions (Edit and Delete buttons). The table contains 3 rows of data, each representing a category: Vegetables, Fruits, and Grains.

Name	Actions
Vegetables	<button>Edit</button> <button>Delete</button>
Fruits	<button>Edit</button> <button>Delete</button>
Grains	<button>Edit</button> <button>Delete</button>

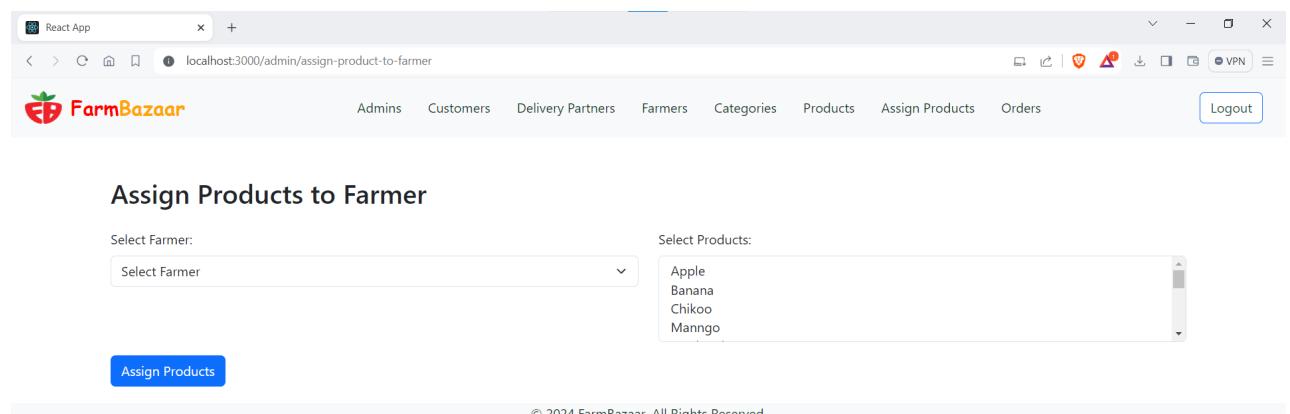
## Products



The screenshot shows a table titled "Products" with the following data:

Name	Price	Quantity	Pre-order Quantity	Actions
Apple	250	100	30	<button>Edit</button> <button>Delete</button>
Banana	139	80	20	<button>Edit</button> <button>Delete</button>
Chikoo	110	50	15	<button>Edit</button> <button>Delete</button>
Mango	380	1000	300	<button>Edit</button> <button>Delete</button>
MuskMelon	225	200	60	<button>Edit</button> <button>Delete</button>
Brinjal	60	200	60	<button>Edit</button> <button>Delete</button>
Carrots	80	99	29	<button>Edit</button> <button>Delete</button>
Cauliflower	30	399	80	<button>Edit</button> <button>Delete</button>
Green Peas	49	89	29	<button>Edit</button> <button>Delete</button>
Wheat	80	1000	300	<button>Edit</button> <button>Delete</button>
Rice	79	599	295	<button>Edit</button> <button>Delete</button>

## Assign Products to Farmer:



The screenshot shows a form titled "Assign Products to Farmer" with the following fields:

- Select Farmer: A dropdown menu currently showing "Select Farmer".
- Select Products: A dropdown menu showing a list of products: Apple, Banana, Chikoo, Mango.
- Assign Products: A blue button at the bottom left.

At the bottom of the page, there is a footer bar with the text "© 2024 FarmBazaar. All Rights Reserved."

## Orders:

The screenshot shows a browser window titled "React App" with the URL "localhost:3000/admin/orders". The header includes the FarmBazaar logo, navigation links for Admins, Customers, Delivery Partners, Farmers, Categories, Products, Assign Products, Orders, and a Logout button. The main content area is titled "Orders" and displays a table with one row of data. The columns are: Order ID, Customer ID, Customer Name, Total Amount, Order Status, Delivery Status, Delivery Address, Placed Date, Expected Delivery Date, and Delivery Date. The data in the first row is: 1, 2, Rohan Das, 250, Placed., Pending, 56/3, Gandhi Chowk, Pune, Maharashtra, 2024-02-23T21:23:01.735+00:00, 2024-02-25T21:23:01.735+00:00, and null. At the bottom of the page is a copyright notice: "© 2024 FarmBazaar. All Rights Reserved."

