INSTITUTE FOR ADVANCED COMPUTING AND SOFTWARE DEVELOPMENT (IACSD),

AKURDI, PUNE

Documentation On

"FRESH BASKET -
EAT WELL, LIVE WELL WITH
FRESH BASKET"

PG-DAC MARCH 2023

Submitted By:

Group No: 70

Roll No.	Name:
233149	Deepak Dhake
233159	Tanishq Kadam

Mrs. Manjiri Deshpande
Project Guide

Mr. Rohit Puranik
Centre Coordinator

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.

ACKNOWLEDGEMENT

The project “Fresh Basket” was a great learning experience for us and we are submitting this work to Advanced Computing Training School (IACSD Akurdi, Pune).

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

Our heartfelt thanks go to Mr. Rohit Puranik (Course Coordinator, PG - DAC) who gave us all the required support and kind coordination to provide all the necessities to complete the project and throughout the course up to the last day of the course.

We would like to express our sincere gratitude towards Mrs. Madhura Anturkar, 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.

DEEPAK DHAKE (233149)

TANISHQ KADAM (233159)

Table of Contents

List of Figures

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 centers 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.

Problem Statement:

As fresh basket wants to expand its business alongside with improving farmers livelihood. There are some major barriers which needs to be addressed to achieve the mentioned objectives. One of the major barriers is as the trends are changing the demand for product is changing. Consumer demands towards healthier, convenient and highly differentiated food. Consumer demands value addition in the food. But for fulfilling consumer demand farmers needs to train in the parallel ways. Farmers are not aware about modern ways of farming like organic farming, value addition in product etc. The next issue of concern is the problem of middlemen. Just because of inefficiency of middlemen the quality of product is declining as well as many cases of farmers exploitations are arising. Because of these issues farmers don't want to do business with ecommerce firms or they won't get ample opportunities and guidance to join the mission.

Aims and Objective:

Farming is not a bad business to be in - people will always need to eat. But farmers are at an inevitable disadvantage as they need to find a market for their produce and one that will pay a good margin, so they can maintain and grow their business. The major objectives of fresh basket is to help farmers to double their income along with increasing awareness and bringing more farmer community on board also fix the supply chain by reinstalling a normal relationship between the producer and the consumer, which has been lost in the endless loopholes that farmers have to go through to sell food in our current system. The major huddle in achieving this is the problem of middlemen. We certainly cannot remove the middle men from the supply chain but we can restructure its function to improve efficiency of the system. Also the another issue of concern is how to increase farmers connect with organization and bring more farmers on community board.

2. OVERALL DESCRIPTION

Proposed Methodology:

"Fresh Basket," which specializes in delivering fresh groceries and produce to customers:

1. **Market Research and Requirement Analysis:** The initial step involves comprehensive market research to understand customer preferences, trends in online grocery shopping, and the specific needs of the target audience. Gathering insights from potential customers and stakeholders will help define the platform's core features and functionalities. This phase also includes identifying the types of groceries and produce to be offered, delivery regions, and any unique selling points.
2. **Design and Development:** With a clear understanding of requirements, the design and development phase begins. This involves crafting the architecture of the Fresh Basket platform, including databases, backend systems, and user interfaces. Designers create an intuitive user interface (UI) that ensures seamless navigation and an engaging shopping experience. Developers work on creating the necessary backend infrastructure, including inventory management, order processing, and payment integration. This phase also covers the development of a mobile app and website, ensuring responsiveness and compatibility across devices.
3. **Product Sourcing and Vendor Onboarding:** This stage focuses on establishing relationships with local farmers, suppliers, and vendors to ensure a steady supply of fresh produce and groceries. Effective communication channels and agreements are set up to maintain the quality and availability of products. Vendors are onboarded onto the platform, and their products are listed with accurate descriptions, images, and prices.
4. **Implementation:** Once the platform is developed and tested internally, it's time to implement it for public use. This involves deploying the platform on servers that can handle the expected traffic and usage. Security measures, such as data encryption and secure payment gateways, are implemented to protect customer information.
5. **Testing and Quality Assurance:** Rigorous testing is essential to ensure that Fresh Basket functions flawlessly and meets user expectations. Different types of testing, including functional testing (ensuring features work as intended), performance testing (assessing platform speed and responsiveness), and user acceptance testing (real users validating the platform), are conducted. Any identified issues are addressed to enhance the platform's reliability.

Operating Environment:

Server Side:

Processor: Intel Core i3 5th Generation

HDD: Minimum 500GB Disk Space

RAM: Minimum 4GB

OS: Windows 10

Database: MySQL

Client Side (minimum requirement):

Processor: Intel Dual Core

HDD: Minimum 80GB Disk Space

RAM: Minimum 2GB

OS: Windows 7

Design and Implementation Constraints:

- The application will use Spring-Boot and React as main web technologies.
- HTTP protocol is used as communication protocol. FTP is used to upload the web application in live domain and the client can access it via HTTP protocol.
- Several types of validations make this web application a secured one and SQL Injections can also be prevented.
- SMTP protocol is used for email communication.

3. REQUIREMENTS SPECIFICATION

External Interface Requirements:

User Interfaces:

- All the users will see the same page when they enter in this website. This page asks the users a username and a password.
- After being authenticated by correct email and password, user will be redirect to their corresponding profile where they can do various activities.
- The user interface will be simple and consistence, using terminology commonly understood by intended users of the system. The system will have simple interface, consistence with standard interface, to eliminate need for user training of infrequent users.

Hardware Interfaces:

- No extra hardware interfaces are needed.
- The system will use the standard hardware and data communication resources.

This includes, but not limited to, general network connection at the server/hosting site, network server and network management tools.

Application Interfaces:

Web Browser:

The system is a web-based application; clients need a modern web browser such as Mozilla Firebox, Internet Explorer, Opera, and Chrome. The computer must have an Internet connection in order to be able to access the system.

Communications Interfaces:

- This system uses communication resources which includes but not limited to, HTTP protocol for communication with the web browser and web server and TCP/IP network protocol with HTTP protocol.
- This application will communicate with the database that holds all the user information. Users can contact with server side through HTTP protocol by means of a function that is called HTTP Service. This function allows the application to use the data retrieved by server to fulfill the request fired by the user.

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 admin can delete a farmer account if they need to, for any purpose.
- Admin 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 Customer 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.

NON-FUNCTIONAL REQUIREMENTS :

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

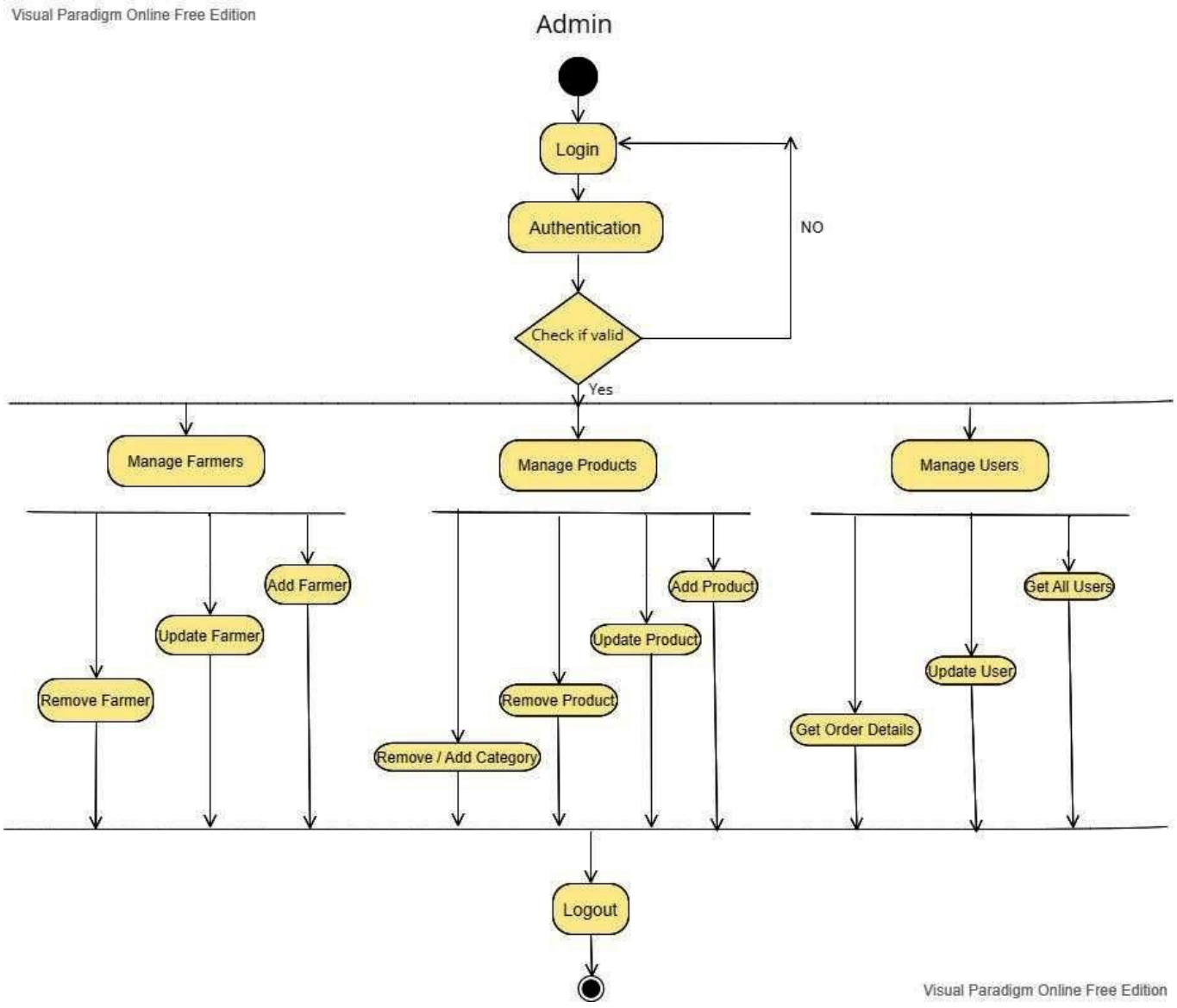
- Security
  - The system's back-end servers shall only be accessible to authenticated administrators. Sensitive data will be encrypted before being sent over insecure connections like the internet.
- Availability
  - The system should be available at all times, meaning the user can access it using a web browser, only restricted by the downtime of the server on which the system runs. In case of an of a hardware failure or database corruption, a replacement page will be shown. Also, in case of a hardware failure or database corruption, backups of the database should be retrieved from the server and saved by the administrator. Then the service will be restarted. It means 24 x 7 availability.

- Reliability
  - The reliability of the overall program depends on the reliability of the separate components. The main pillar of the reliability of the system is the backup of the database which is continuously maintained and updated to reflect the most recent changes. Thus, the overall stability of the system depends on the stability of container and its underlying operating system.
- Maintainability
  - A commercial database is used for maintaining the database and the application server takes care of the site. In case of a failure, a re-initialization of the program will be done. Also, the software design is being done with modularity in mind so that maintainability can be done efficiently.
- Accessibility
  - The system will be a web-based application it is going to be accessible on the web browser.
- Backup
  - We will take a backup in our system database. In order to enable the administrator and the user to access the data from our system.
- Performance
  - The product shall be based on web and has to be run from a web server. The product shall take initial load time depending on internet connection strength which also depends on the media from which the product is run. The performance shall depend upon hardware components of the client/customer.
- Supportability
  - The source code developed for this system shall be maintained in configuration management tool.

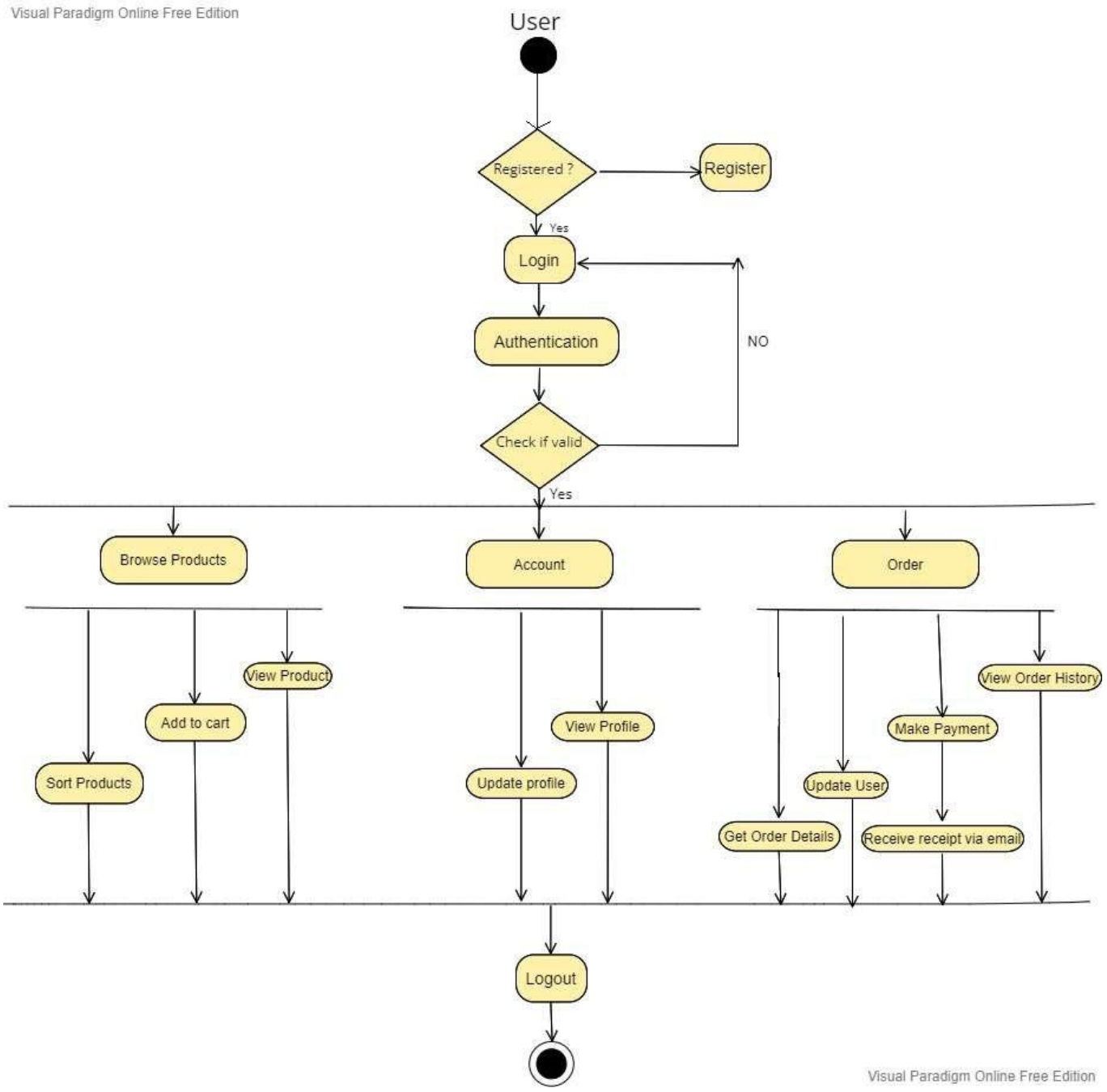
4. SYSTEM DIAGRAMS

- **Activity Diagram:**

  - Admin Activity:



- User Activity:



