|  |
| --- |
| **A**  **PROJECT REPORT ON** |
|  |
|  |
| Agricultural Portal |
|  |
|  |
| SUBMITTED IN  PARTIAL FULFILLMENT OF  **DIPLOMA IN ADVANCED COMPUTING (PG-DAC)** |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
| **BY**  **JANAK KHORGADE (D1-80751)**  **TEJAS MOHTKAR(D1-80507) DEEPAK NIKHARE (D1-80839)**  **ASHUTOSH NAGRALE (D1-80636)** |
|  |
|  |
| **UNDER THE GUIDENCE OF**  **Shubham Borle** |
|  |
|  |
|  |
| **AT**  **SUNBEAM INSTITUTE OF INFORMATION TECHNOLOGY,**  **PUNE** |
|  |

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
|  | | | |
|  | | | |
|  | | | |
| **CERTIFICATE** | | | |
|  | | | |
| This is to certify that the project | | | |
|  | | | |
| Cropify | | | |
|  | | | |
| Has been submitted by | | | |
|  | | | |
| **JANAK KHORGADE**  **TEJAS MOHTKAR DEEPAK NIKHARE**  **ASHUTOSH NAGRALE** | | | |
|  | | | |
|  | | | |
| In partial fulfillment of the requirement for the Course of **PG Diploma in Advanced Computing (PG-DAC Sep 2023)** as prescribed by The **CDAC** ACTS, PUNE. | | | |
|  | | | |
|  | | | |
| Place: Pune | | | Date: 22-FEB-2024 |
|  | | | |
|  | | | |
|  | | | |
|  | | | |
|  | | | |
|  | | | **Shubham Borle** |
|  | | | **Project Guide** |
|  |

**ACKNOWLEDGEMENT**

I would like to express my special thanks of gratitude to my mentor and guide Shubham Borle as well as our Director Nitin Kudale who gave me the golden opportunity to do this wonderful project on the topic Cropify, which also helped me in doing a lot of Research and I came to know about so many new things I am really thankful to them.

Janak Khorgade

**ABSTRACT**

Cropify is an online platform that empowers farmers by providing a direct channel to sell their produce at fair prices. It also serves as a marketplace for farmers to procure essential agricultural supplies. Additionally, Cropify offers valuable farming information, including a weather index. This initiative promotes transparency and sustainability in the agricultural sector.

|  |  |  |
| --- | --- | --- |
|  | **INTRODUCTION** | 1 |
|  | 1.1 Introduction | 1 |
|  | **Product Overview and Summary** | 2 |
|  | 2.1 Purpose | 2 |
|  | 2.2 Scope | 2 |
|  | 2.3 Overview | 3 |
|  | 2.4 Feasibility Study | 8 |
|  | **REQUIREMENTS** | 10 |
|  | 3.1 Functional Requirements | 10 |
|  | 3.2 Non - Functional Requirements | 11 |
|  | **PROJECT DESIGN** | 12 |
|  | 4.1 Data Model | 12 |
|  | 4.1.1 Database Design | 18 |
|  | 4.2 Project Architecture | 19 |
|  | 4.3 Use case Diagram | 20 |
|  | 4.4 Data-flow Diagram | 21 |
|  | **PROJECT SCREENSHOTS** | 20 |
|  | **CONCLUSION** | 29 |
|  | **FUTURE SCOPE** | 30 |
|  | **REFERENCES** | 31 |

**INDEX**

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **Section** | **Table Title** | **Page** |
| **Fig 1** | user Table | 13 |
| **Fig 2** | admin table | 13 |
| **Fig 3** | agriculture\_products table | 14 |
| **Fig 4** | farm\_products table | 14 |
| **Fig 5** | machinery table | 14 |
| **Fig 6** | farm\_product\_details table | 15 |
| **Fig 7** | seller\_agriculture\_product\_details table | 15 |
| **Fig 8** | seller\_machinery\_details table | 15 |
| **Fig 9** | order\_farm\_product\_details table | 16 |
| **Fig 10** | order\_agriculture\_product\_details | 16 |
| **Fig 11** | order\_machine\_details table | 16 |
| **Fig 12** | cart\_farm\_product table | 17 |
| **Fig 3** | cart\_agriculture\_product table | 17 |
| **Fig 14** | cart\_machinery table | 17 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Section** | **Figure Title** | **Page** |
| **Fig 15** | Complete Database | 18 |
| **Fig 16** | Project Architecture | 19 |
| **Fig 17** | Use Case Diagram | 20 |
| **Fig 18** | Data-flow Diagram | 21 |
| Project UI Images | | |
| **Fig 17** | Frontpage | 22 |
| **Fig 18** | Information Page | 22 |
| **Fig 19** | Registration Page | 23 |
| **Fig 20** | Login Page | 23 |
| Images related to Customer | | |
| **Fig 21** | Customer Homepage | 24 |
| **Fig 22** | Customer Shopping Page | 25 |
| **Fig 23** | Cart | 23 |
| **Fig 24** | Orders | 23 |
| **Fig 25** | Contact Us | 24 |
| Images related to Admin | | |
| **Fig 26** | Dashboard | 24 |
| **Fig 27** | Listed Products | 25 |
| **Fig 28** | Farmer And Seller List(Verify/Block Farmer and Seller) | 25 |
| **Fig 29** | Add Products | 25 |
| Images related to Farmers and Sellers | | |
| **Fig 30** | Profile | 26 |
| **Fig 31** | Dashboard | 26 |
| **Fig 32** | Product List | 27 |
| **Fig 33** | Order List | 27 |
| **Fig 34** | Add Product | 27 |
| **Fig 35** | Rent Machinery (For Farmers) | 28 |

# 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. Online Shopping of Fresh Food and lending agricultural tools and machineries to farmers opens up a new world of options. Users won’t have to go from store to store to hunt for fresh food whereas farmers who are not able afford expensive agricultural machineries and tools can now lend them from other farmers and can also lend their machineries and tools to other farmers as well. Users 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. Farmers don’t have to worry about renting tools and machineries because every tool and machinery is verified.