Order ID	Customer ID	Customer Name	Total Amount	Order Status	Delivery Status	Delivery Address	Placed Date	Expected Delivery Date	Delivery Date
1	2	Rohan Das	250	Placed.	Pending	56/3, Gandhi Chowk, Pune, Maharashtra	2024-02-23T21:23:01.735+00:00	2024-02-25T21:23:01.735+00:00	

## Farmer:

The screenshot shows a browser window titled "React App" with the URL "localhost:3000/farmer/farmer-stocks". The header includes the FarmBazaar logo and a Logout button. The main content area is titled "Farmer Stocks" and displays a table with three rows of data. The columns are: Name, Price, Quantity, Pre-order Quantity, and Actions. The data in the rows is: Apple (Price 250, Quantity 150, Pre-order 45, Actions Edit Quantity), Banana (Price 139, Quantity 80, Pre-order 20, Actions Edit Quantity), and Chikoo (Price 110, Quantity 50, Pre-order 15, Actions Edit Quantity). At the bottom of the page is a copyright notice: "© 2024 FarmBazaar. All Rights Reserved."

Name	Price	Quantity	Pre-order Quantity	Actions
Apple	250	150	45	Edit Quantity
Banana	139	80	20	Edit Quantity
Chikoo	110	50	15	Edit Quantity

The screenshot shows a browser window titled "React App" with the URL "localhost:3000/farmer-dashboard". At the top left is the FarmBazaar logo. On the right side of the header are icons for file operations, a shield, a person, a download, and a "Logout" button. The main content area has two sections: "Update Stock" on the left with a sub-instruction "Update the stocks for your assigned product." and a blue "Update Stock" button; and "Profit" on the right showing "Profit: 3750.00₹". A footer bar at the bottom contains the text "© 2024 FarmBazaar. All Rights Reserved."

## Users:

The screenshot shows a browser window titled "React App" with the URL "localhost:3000/customer-dashboard". The header includes the FarmBazaar logo, navigation links for "Dashboard", "Cart", and "Orders", and a "Logout" button. Below the header is a search bar labeled "Filter by Category:" with a dropdown menu showing "All". The main content displays a grid of five fruit items in the first row and four vegetable items in the second row. Each item card includes a thumbnail image, the name, price, quantity, pre-order quantity, a quantity selector (1), and an "Add to Cart" button.

Product	Price	Quantity	Pre-order Quantity
Apple	250	150	45
Banana	139	80	20
Chikoo	110	50	15
Mango	380	1000	300
MuskMelon	225	200	60
Eggplant			
Carrots			
Cabbage			
Peas			
Wheat			

## Filter by Category:

The screenshot shows a web browser window titled "React App" with the URL "localhost:3000/customer-dashboard". The header includes the FarmBazaar logo, "Dashboard", "Cart", "Orders", and a "Logout" button. A search bar at the top is labeled "Filter by Category:" and contains the text "Vegetables". Below the search bar is a grid of four vegetable items, each with a thumbnail image, name, price, quantity, pre-order quantity, a quantity selector (1), and an "Add to Cart" button.

Image	Name	Price	Quantity	Pre-order Quantity	Action
	Brinjal	60	200	60	1 Add to Cart
	Carrots	80	99	29	1 Add to Cart
	Cauliflower	30	399	80	1 Add to Cart
	Green Peas	49	89	29	1 Add to Cart

© 2024 FarmBazaar. All Rights Reserved.

## Cart:

The screenshot shows a web browser window titled "React App" with the URL "localhost:3000/customer-cart". The header includes the FarmBazaar logo, "Dashboard", "Cart", "Orders", and a "Logout" button. The main content is divided into two sections: "Your Cart" on the left and "Billing Details" on the right. "Your Cart" displays three items: Cauliflower, Mango, and Corn, each with a thumbnail image, name, price, quantity, and a "Go to Checkout" button. "Billing Details" includes fields for "Order Type" (set to "Regular") and "Delivery Address" (set to "56/3, Gandhi Chowk, Pune, Maharashtra").

Image	Name	Price	Quantity	Action
	Cauliflower	\$30	3	Go to Checkout
	Mango	\$380	3	
	Corn	\$78	6	

Total: \$1698

© 2024 FarmBazaar. All Rights Reserved.

## Checkout:

The screenshot shows a web browser window for a FarmBazaar customer cart. The page has a header with the FarmBazaar logo, navigation links for Dashboard, Cart, Orders, and Logout. The main content area is divided into two sections: 'Your Cart' on the left and 'Billing Details' on the right.

**Your Cart**

- Cauliflower**  
Price: \$30  
Quantity: 3
- Mango**  
Price: \$380  
Quantity: 3
- Corn**  
Price: \$78  
Quantity: 6

Total: \$1698

**Billing Details**

Order Type: Regular

Delivery Address: 56/3, Gandhi Chowk, Pune, Maharashtra

**Go to Checkout**

**Payment Method**

Credit Card  
 PayPal  
 Bank Transfer

**Pay Amount: \$1698**

© 2024 FarmBazaar. All Rights Reserved.

localhost:3000 says  
Order placed successfully!