- **Data Flow diagram:**

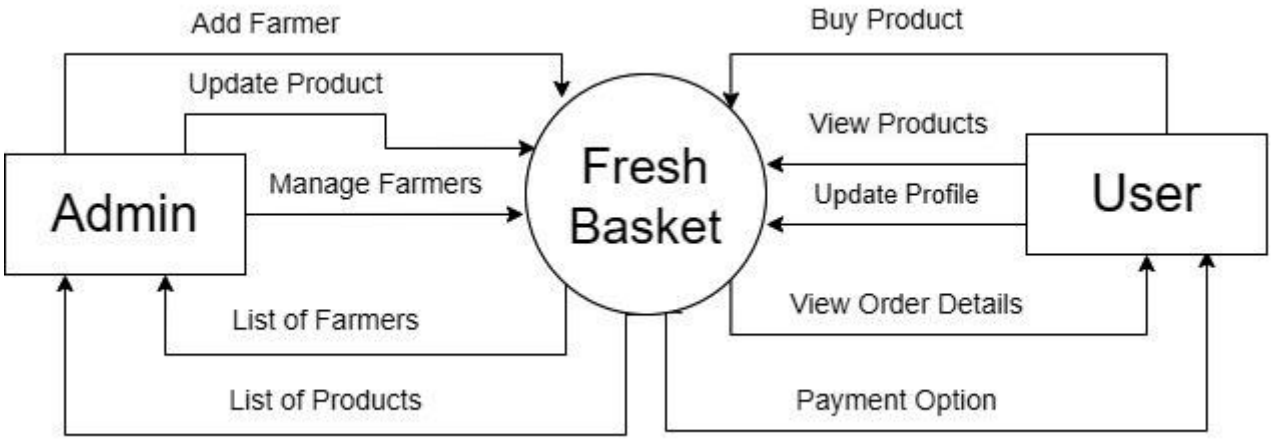


Figure: Level 0 DFD

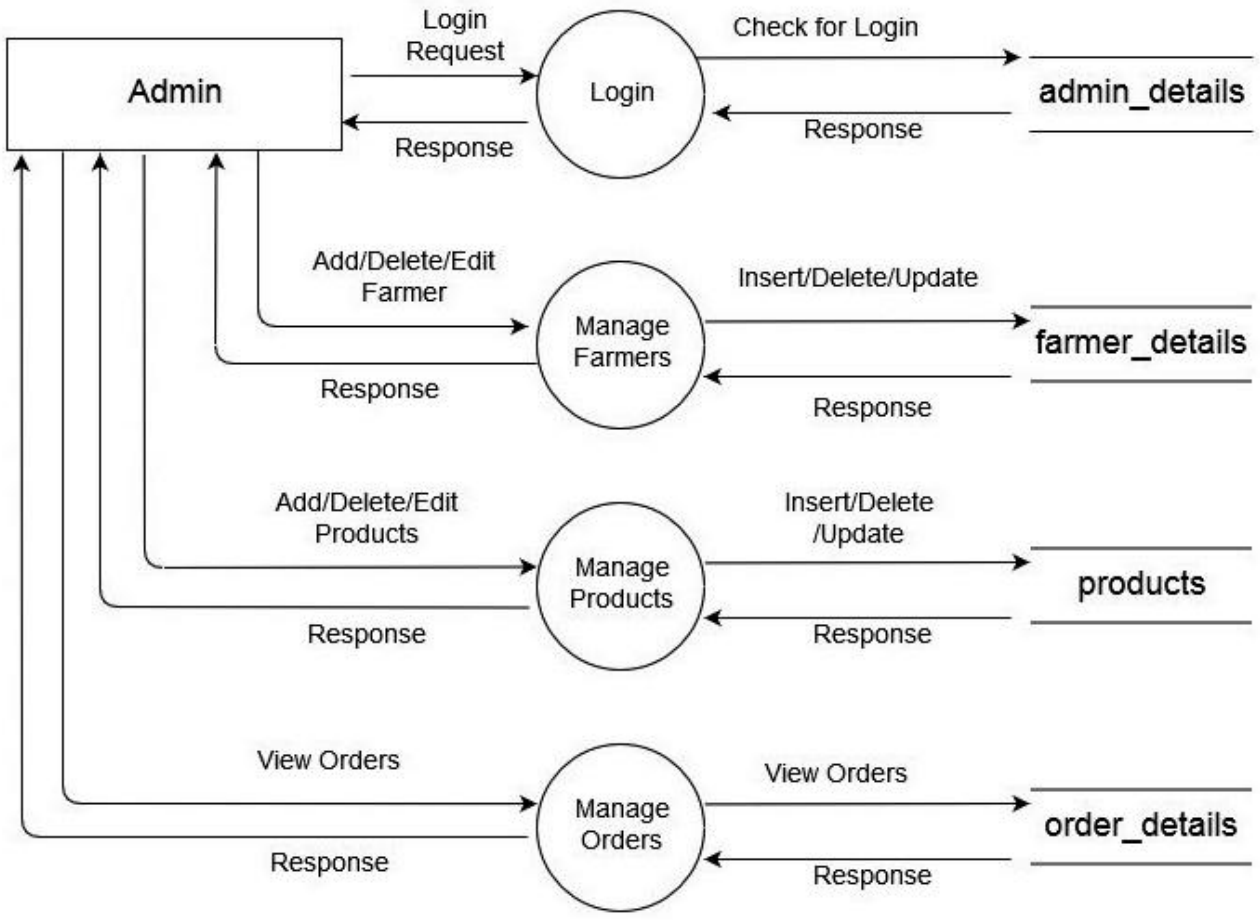


Figure: Level 1 Admin DFD

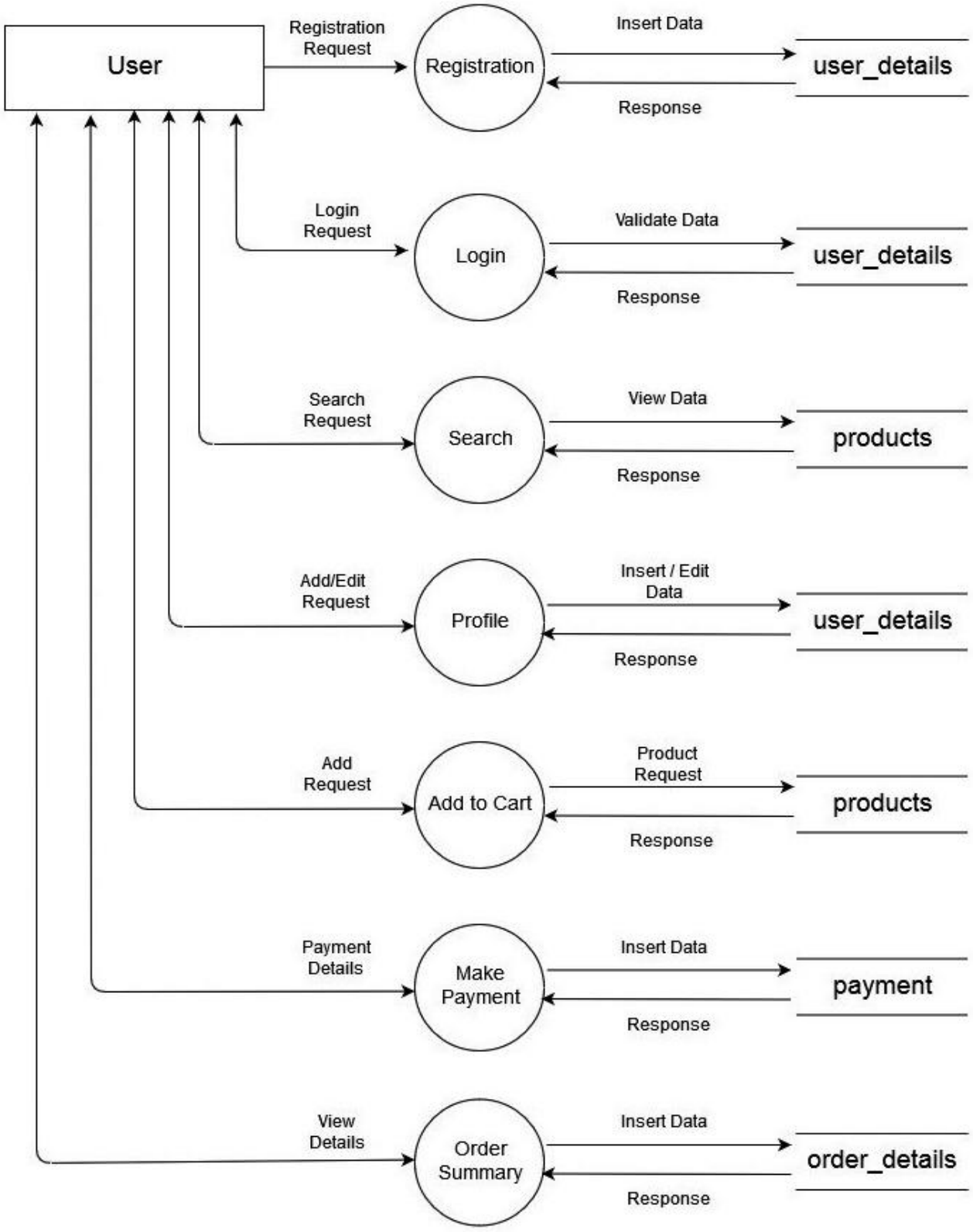
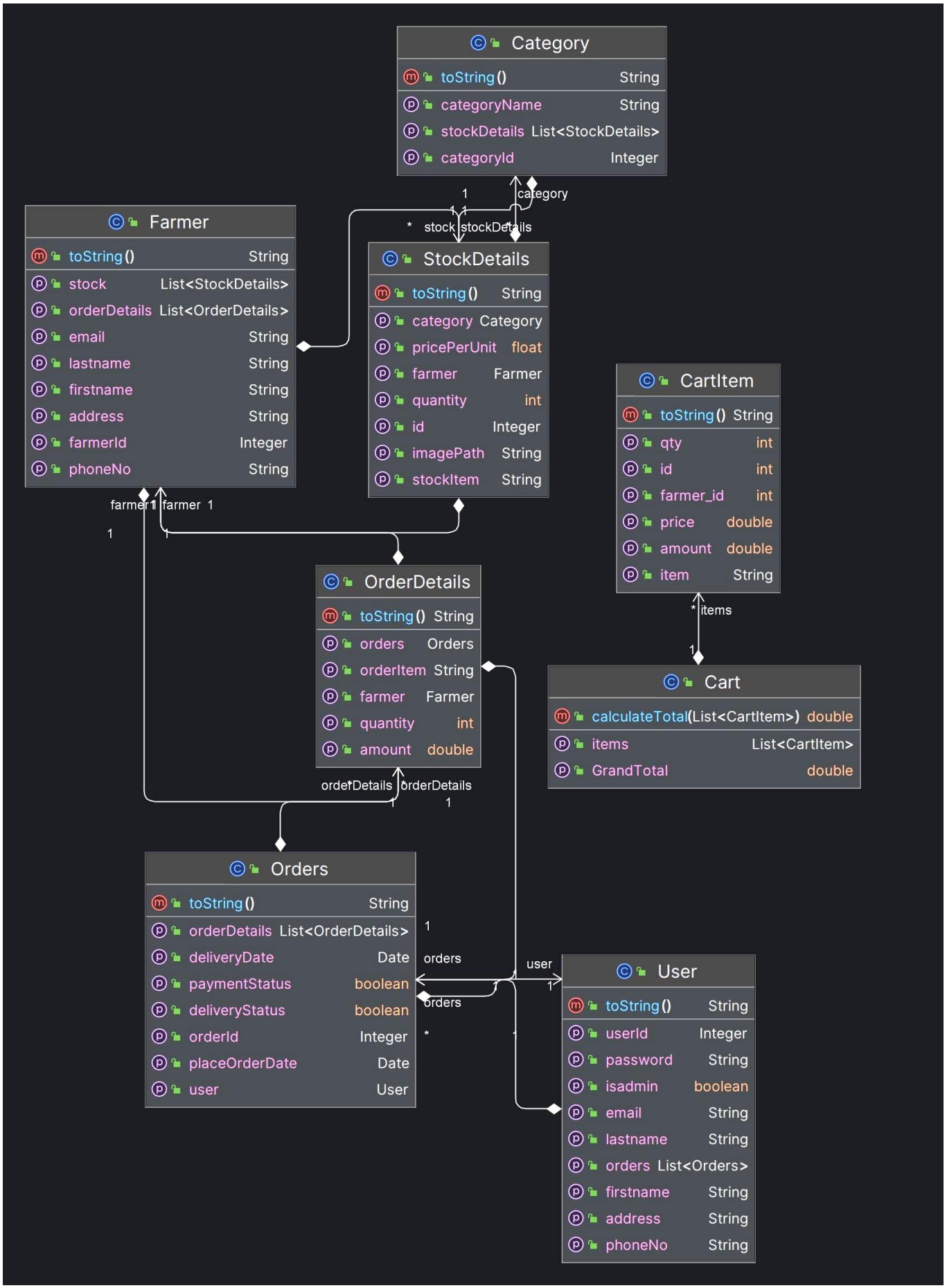