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

Customer can pick what food they want to order and add to the cart, farmers can lend all the tools and machineries and sellers can rent their machineries. Once they are done selecting what they require, after reviewing cart summary they can simply click on check out button to pay bill verifying them on their given mails and with a secured transaction. 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

# Cropify, 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 and connecting farmers directly to sellers of agriculture products, tools and machineries. 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

# “Cropify” aims to deliver a web-based application that hosts a wide collection of the food-items, agricultural products such as tools and machineries that users and farmer can browse through. Users can place orders and make payment. Farmers can lend and rent agricultural tools and machineries. 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 and sellers. Admins can even delete users and/or farmers/sellers, if the need arises.

# This project does not support the actual logistics and delivery of food items. 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. Cropify is only an interface for both customers (for browsing and shopping for food items, agricultural tools and machineries) and admins (for managing products, farmers, sellers and users listing).

# OVERVIEW

# TECHNOLOGIES USED

1. **FRONT END**
   * HTML
   * CSS
   * Bootstrap
   * JavaScript
   * React
   * Axios
2. **BACK END**
   * Spring Boot
   * Spring Data JPA
   * Hibernate
3. **DATABASE MANAGEMENT SYSTEM**
   * MySQL

# FEATURES PROVIDED

1. **FOR ADMINS**
   1. Login & Logout – Similar to customers, admins can login & logout to access their account.
   2. Add / Update Farmers –Only admin is responsible for adding and updating the details of farmer
   3. Add / Update Sellers –Only admin is responsible for adding and updating the details of seller.
   4. Delete Farmer –The admins can delete a farmer account if they need to for any purpose.
   5. Delete Seller –The admins can delete a seller account if they need to for any purpose.
   6. Add New Category – Admins can add category.
   7. Delete Category – Admins can remove category.
   8. Add New Products – Admin can add new product with details as stock, price, name, quantity, image, category, etc.
   9. Manage Products– Admin can update the product details.
   10. View Users – Admin can view all registered users.
   11. Delete User – Admin can delete a user if need arises.
   12. View order details – Admin can view order details for all users.
2. **FOR CUSTOMERS**
   1. Browse – Customers can browse the homepage to explore the entire products available.
   2. 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.
   3. View & Update Profile – When logged in, customers can view their profile and update their details.
   4. Update Delivery Address – When purchasing listed items, a customer can update delivery addresses which they can associate with their account.
   5. 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.
   6. 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.
3. **FOR FARMERS**
   1. Browse – Farmers can browse the homepage to explore his dashboard.
   2. Register, Login & Logout – New farmers can register on the site. Existing farmers can then login to access their account information and logout when the account is not in use.
   3. View & Update Profile – When logged in, farmers can view their profile and update their details.
   4. Update Delivery Address – When purchasing listed items, a customer can update delivery addresses which they can associate with their account.
   5. 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.
   6. 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.
4. **FOR SELLERS**
   1. Browse – Sellers can browse the homepage to explore his dashboard.
   2. Register, Login & Logout – New sellers can register on the site. Existing sellers can then login to access their account information and logout when the account is not in use.
   3. View & Update Profile – When logged in, sellers can view their profile and update their details.
   4. Update Delivery Address – When lending listed items, a seller can update delivery address of customers/farmers.

# 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:

### **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 is used in front-end is 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 users and admins need to be stored in a robust and secure database, MySQL database management system was chosen as it is an industry standard.

### **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.

### **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 are free, open-source, and widely available, with each of the technologies having an extensive community support. This makes “Cropify” an economically feasible solution to the organizations that wish to implement it.

# REQUIREMENTS FULFILLED

#### **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, sellers 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.