OK

**Billing D**

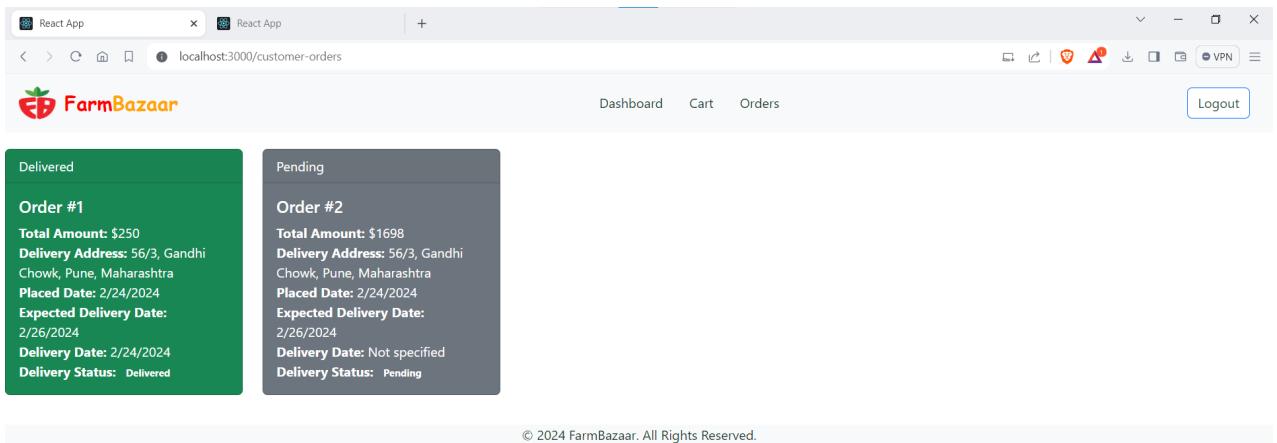
Order Type:

Regular

flower

\$30

Orders



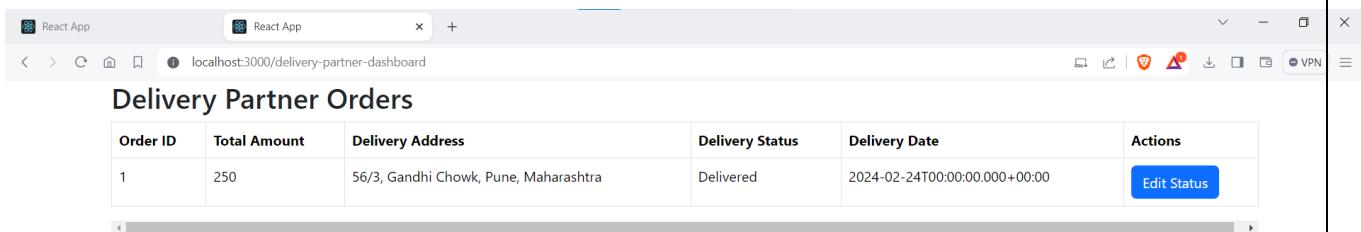
The screenshot shows a web browser window with two tabs both titled "React App". The active tab is at "localhost:3000/customer-orders". The page header includes the FarmBazaar logo, a navigation bar with "Dashboard", "Cart", "Orders", and a "Logout" button. Below the header, there are two main sections: "Delivered" (green background) containing Order #1 details, and "Pending" (grey background) containing Order #2 details.

Order #1	
Total Amount:	\$250
Delivery Address:	56/3, Gandhi Chowk, Pune, Maharashtra
Placed Date:	2/24/2024
Expected Delivery Date:	2/26/2024
Delivery Date:	2/24/2024
Delivery Status:	Delivered

Order #2	
Total Amount:	\$1698
Delivery Address:	56/3, Gandhi Chowk, Pune, Maharashtra
Placed Date:	2/24/2024
Expected Delivery Date:	2/26/2024
Delivery Date:	Not specified
Delivery Status:	Pending

© 2024 FarmBazaar. All Rights Reserved.

## Delivery Partner:



The screenshot shows a web browser window with two tabs both titled "React App". The active tab is at "localhost:3000/delivery-partner-dashboard". The page title is "Delivery Partner Orders". A table displays a single order row.

Order ID	Total Amount	Delivery Address	Delivery Status	Delivery Date	Actions
1	250	56/3, Gandhi Chowk, Pune, Maharashtra	Delivered	2024-02-24T00:00:00.000+00:00	<button>Edit Status</button>

## **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.

### **A. 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

## D. 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 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. Product Display based on Categories, distributing Farmers and respective 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.