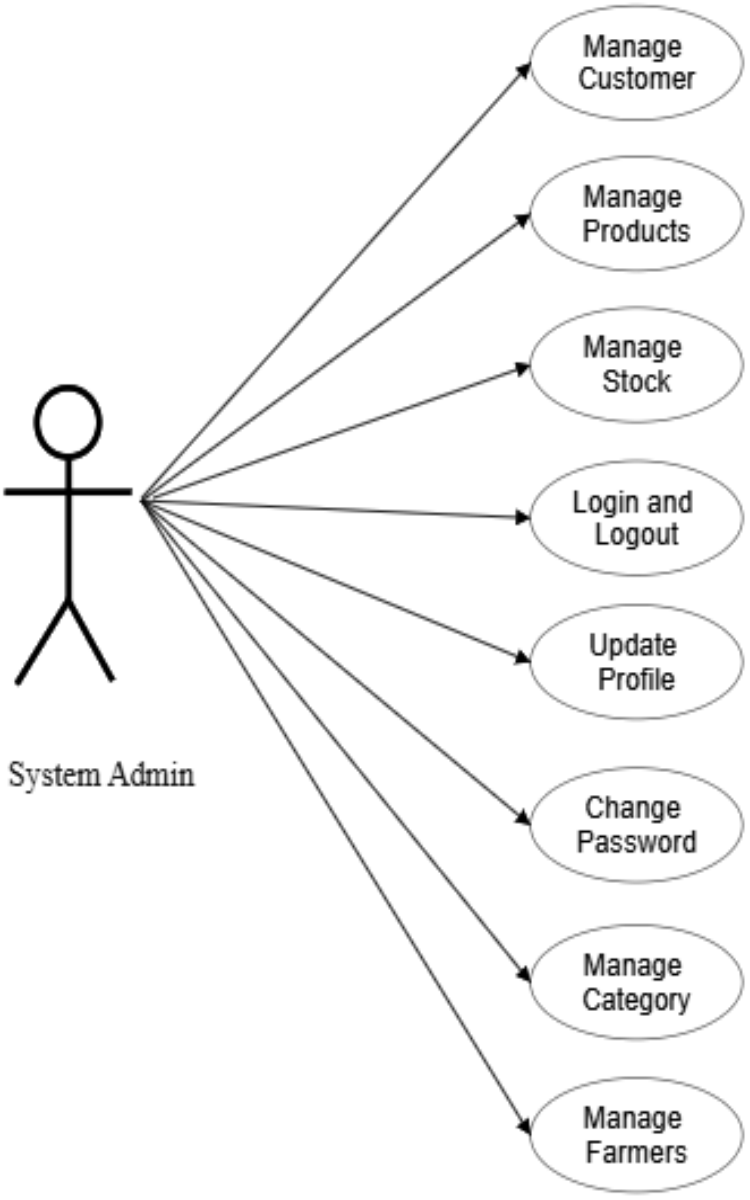
Figure : Level 1 User DFD

- **Class Diagram:**

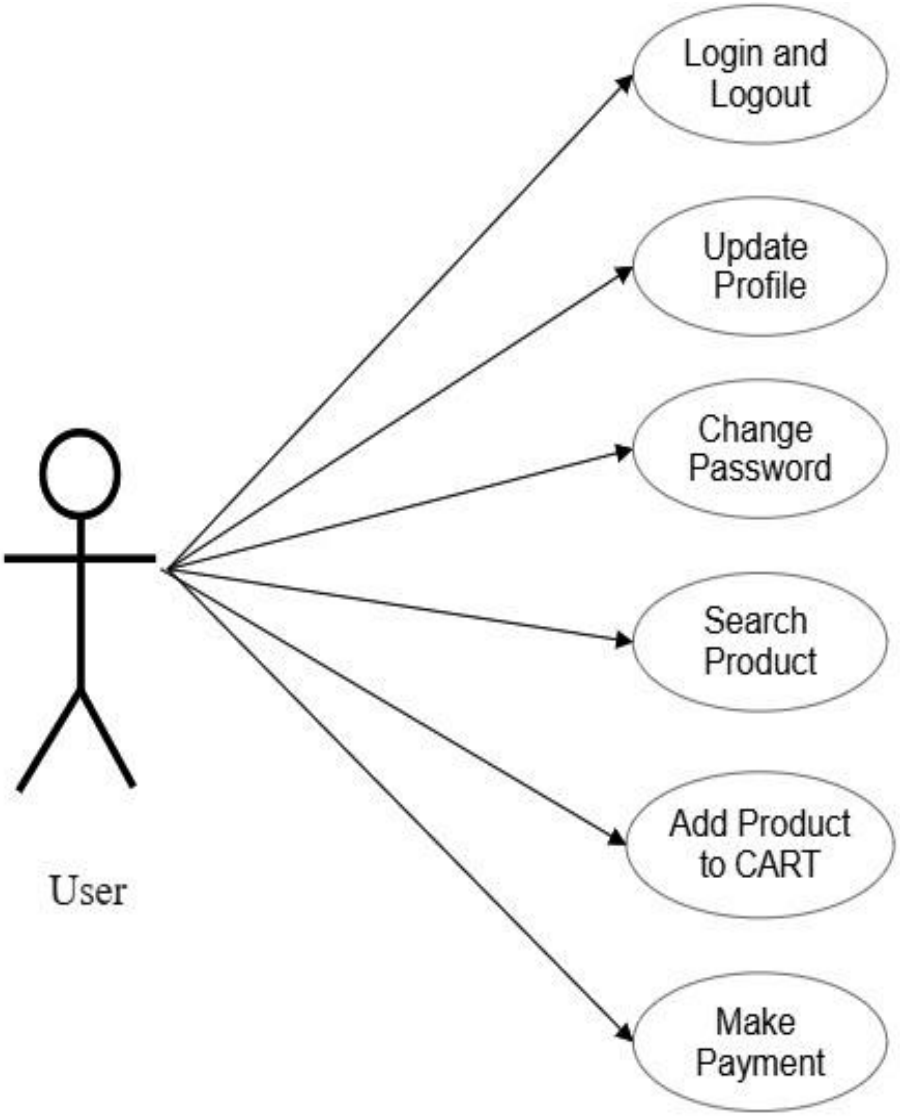


- **Use Case Diagram:**

  - admin Use-case:



- User Use-case:



- **ER Diagram:**

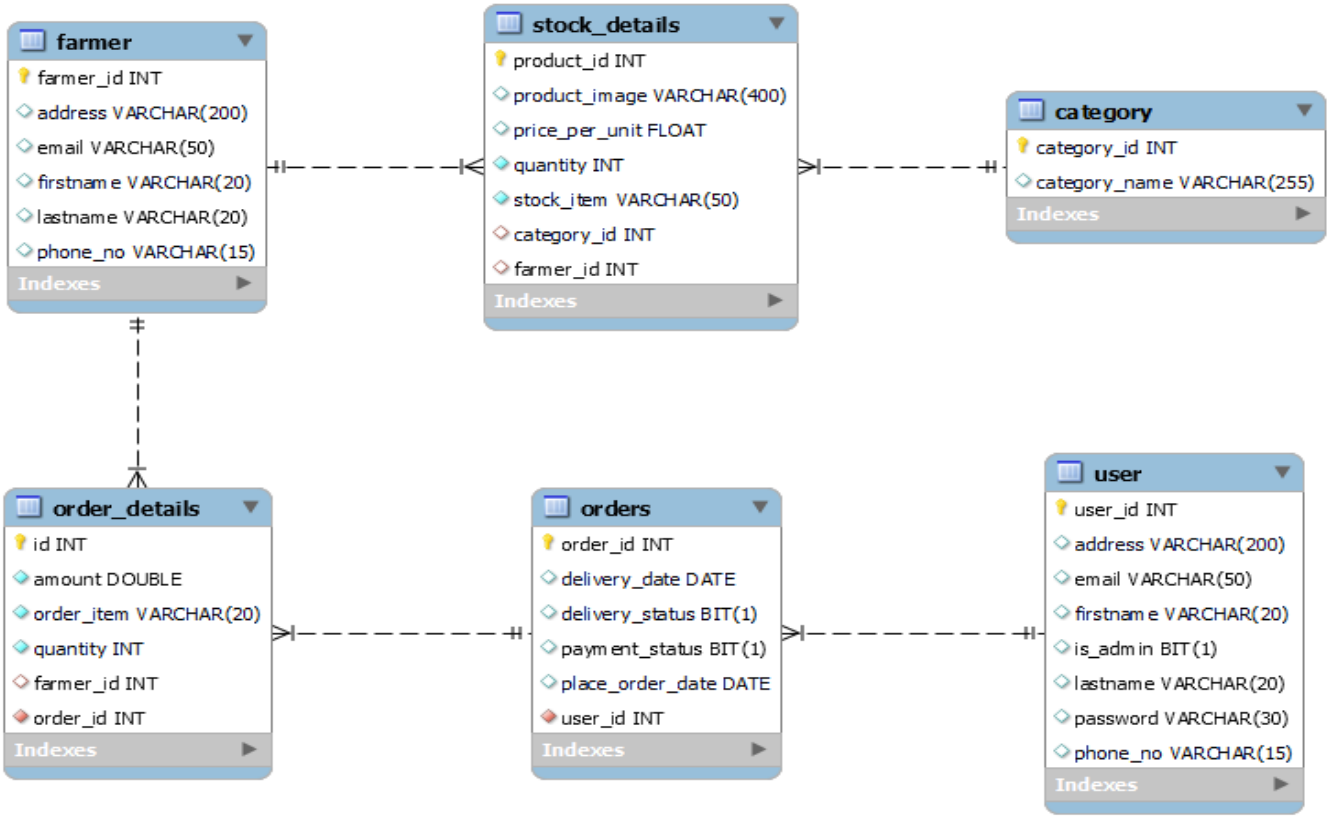
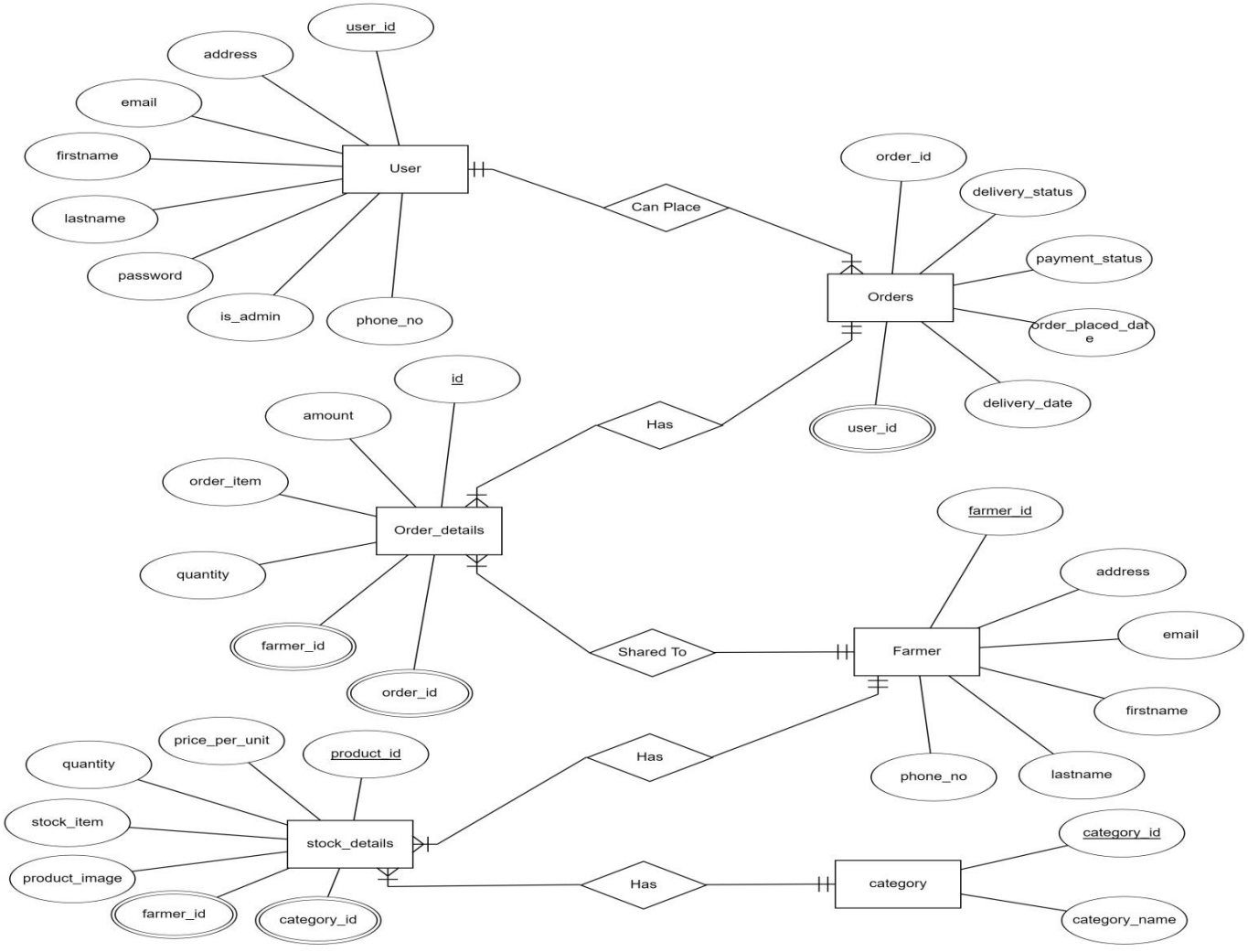
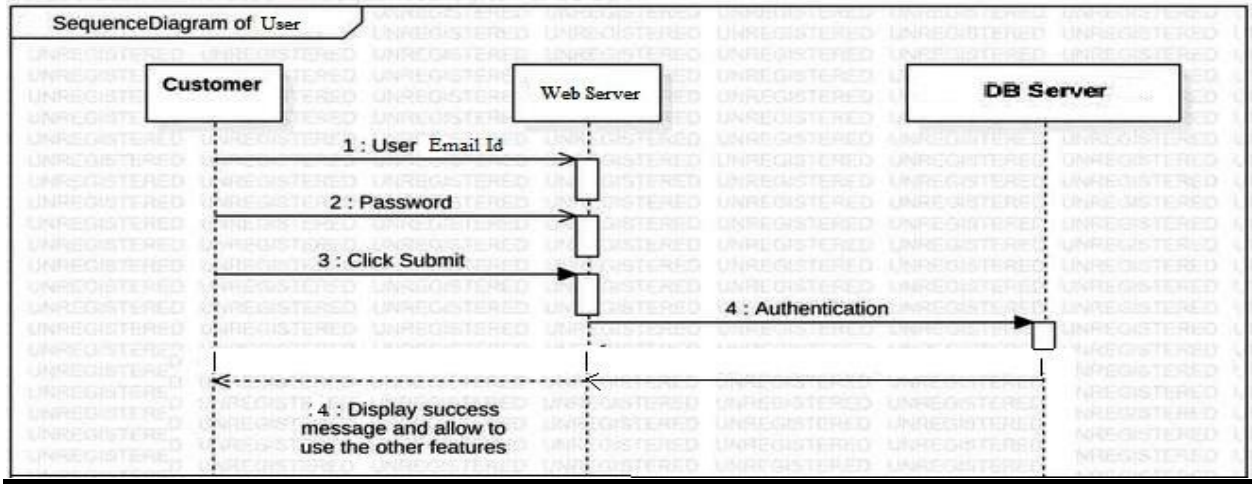


Figure: System generated ER diagram



- **Sequence Diagram:**



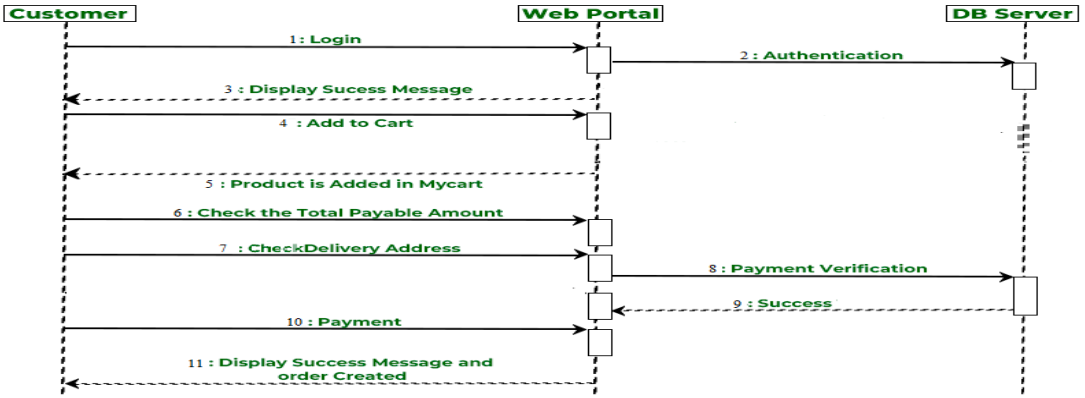


Figure: User Side Sequence Diagram

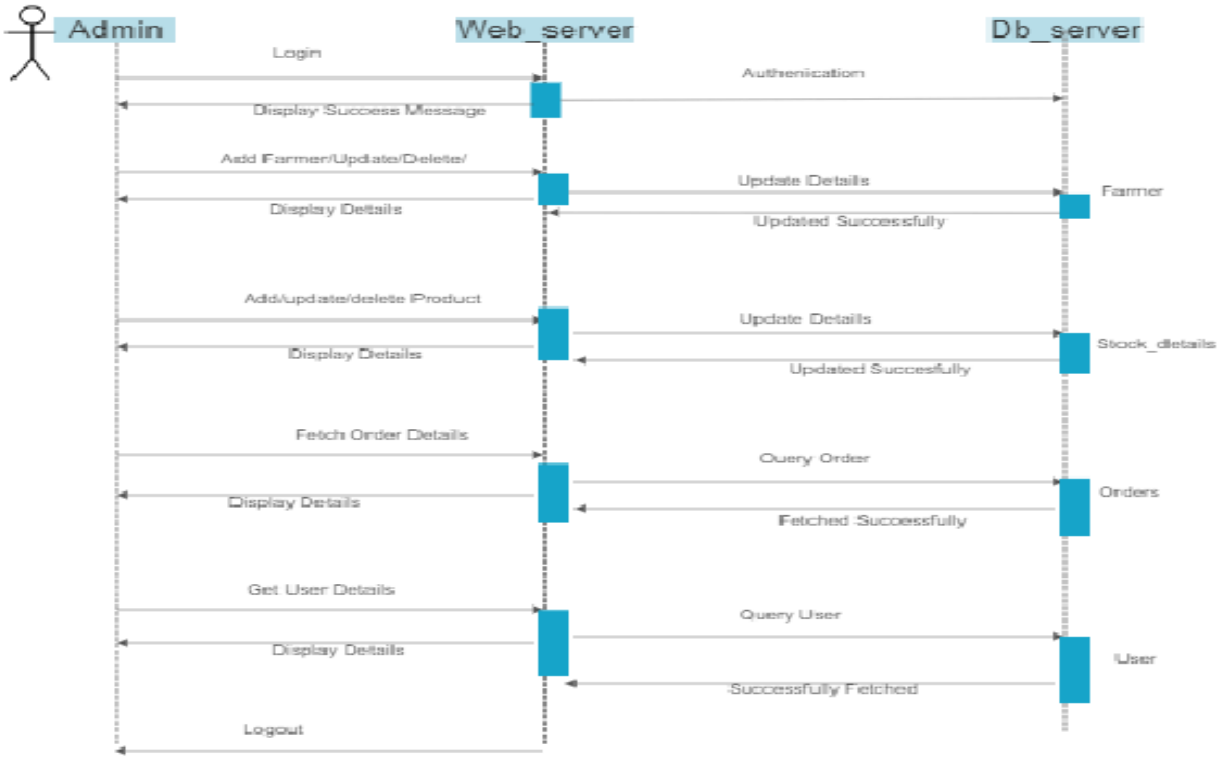


Figure: Admin Side Sequence Diagram

5. TABLE STRUCTURE

- User:

```
mysql> desc user;
+-----------+--------------+------+-----+---------+----------------+
| Field     | Type         | Null | Key | Default | Extra          |
+-----------+--------------+------+-----+---------+----------------+
| user_id   | int          | NO   | PRI | NULL    | auto_increment |
| address   | varchar(200) | YES  |     | NULL    |                |
| email     | varchar(50)  | YES  | UNI | NULL    |                |
| firstname | varchar(20)  | YES  |     | NULL    |                |
| is_admin  | bit(1)       | YES  |     | NULL    |                |
| lastname  | varchar(20)  | YES  |     | NULL    |                |
| password  | varchar(30)  | YES  |     | NULL    |                |
| phone_no  | varchar(15)  | YES  |     | NULL    |                |
+-----------+--------------+------+-----+---------+----------------+
8 rows in set (0.07 sec)
```