#### **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.

# PROJECT DESIGN

# 4.1 DATA MODEL

# The following diagram depict the database design used for “Cropify” application.

# 

# Fig 1 user Table

# 

# Fig 2 admin table

# 

# Fig 3 agriculture\_products table

# 

# Fig 4 farm\_products table

# 

# Fig 5 machinery table

# Fig 6 farm\_product\_details table

# Fig 7 seller\_agriculture\_product\_details table

# Fig 8 seller\_machinery\_details table

# Fig 9 order\_farm\_product\_details table

# Fig 10 order\_agriculture\_product\_details table

# Fig 11 order\_machine\_details table

# Fig 12 cart\_farm\_product table

# Fig 13 cart\_agriculture\_product table

# Fig 14 cart\_machinery table

# 

Fig 15 Complete Database

# 4.2 PROJECT ARCHITECTURE

# 

# Fig 16 Project Architecture

# 4.3 USE CASE DIAGRAM

# 

# Fig 17 Use Case Diagram

# 4.4 DATA-FLOW DIAGRAM

# 

# Fig 18 Data-Flow Diagram

# PROJECT SCREENSHOTS

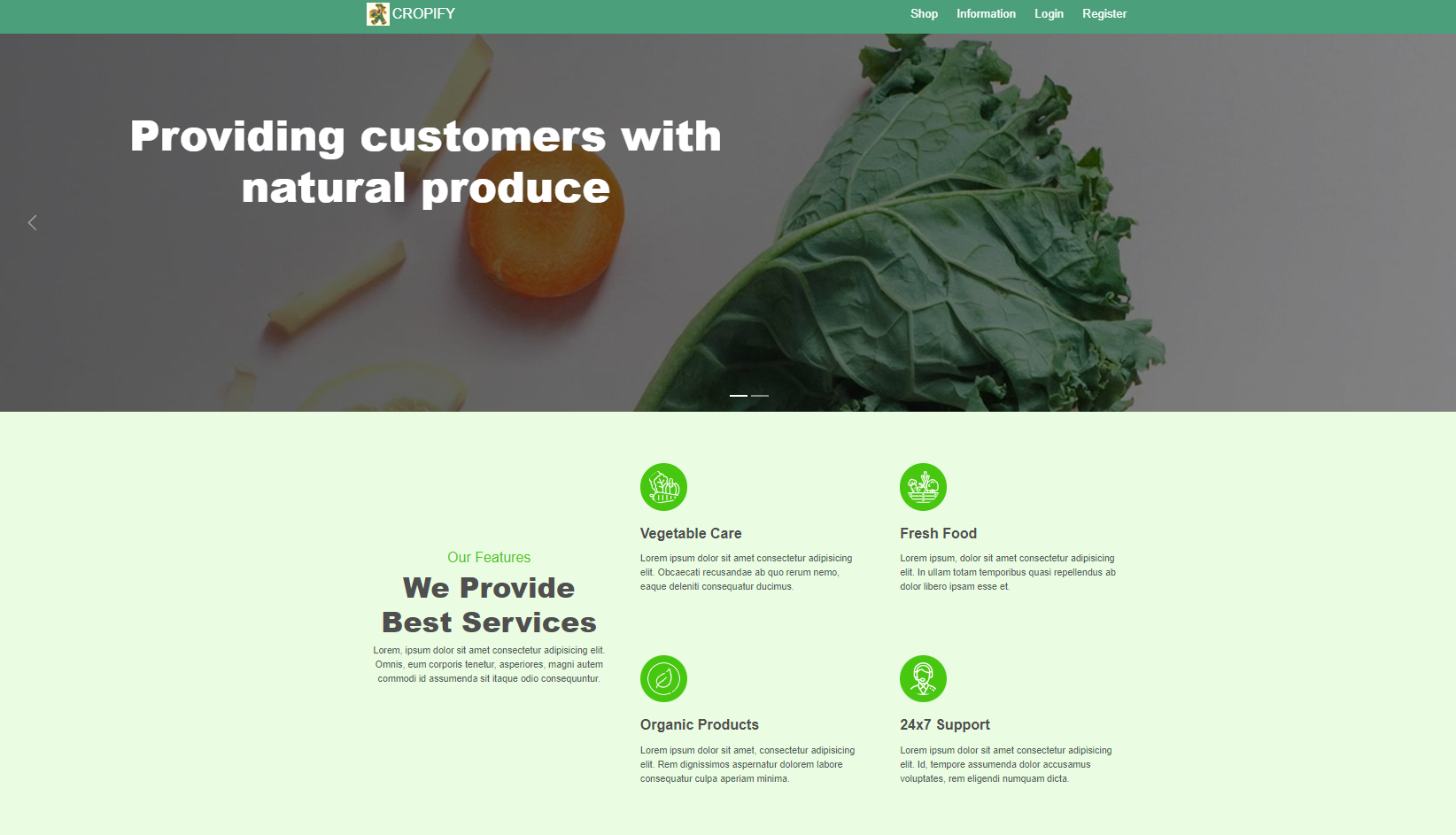