- Farmer:

```
mysql> desc farmer;
+-----------+--------------+------+-----+---------+----------------+
| Field     | Type         | Null | Key | Default | Extra          |
+-----------+--------------+------+-----+---------+----------------+
| farmer_id | int          | NO   | PRI | NULL    | auto_increment |
| address   | varchar(200) | YES  |     | NULL    |                |
| email     | varchar(50)  | YES  |     | NULL    |                |
| firstname | varchar(20)  | YES  |     | NULL    |                |
| lastname  | varchar(20)  | YES  |     | NULL    |                |
| phone_no  | varchar(15)  | YES  | UNI | NULL    |                |
+-----------+--------------+------+-----+---------+----------------+
6 rows in set (0.00 sec)
```

- Category:

```
mysql> desc category;
+---------------+--------------+------+-----+---------+----------------+
| Field         | Type         | Null | Key | Default | Extra          |
+---------------+--------------+------+-----+---------+----------------+
| category_id   | int          | NO   | PRI | NULL    | auto_increment |
| category_name | varchar(255) | YES  | UNI | NULL    |                |
+---------------+--------------+------+-----+---------+----------------+
2 rows in set (0.00 sec)
```

- Stock Details:

```
mysql> desc stock_details;
+---------------+-------------+------+-----+---------+----------------+
| Field         | Type        | Null | Key | Default | Extra          |
+---------------+-------------+------+-----+---------+----------------+
| product_id    | int         | NO   | PRI | NULL    | auto_increment |
| product_image | varchar(400)| YES  |     | NULL    |                |
| price_per_unit| float       | YES  |     | NULL    |                |
| quantity      | int         | NO   |     | NULL    |                |
| stock_item    | varchar(50) | NO   | UNI | NULL    |                |
| category_id   | int         | YES  | MUL | NULL    |                |
| farmer_id     | int         | YES  | MUL | NULL    |                |
+---------------+-------------+------+-----+---------+----------------+
7 rows in set (0.06 sec)
```

- Orders:

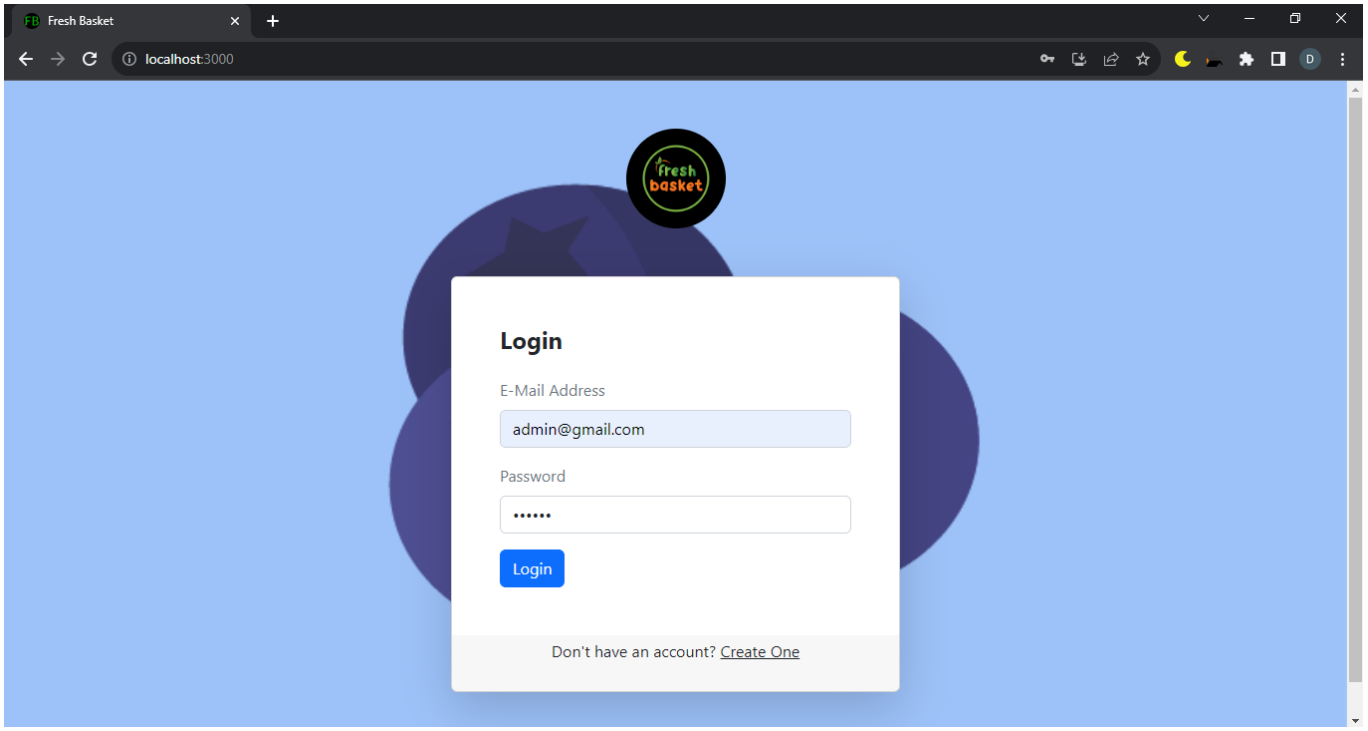
```
mysql> desc orders;
+------------------+--------+------+-----+---------+----------------+
| Field            | Type   | Null | Key | Default | Extra          |
+------------------+--------+------+-----+---------+----------------+
| order_id         | int    | NO   | PRI | NULL    | auto_increment |
| delivery_date    | date   | YES  |     | NULL    |                |
| delivery_status  | bit(1) | YES  |     | NULL    |                |
| payment_status   | bit(1) | YES  |     | NULL    |                |
| place_order_date | date   | YES  |     | NULL    |                |
| user_id          | int    | NO   | MUL | NULL    |                |
+------------------+--------+------+-----+---------+----------------+
6 rows in set (0.00 sec)
```

- Order Details:

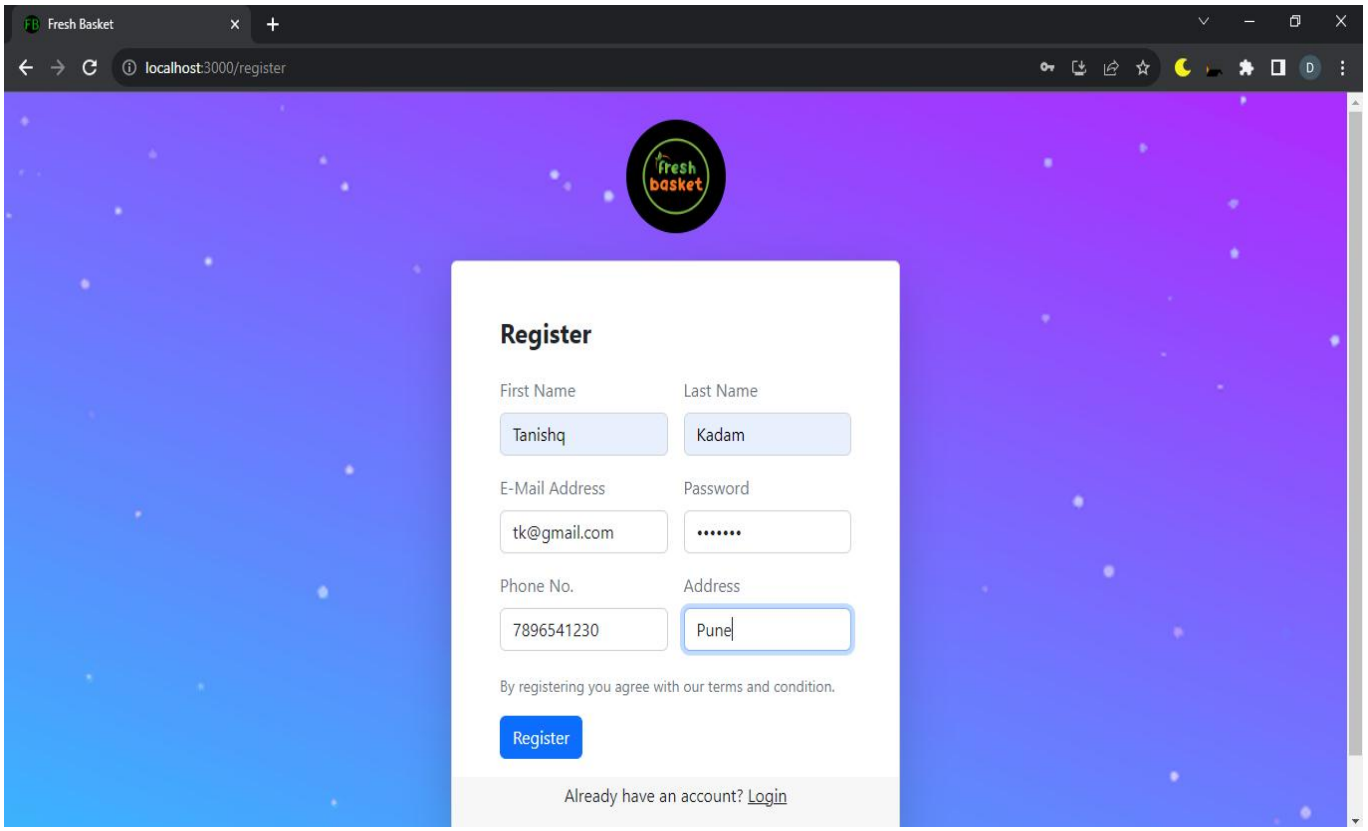
```
mysql> desc order_details;
+------------+-------------+------+-----+---------+----------------+
| Field      | Type        | Null | Key | Default | Extra          |
+------------+-------------+------+-----+---------+----------------+
| id         | int         | NO   | PRI | NULL    | auto_increment |
| amount     | double      | NO   |     | NULL    |                |
| order_item | varchar(20) | NO   |     | NULL    |                |
| quantity   | int         | NO   |     | NULL    |                |
| farmer_id  | int         | YES  | MUL | NULL    |                |
| order_id   | int         | NO   | MUL | NULL    |                |
+------------+-------------+------+-----+---------+----------------+
6 rows in set (0.00 sec)
```

6. UI SCREENSHOTS

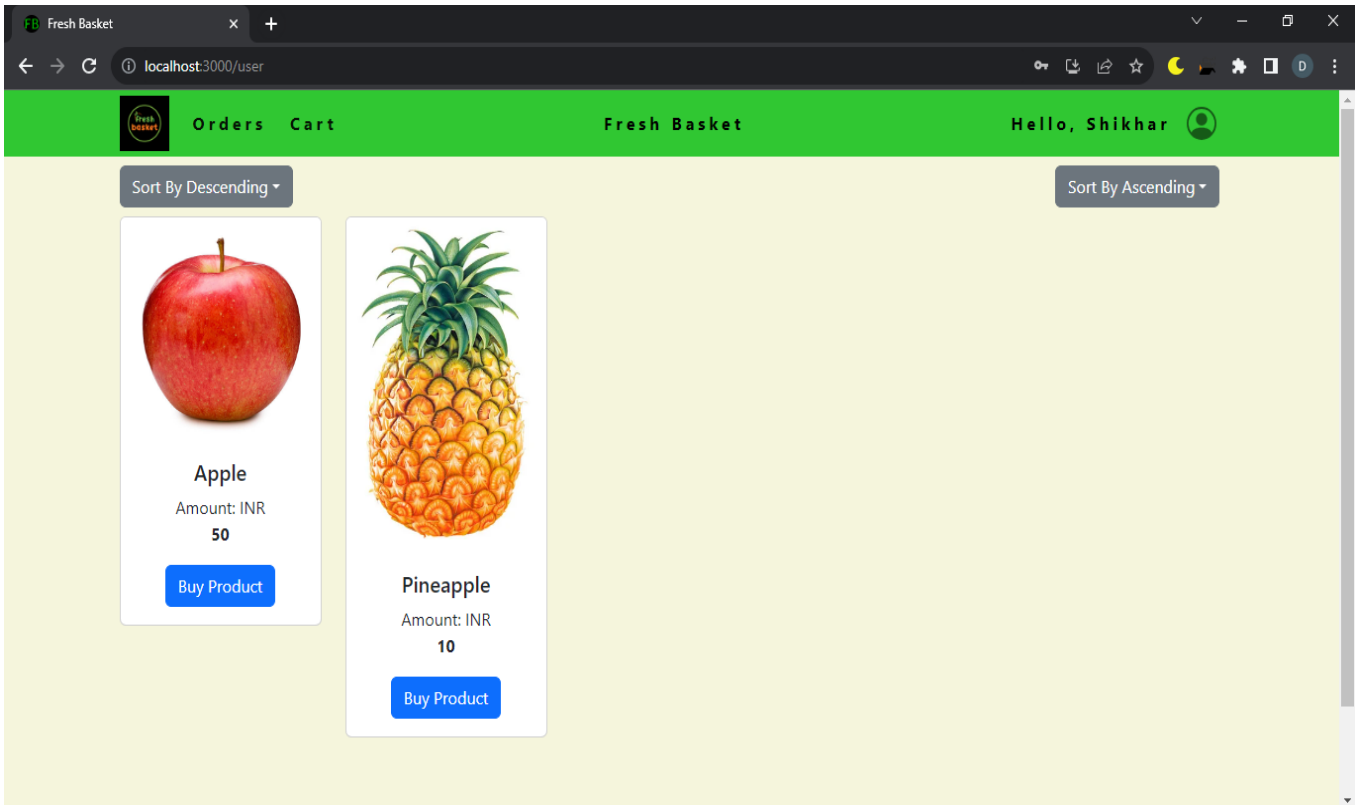
- **LOGIN PAGE:**



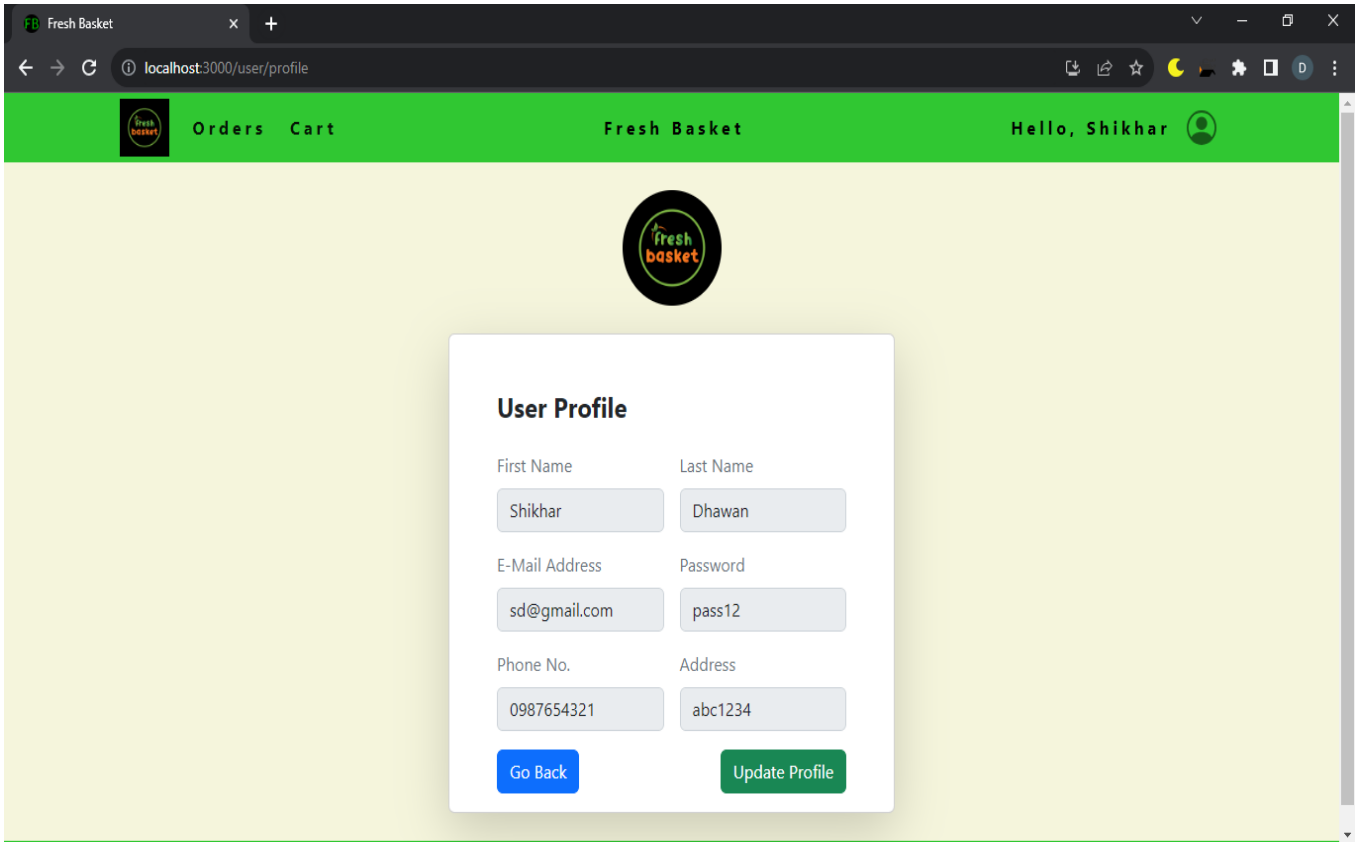
- **REGISTERATION PAGE:**



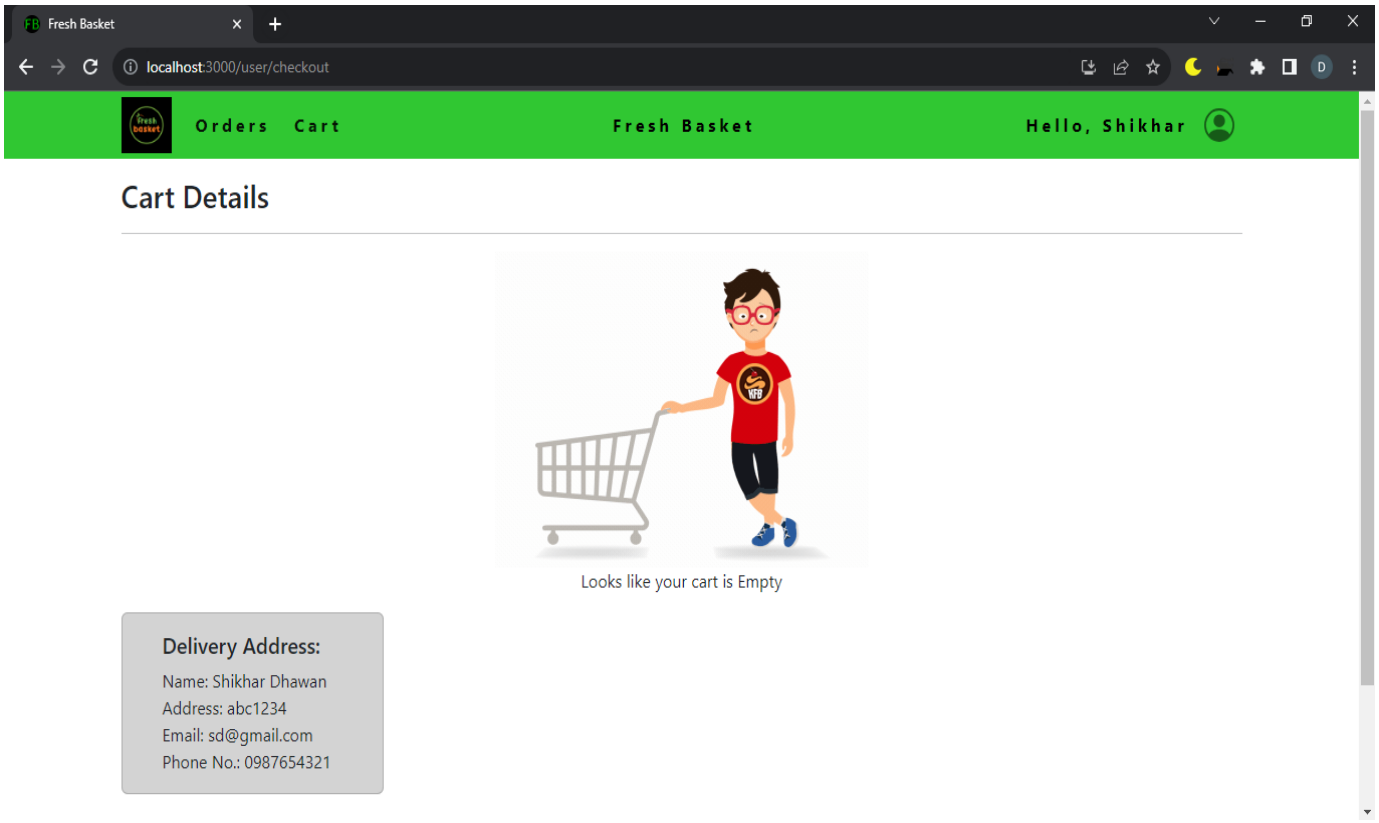
- **HOME PAGE:**



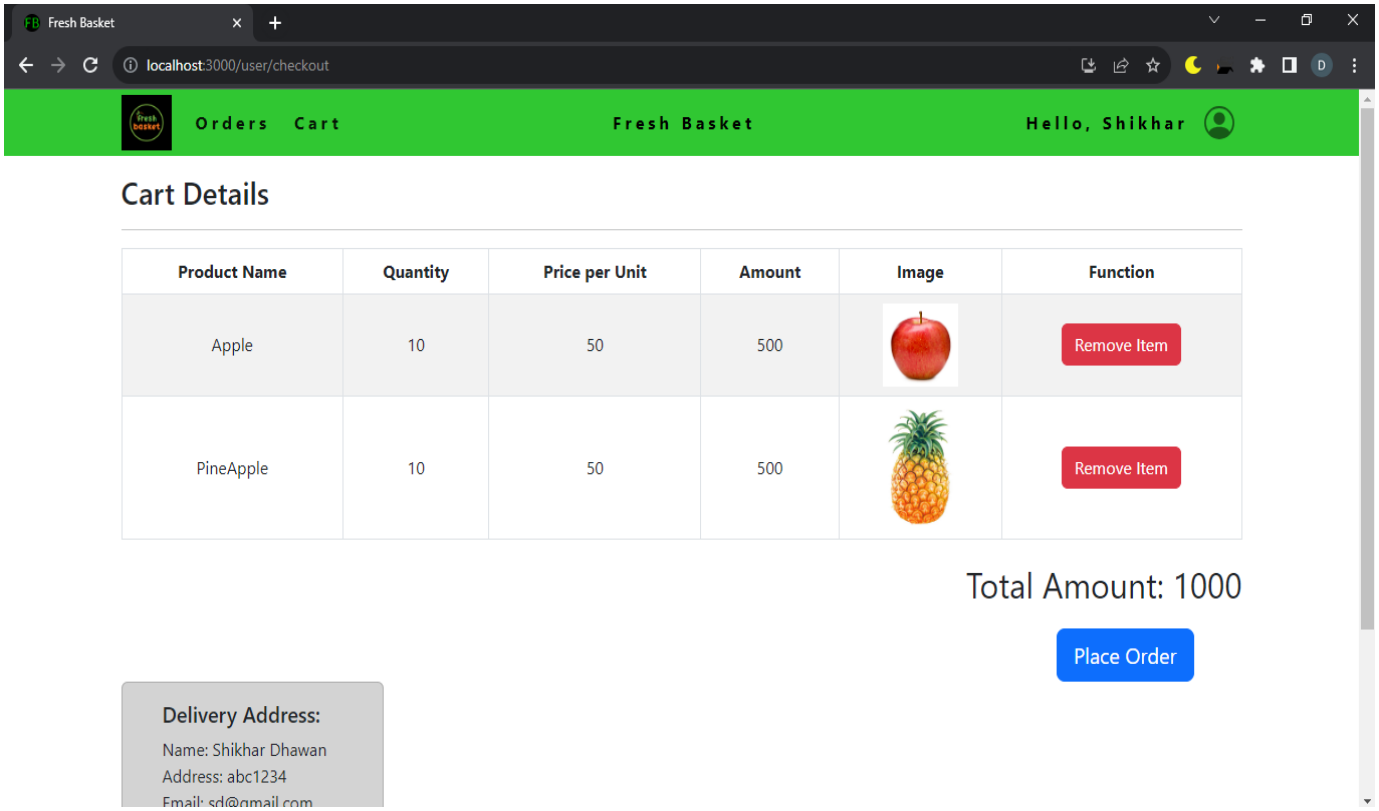
- **USER PROFILE:**



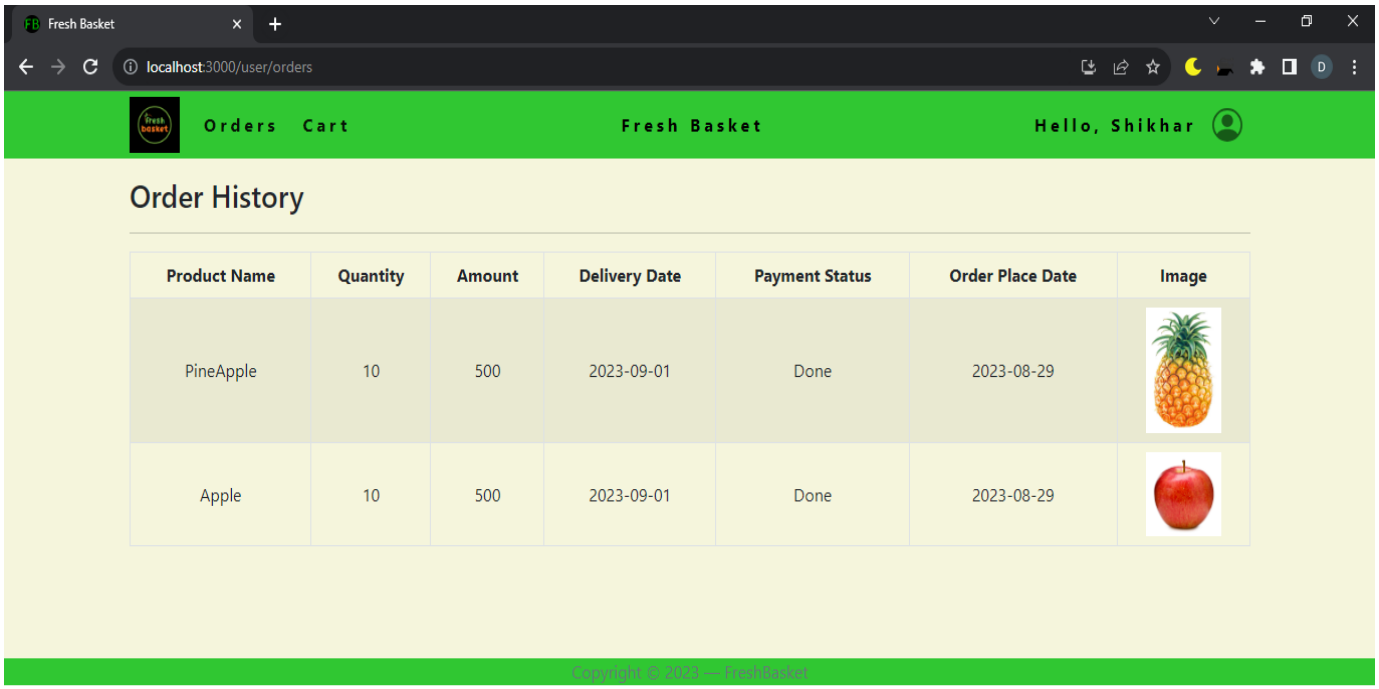
- **EMPTY CART:**



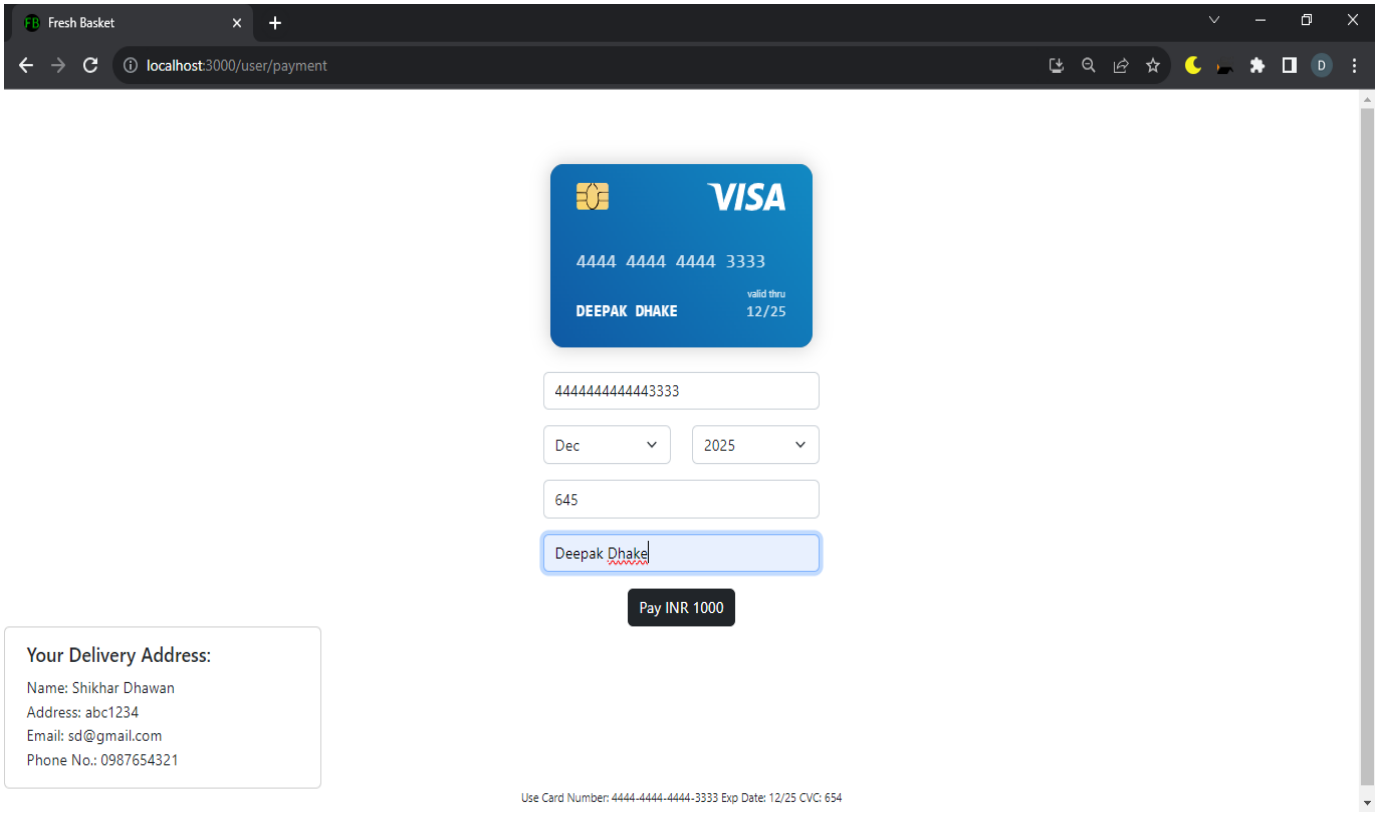
- **USER CART:**



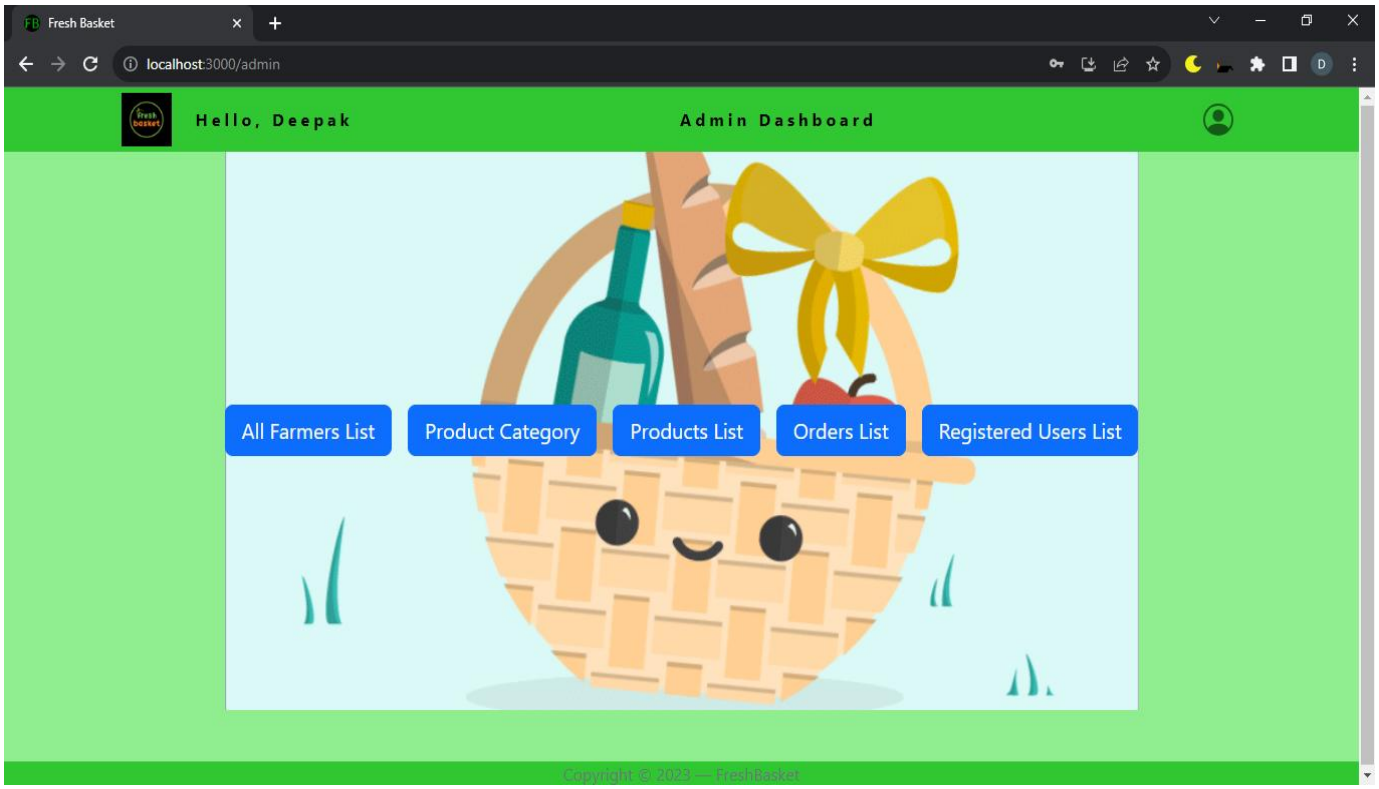
- **ORDER HISTORY:**



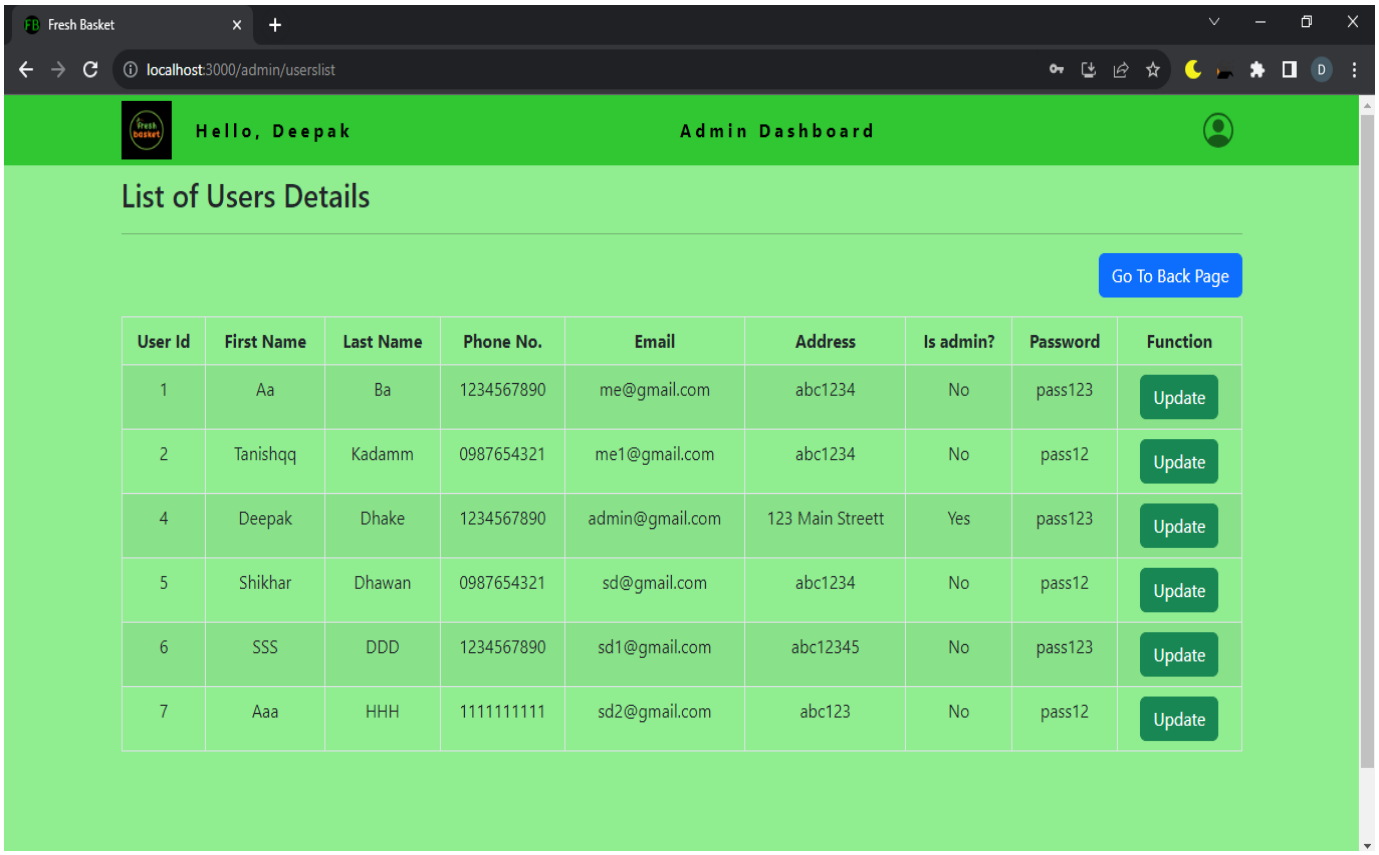
- **PAYMENT GATEWAY:**



- **ADMIN DASHBOARD:**