Fig 19 Frontpage

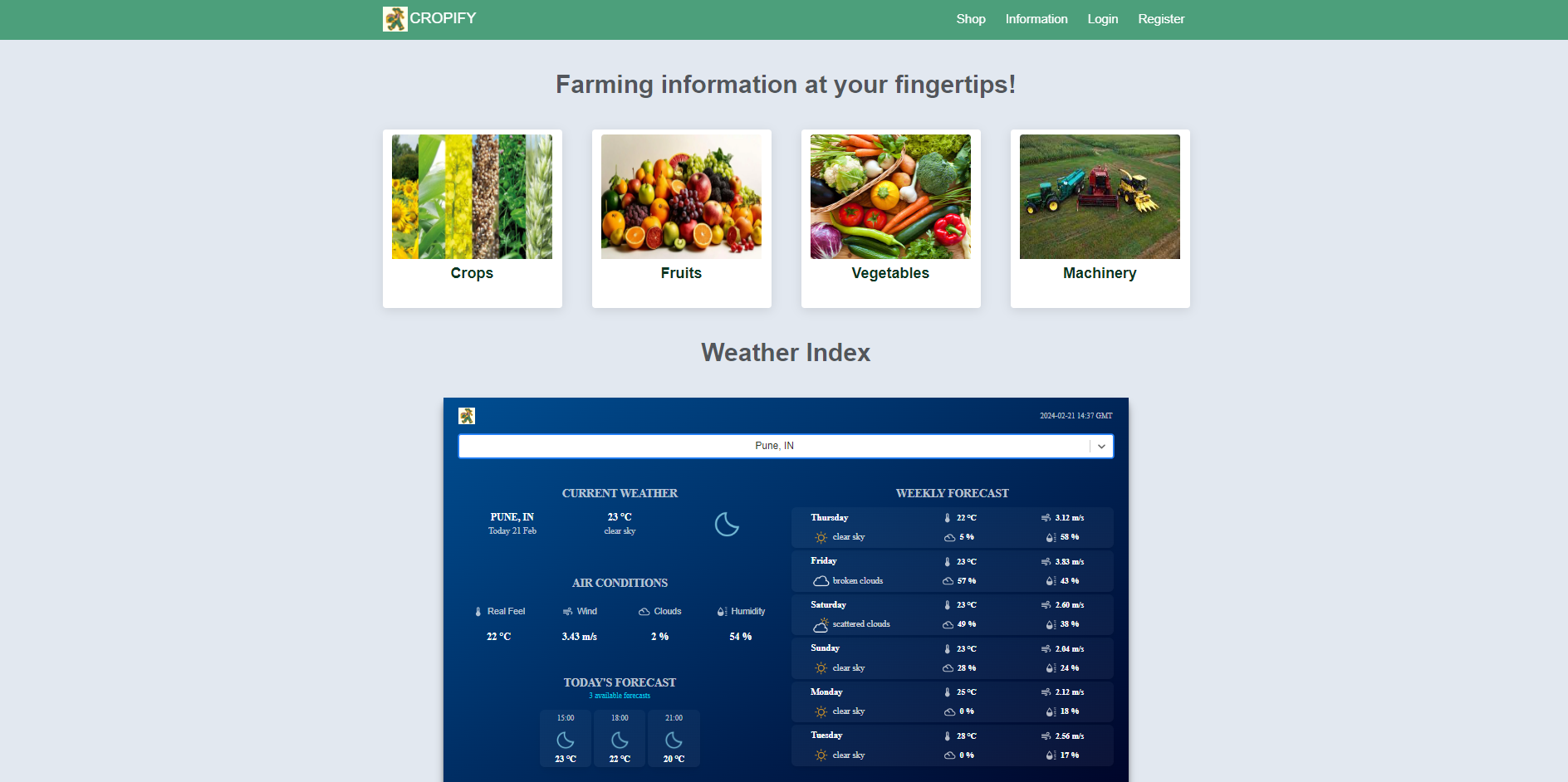


Fig 20 Information Page

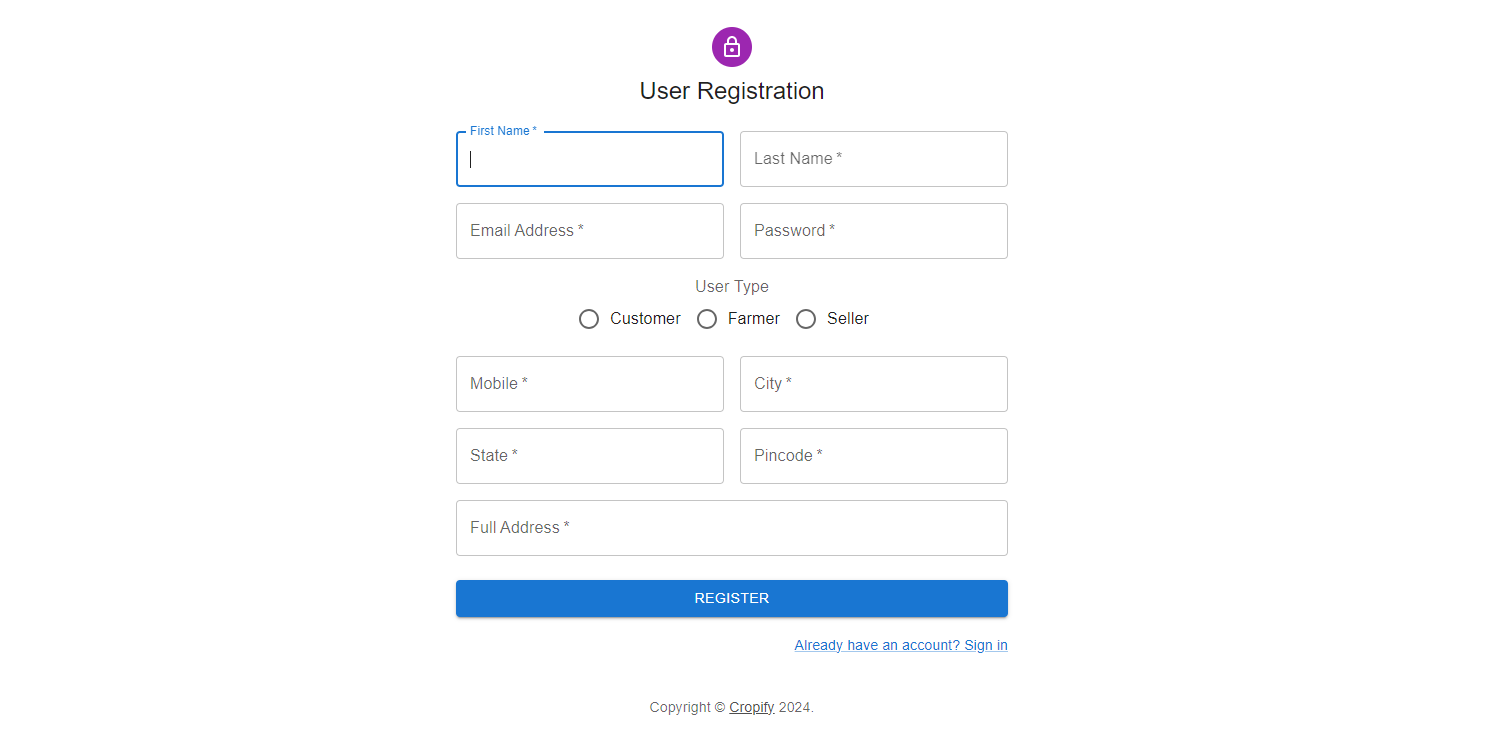


Fig 21 Registration Page

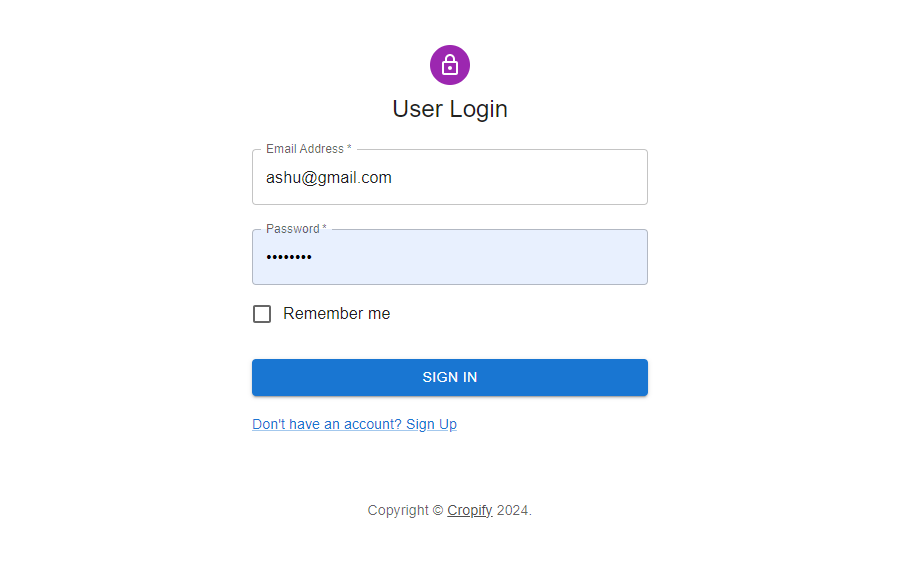


Fig 22 Login Page

**Images related to Customer**

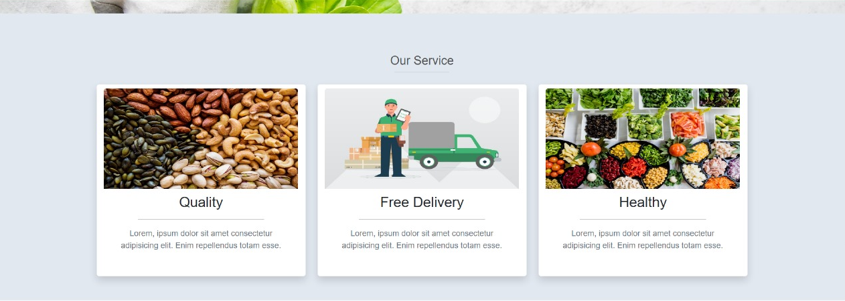
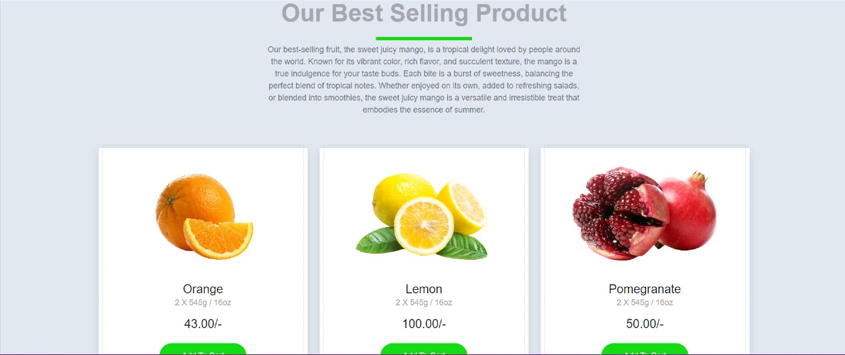
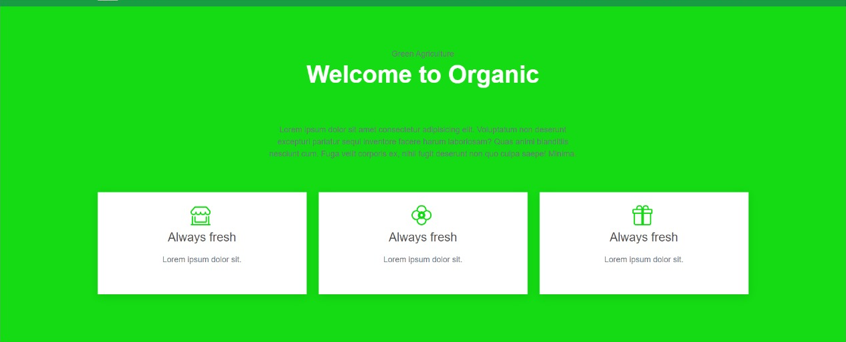
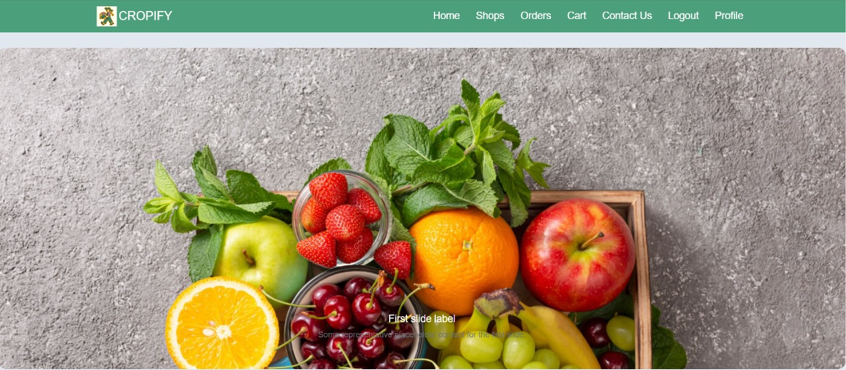