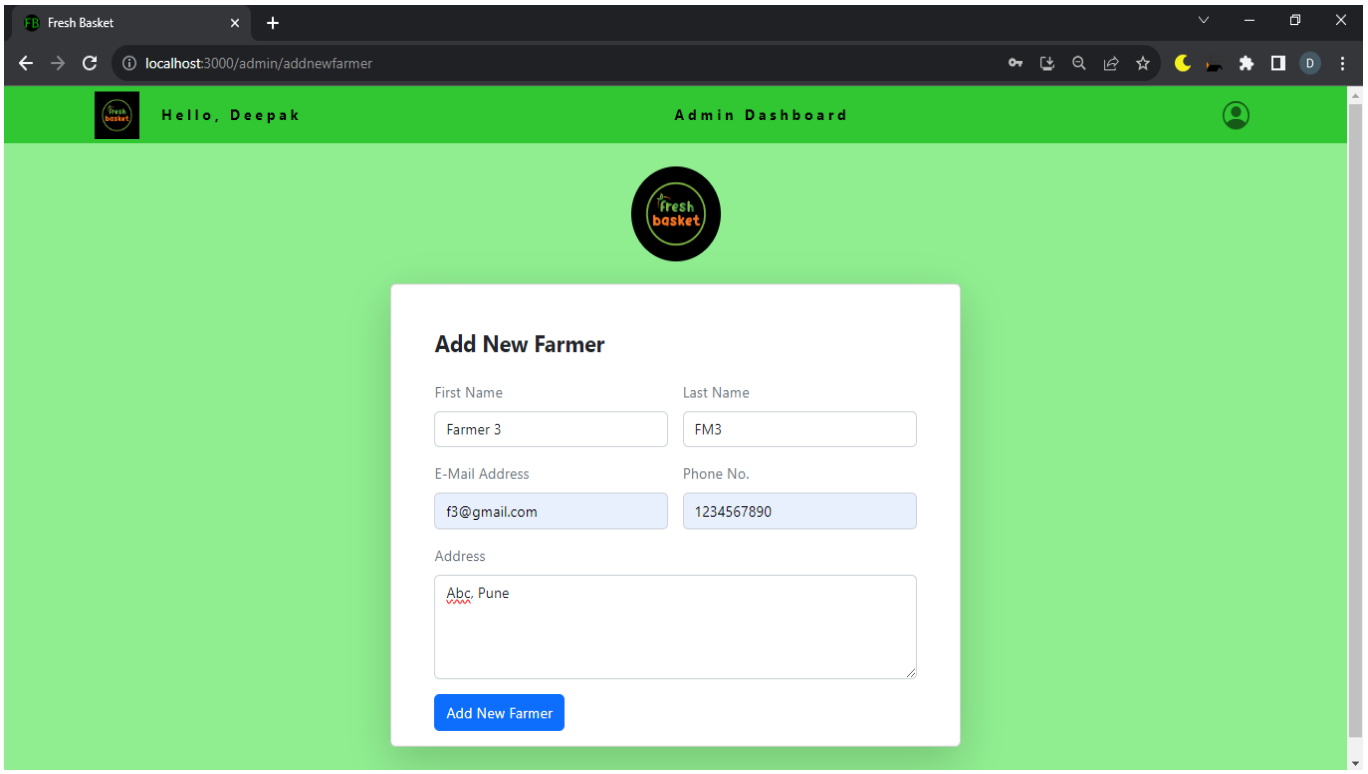


- **VIEW ALL USERS:**

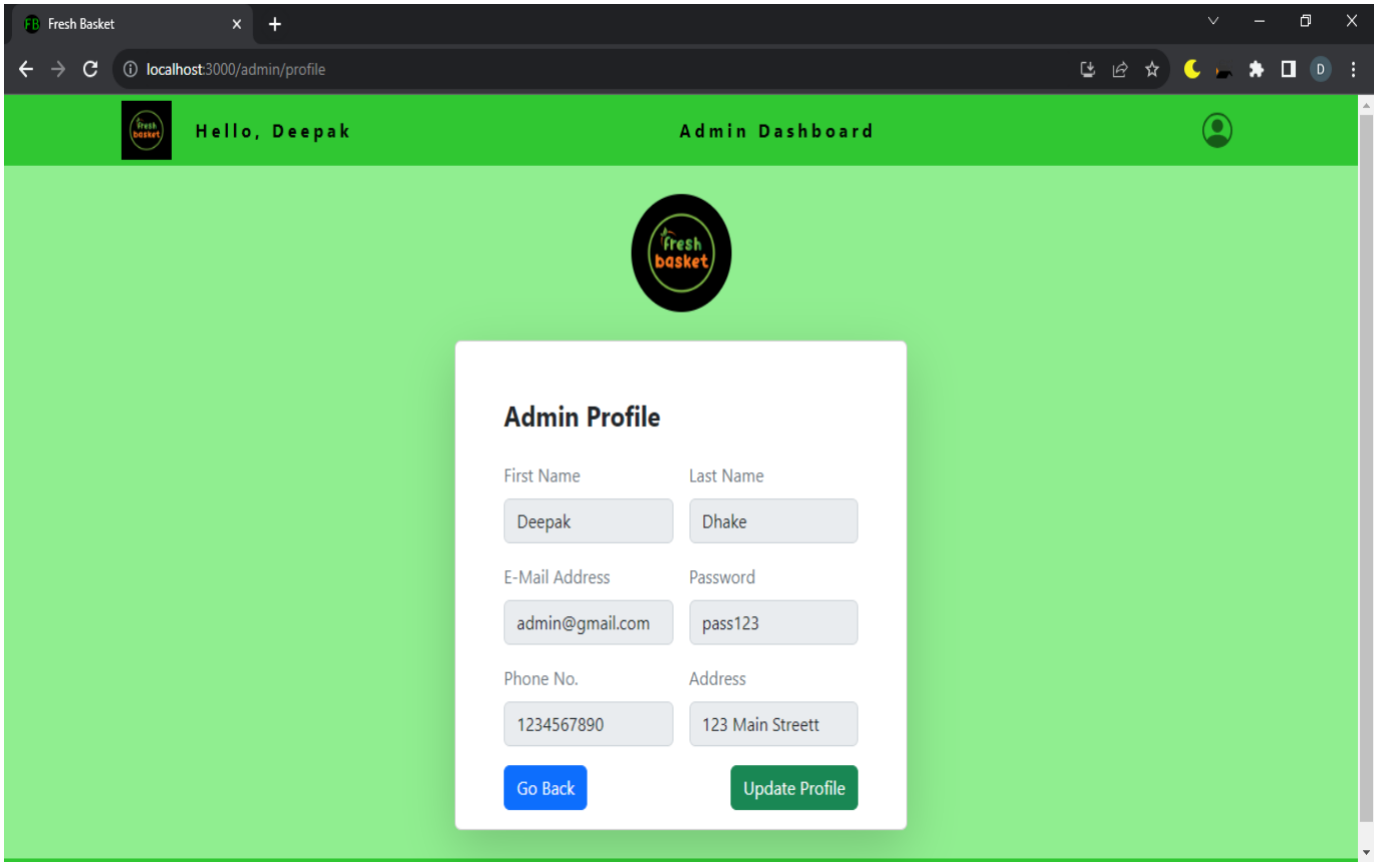


User Id	First Name	Last Name	Phone No.	Email	Address	Is admin?	Password	Function
1	Aa	Ba	1234567890	me@gmail.com	abc1234	No	pass123	Update
2	Tanishqq	Kadamm	0987654321	me1@gmail.com	abc1234	No	pass12	Update
4	Deepak	Dhake	1234567890	admin@gmail.com	123 Main Streett	Yes	pass123	Update
5	Shikhar	Dhawan	0987654321	sd@gmail.com	abc1234	No	pass12	Update
6	SSS	DDD	1234567890	sd1@gmail.com	abc12345	No	pass123	Update
7	Aaa	HHH	1111111111	sd2@gmail.com	abc123	No	pass12	Update