Fig 23 Customer Homepage

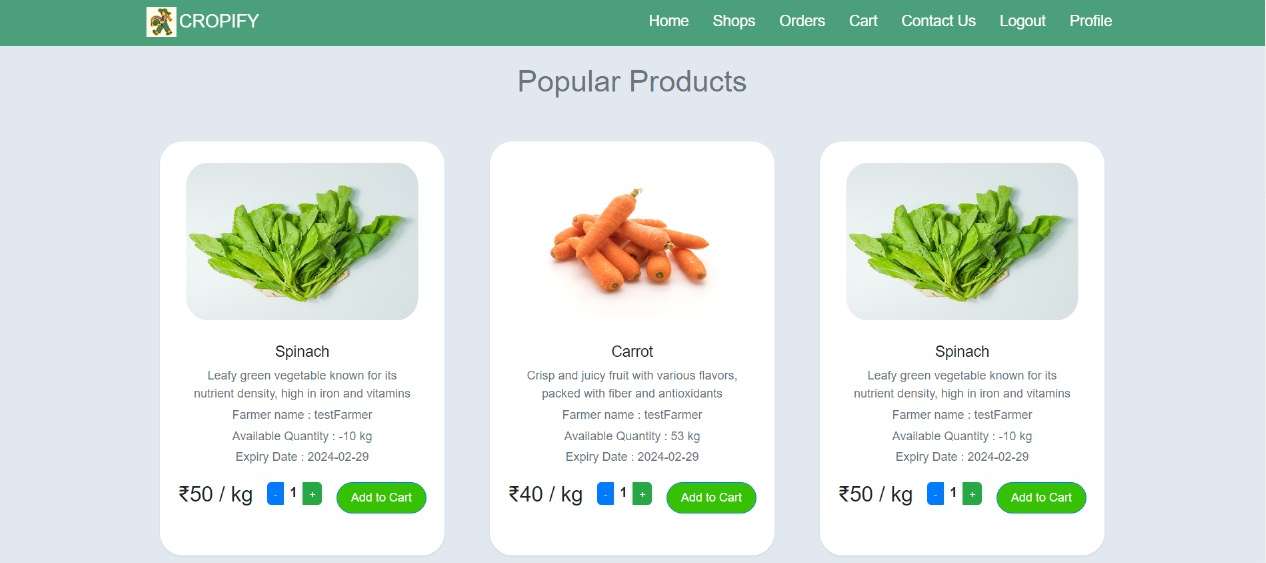


Fig 24 Customer Shop

=

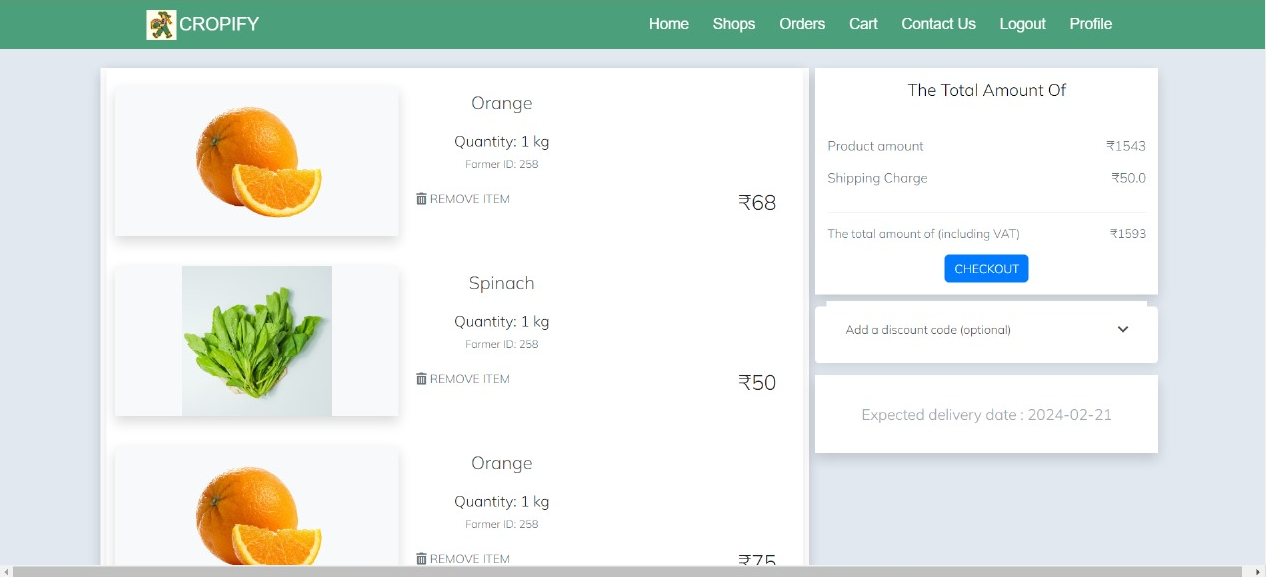


Fig 25 Cart

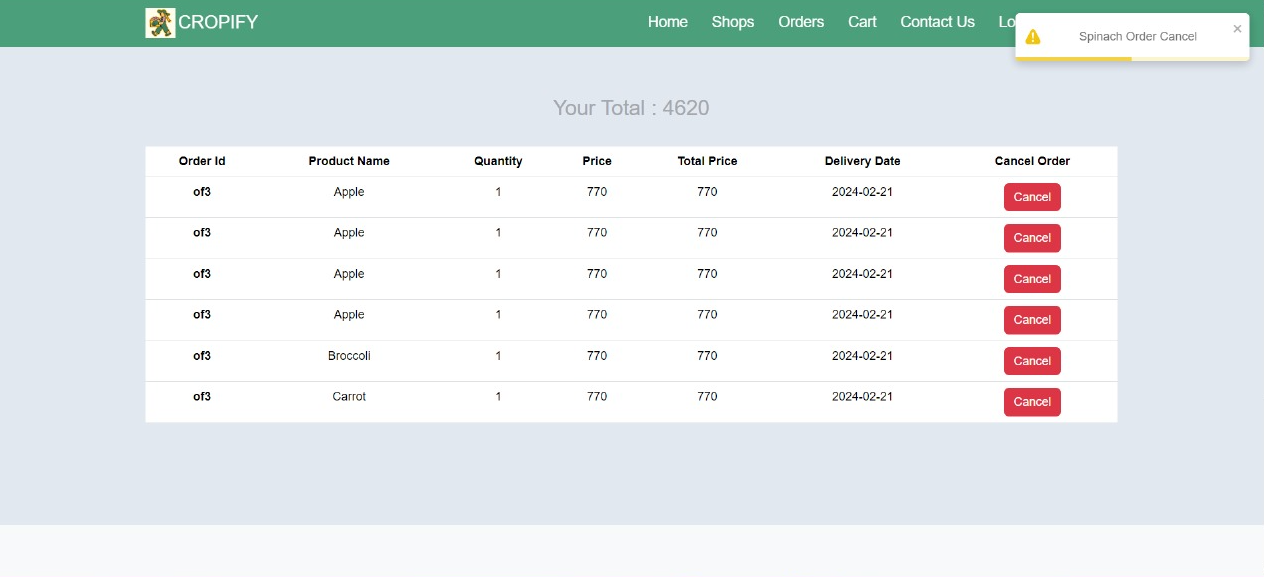


Fig 26 Orders

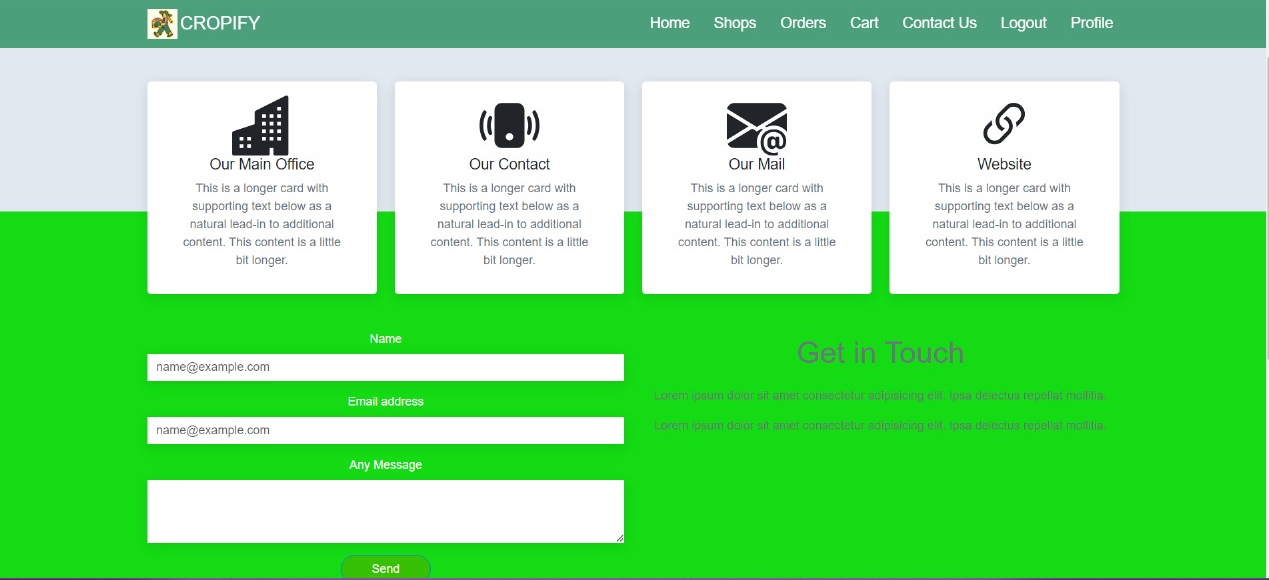


Fig 27 Contact Us

**Images related to Admin**

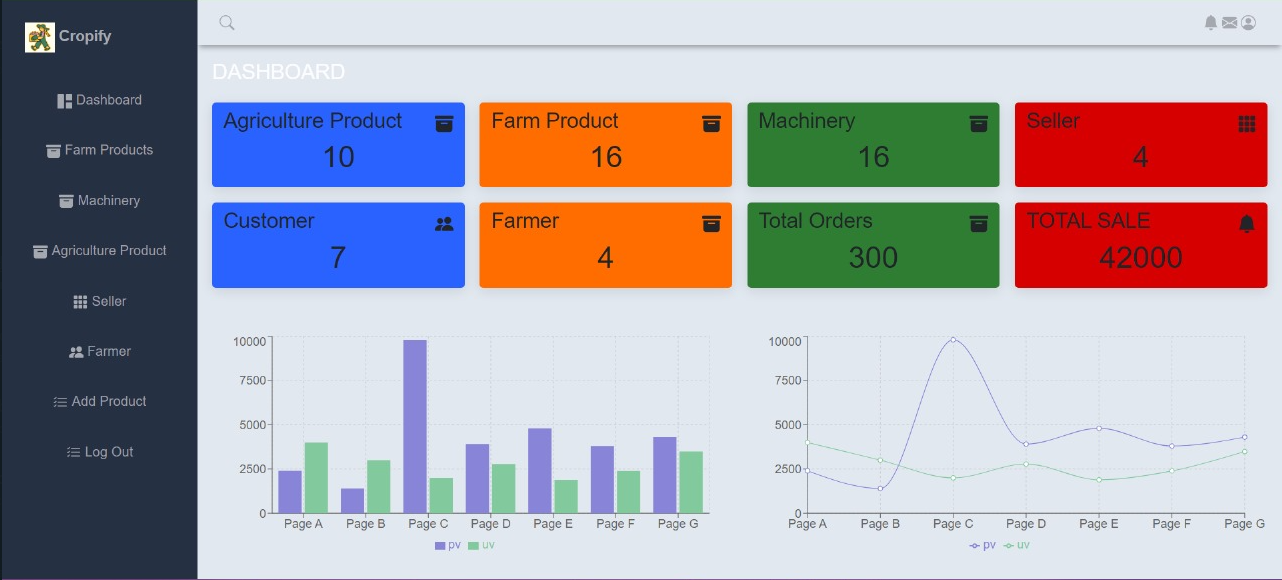


Fig 28 Dashboard

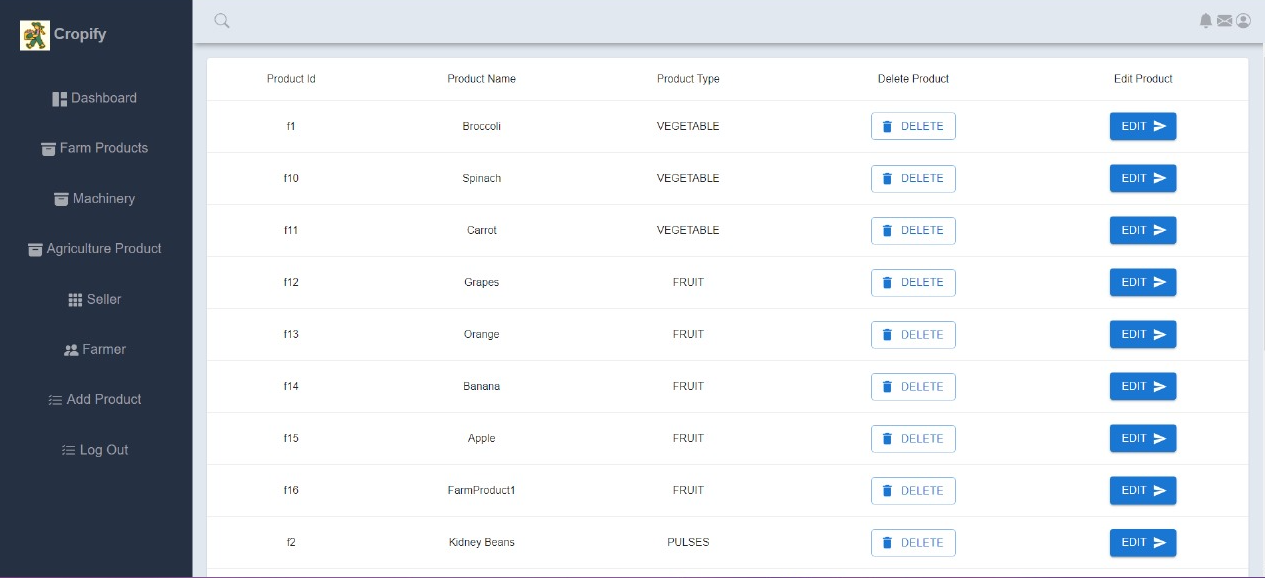


Fig 29 Listed Products

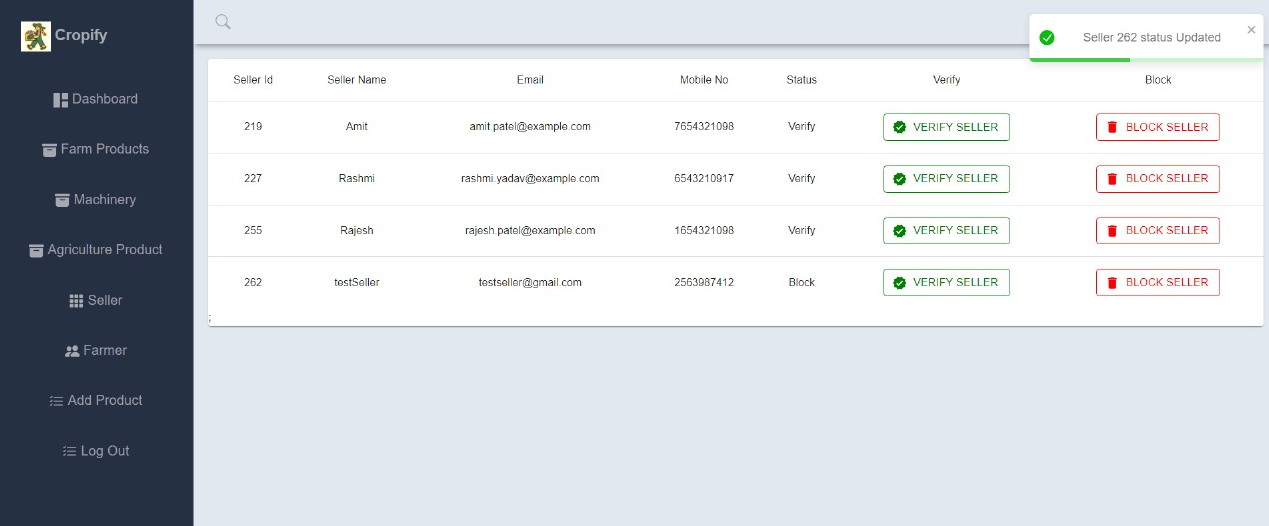