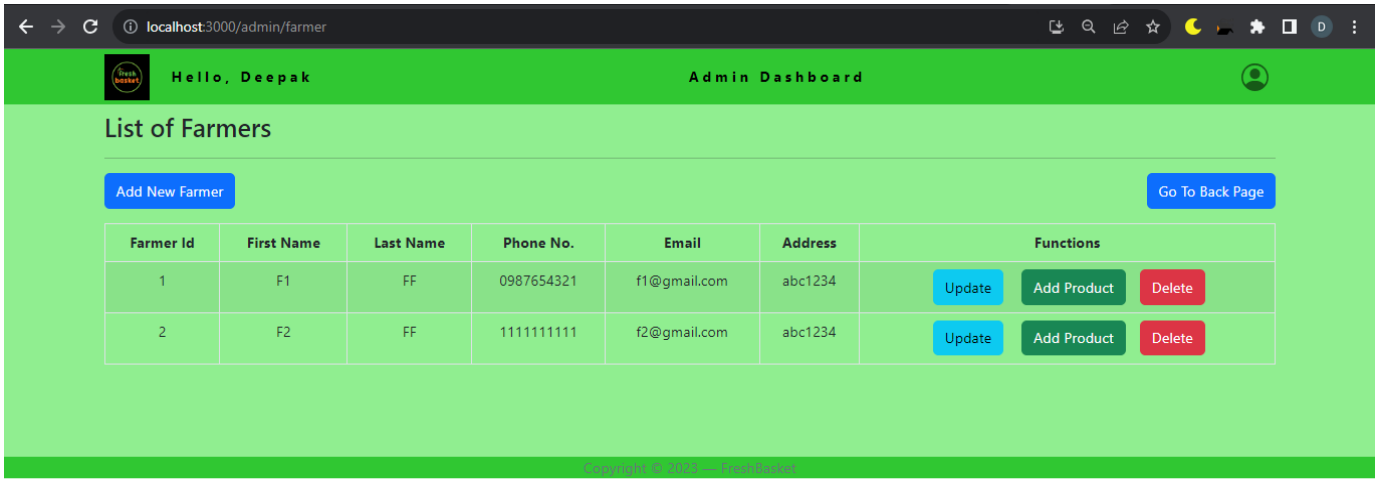
- **ADD NEW FARMER:**



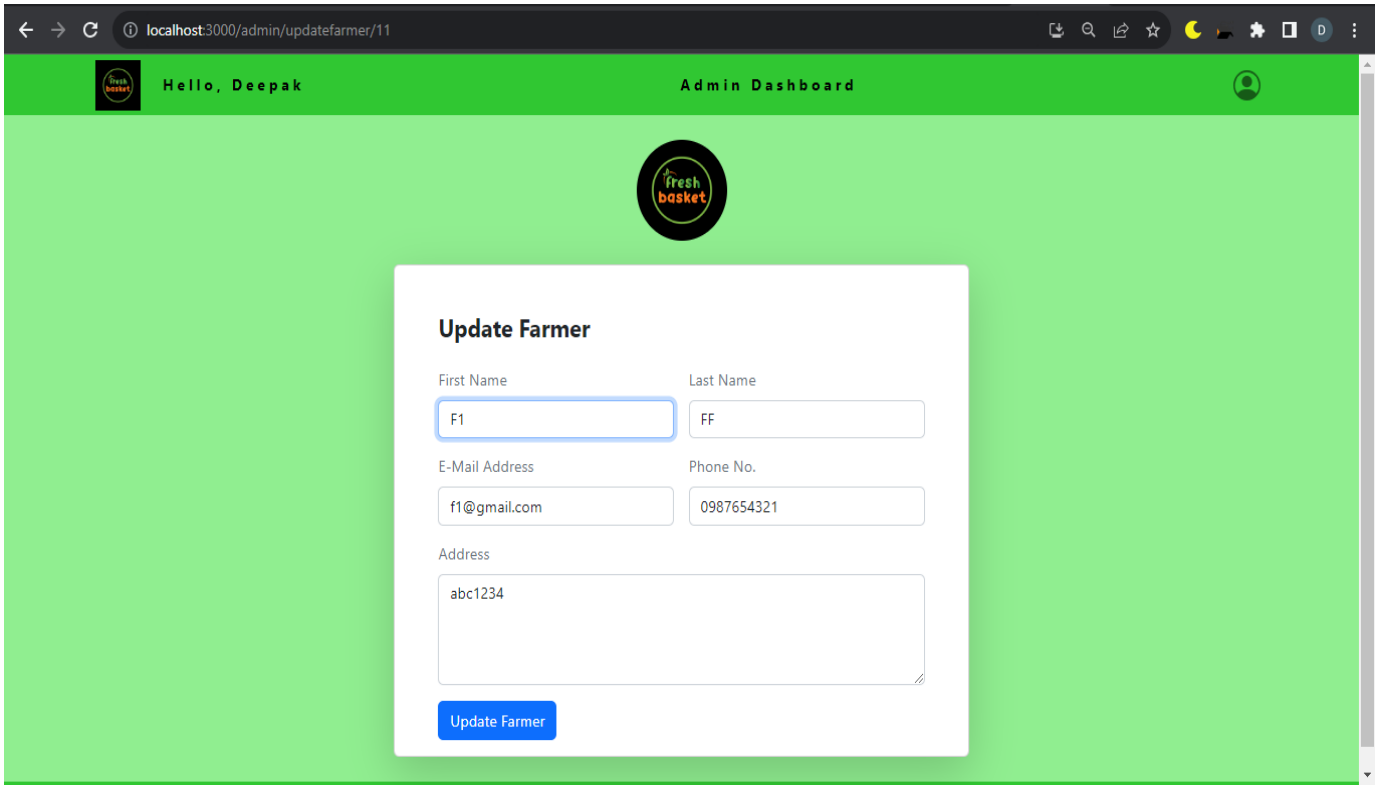
- **ADMIN PROFILE:**



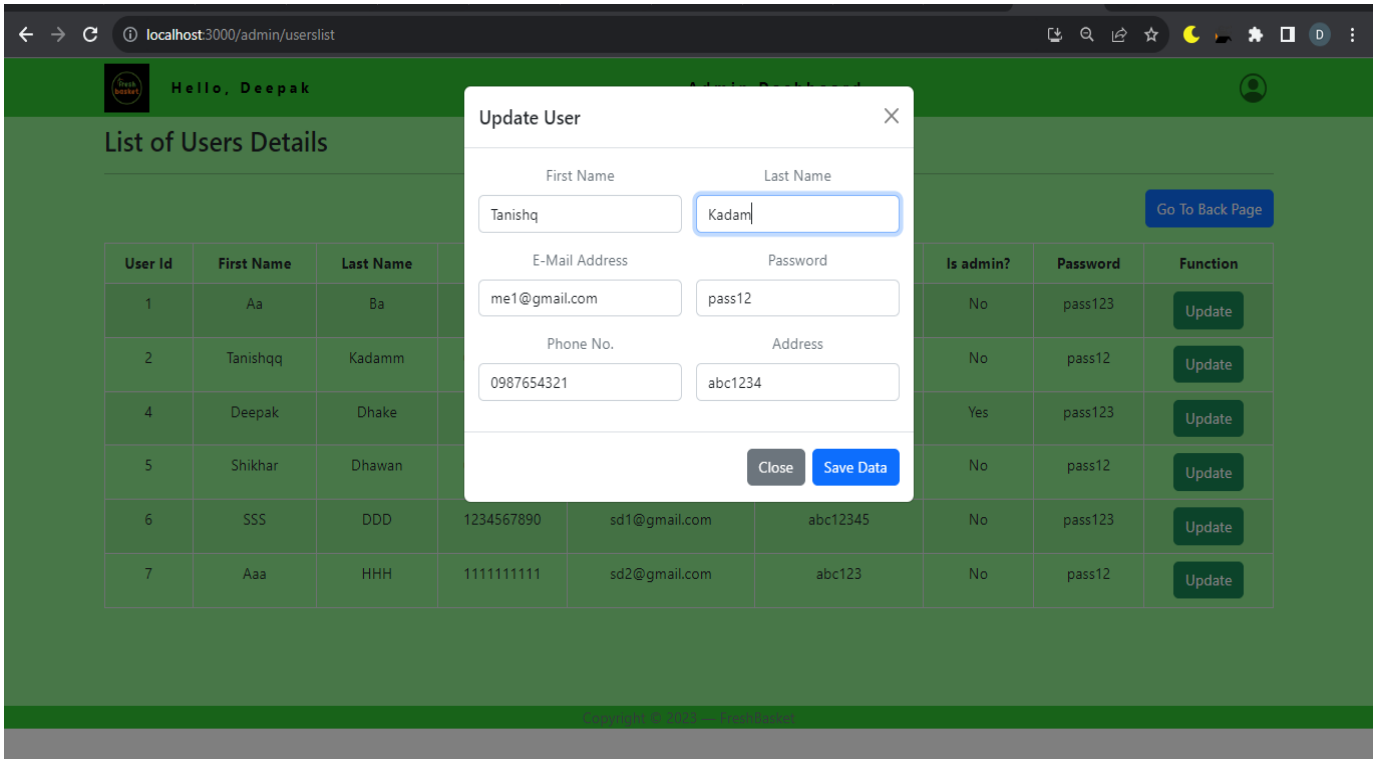
- **LIST OF FARMERS:**



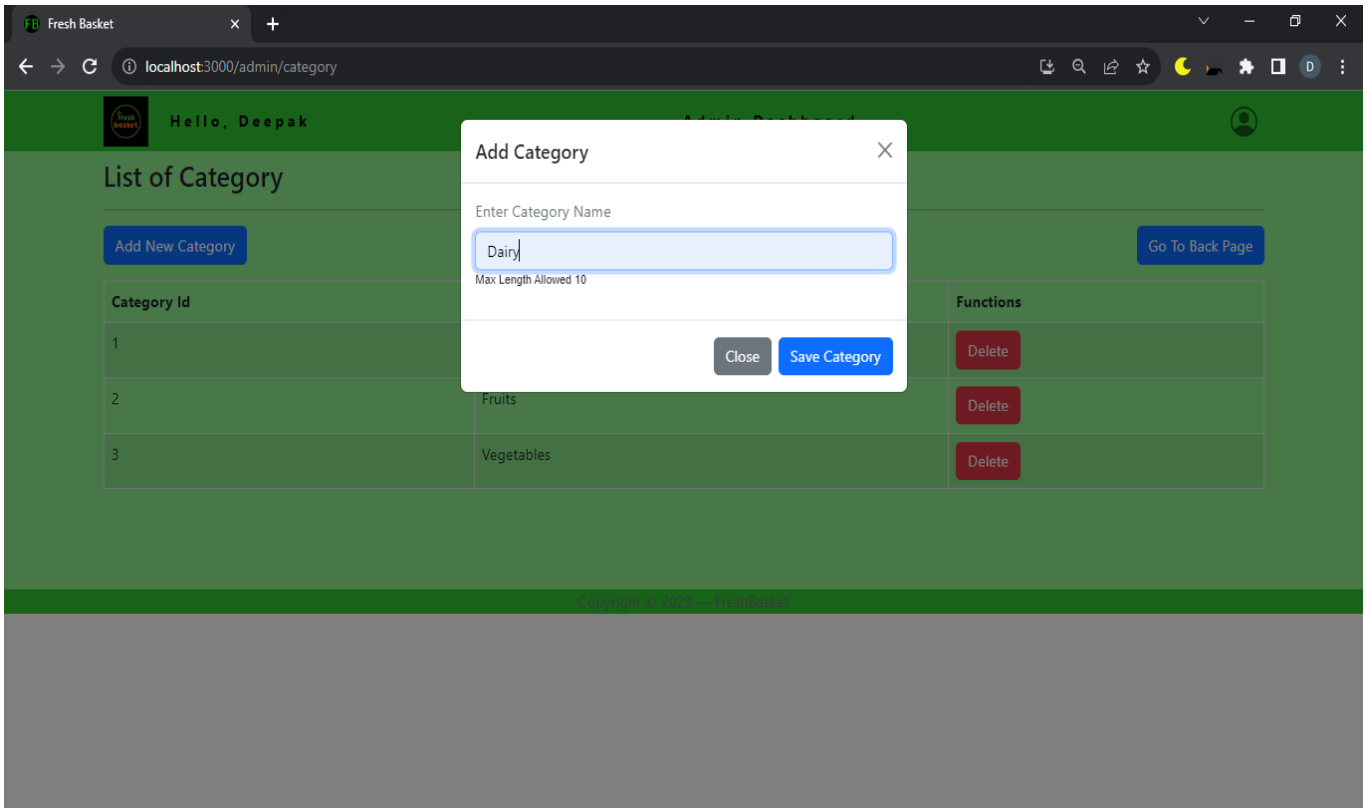
- **UPDATE FARMER DETAILS:**



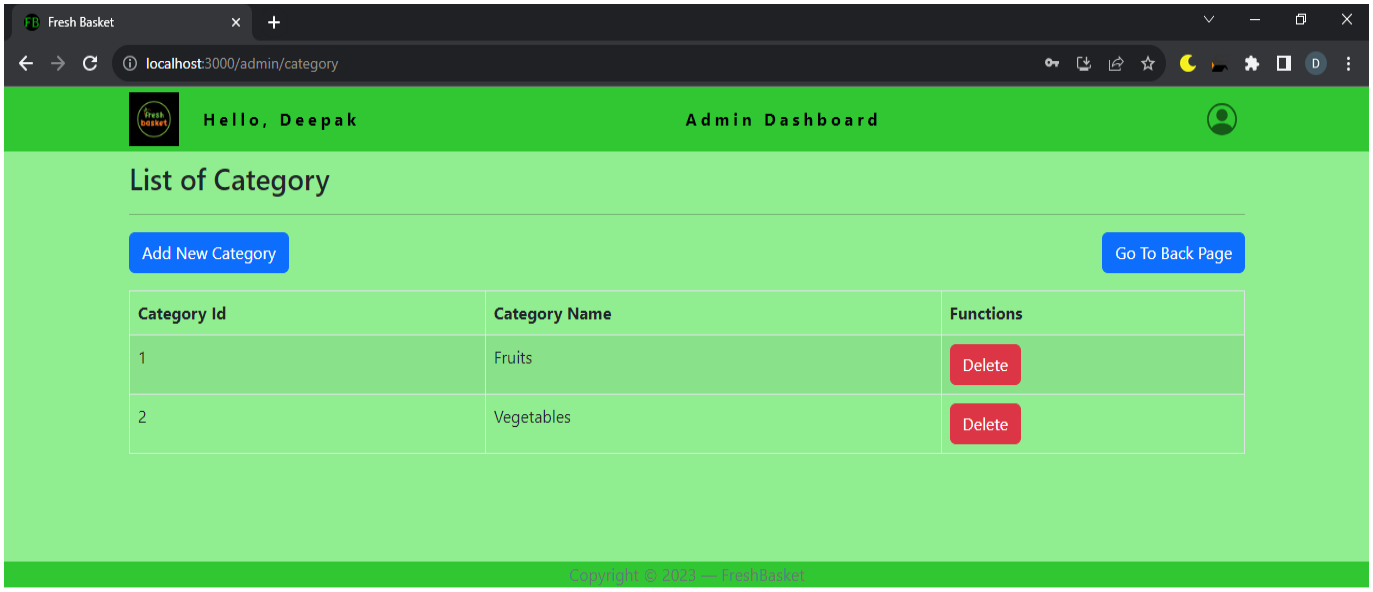
- **UPDATE USER DETAILS:**



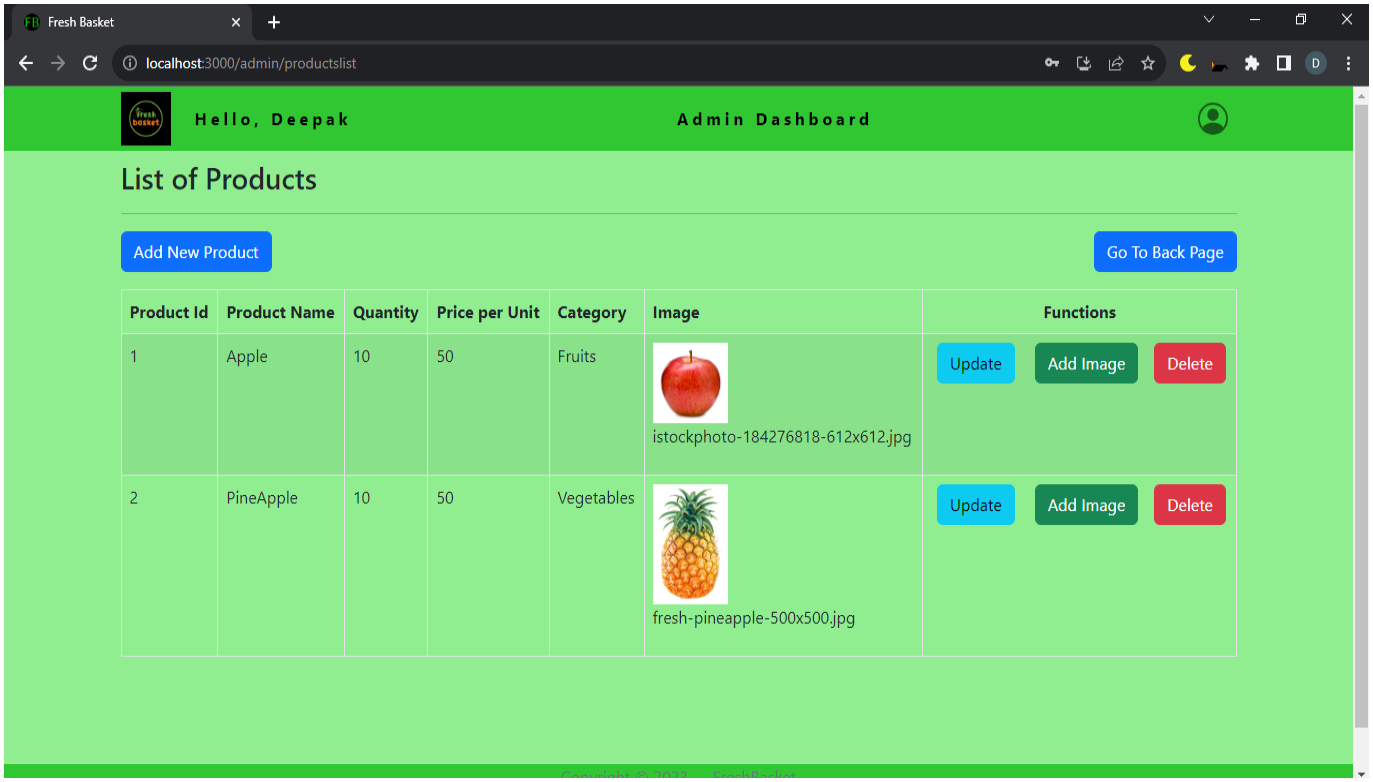
- **ADD NEW CATEGORY:**



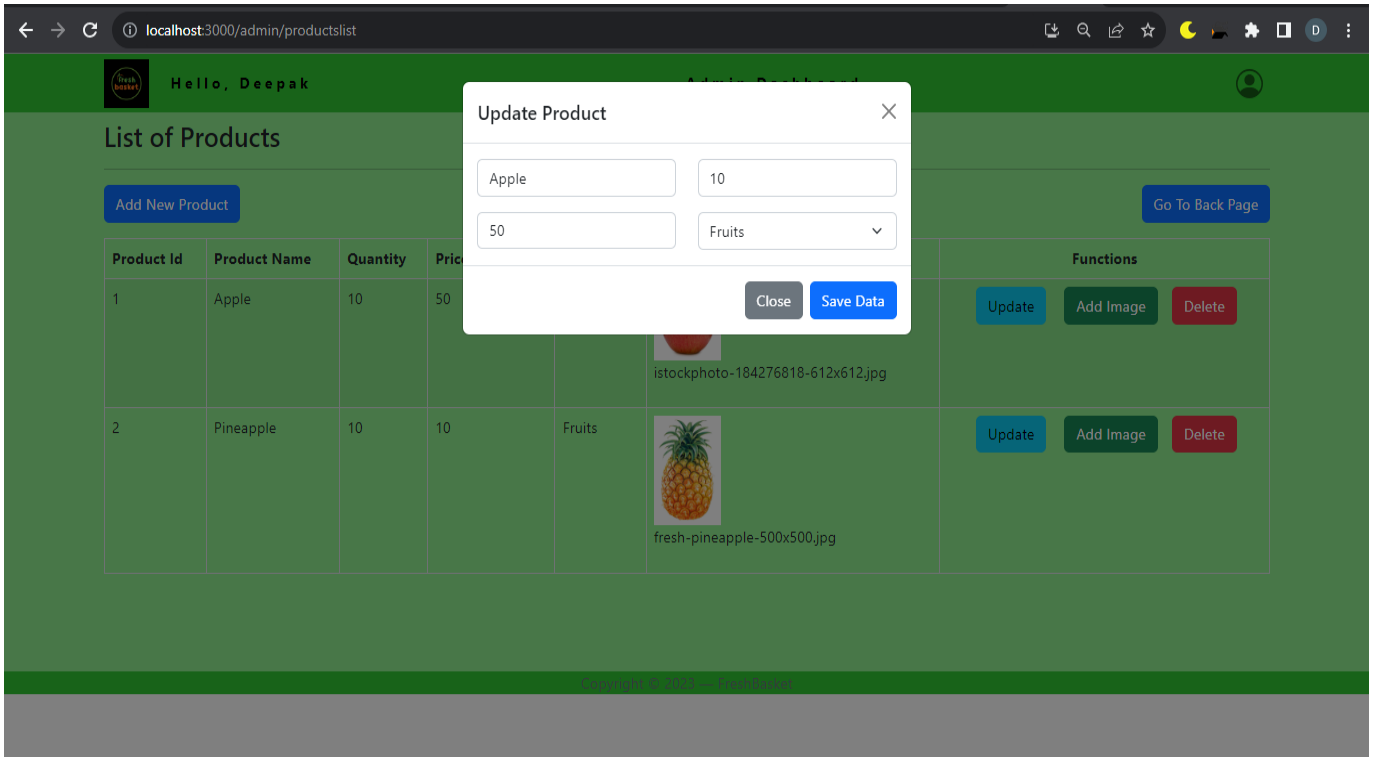
- **LIST OF CATEGORY:**



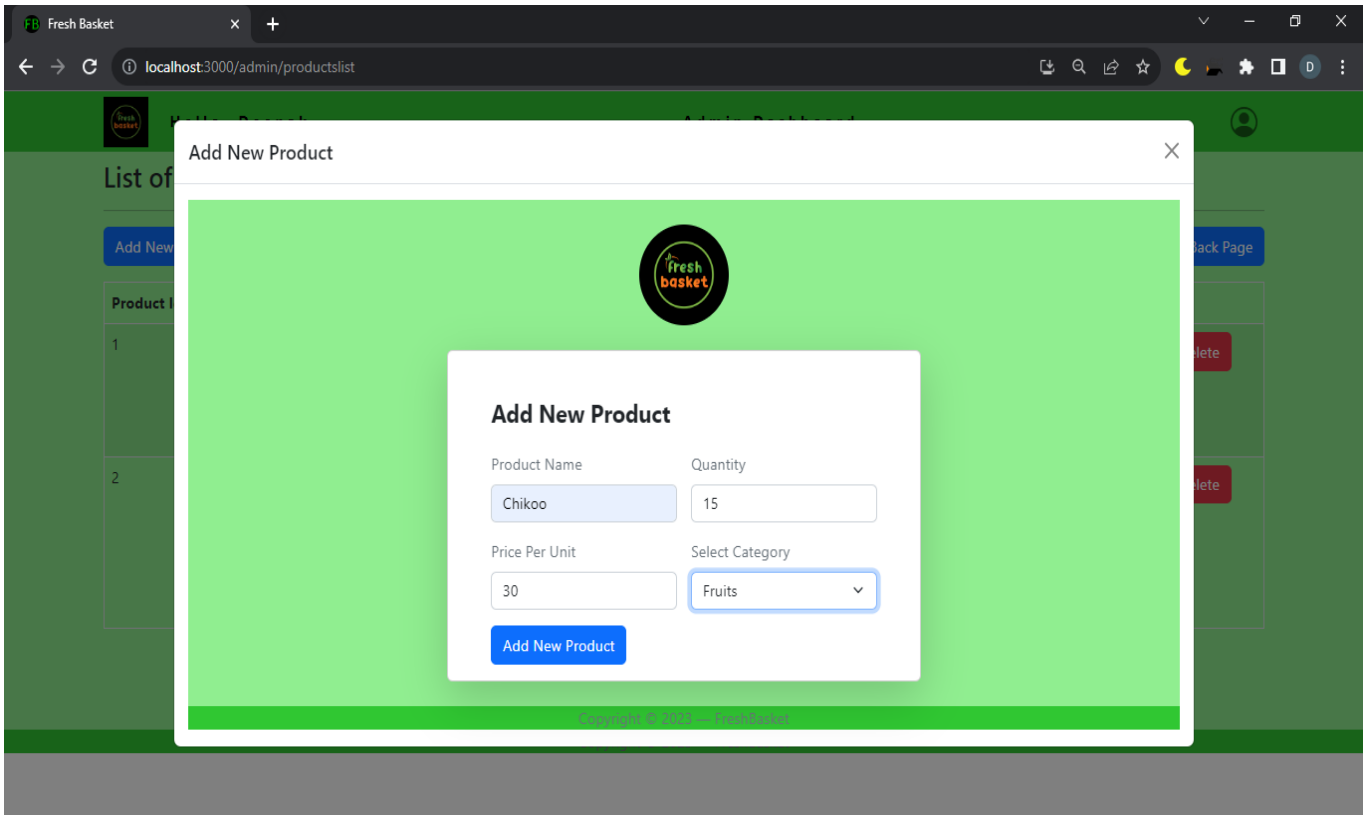
- **LIST OF PRODUCTS:**



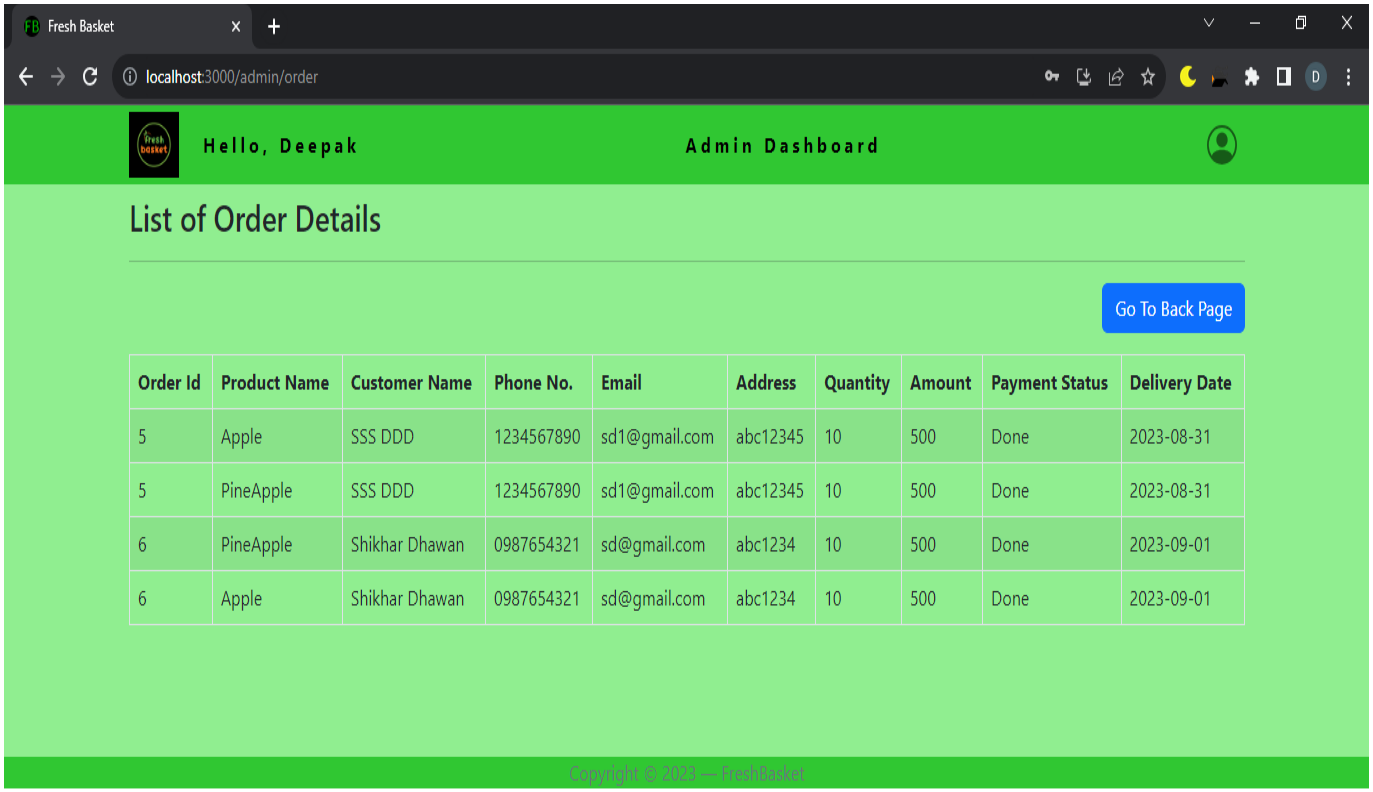
- **UPDATE PRODUCT:**



- **ADD NEW PRODUCT:**



- **ORDER DETAILS:**



7. CONCLUSION

"Fresh Basket", an online Grocery store application, was developed by our project team to simplify the online sale and purchase of Fresh-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, "Fresh Basket" 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.

- **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 favorites 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 upvote /downvote / report feedbacks.

6. Additional payment means can be added other than cards.

7. In case the user forgets the password, a 'reset password' functionality can be added.

8. CAPTCHA can be added to login page

8. REFERENCES

- Following is the list of websites we referred during the course of our project:

i. https://getbootstrap.com/docs/5.1/getting-started/introduction/
ii. https://reactjs.org/docs/getting-started.html
iii. https://www.baeldung.com/
iv. https://www.w3schools.com/
v. https://docs.spring.io/spring-data/jpa/docs/current/reference/html/#reference
vi. https://javaee.github.io/javaee-spec/javadocs/
vii. https://javadoc.io/doc/org.springframework.data/spring-data-jpa/latest/index.html