Fig 30 Farmer and Seller List



Fig 31 Add Products

**Images related to farmers and sellers**

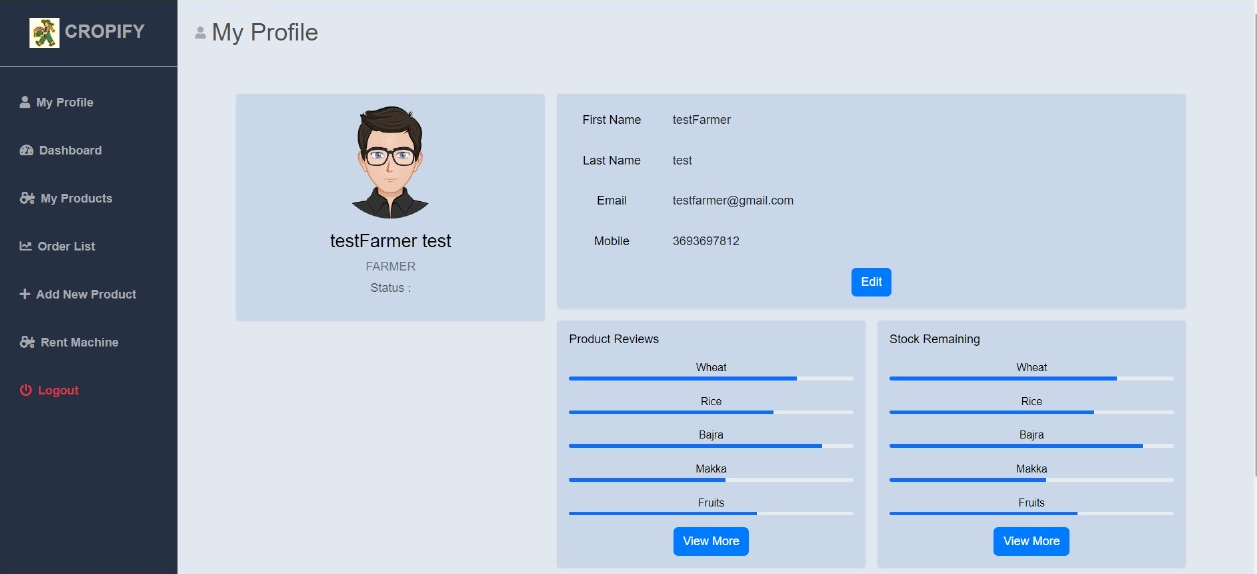


Fig 32 Profile



Fig 33 Profile

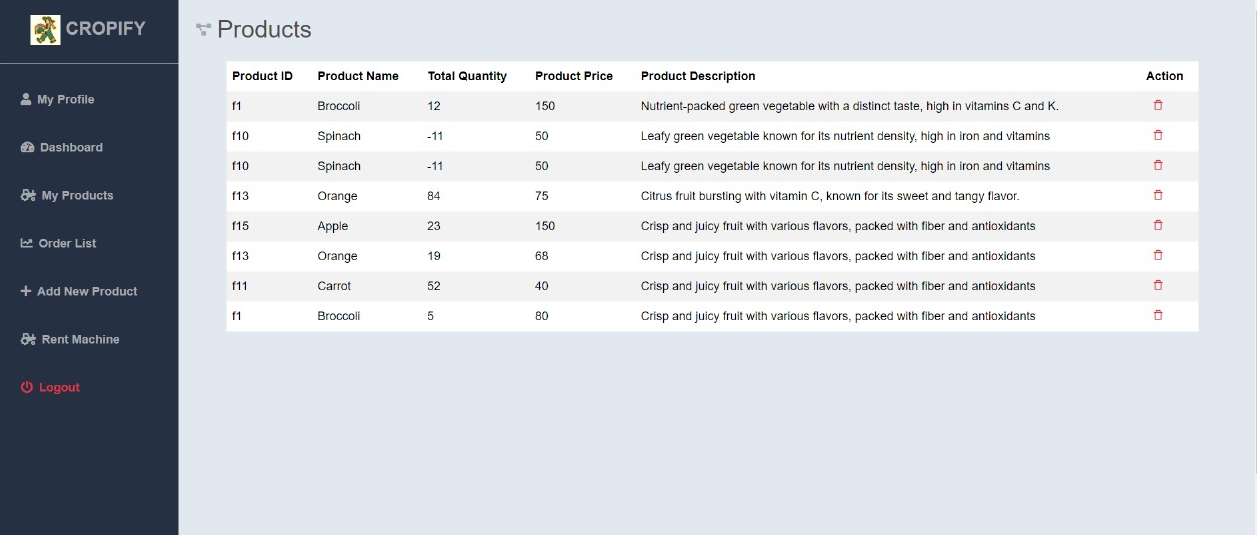


Fig 34 Product List

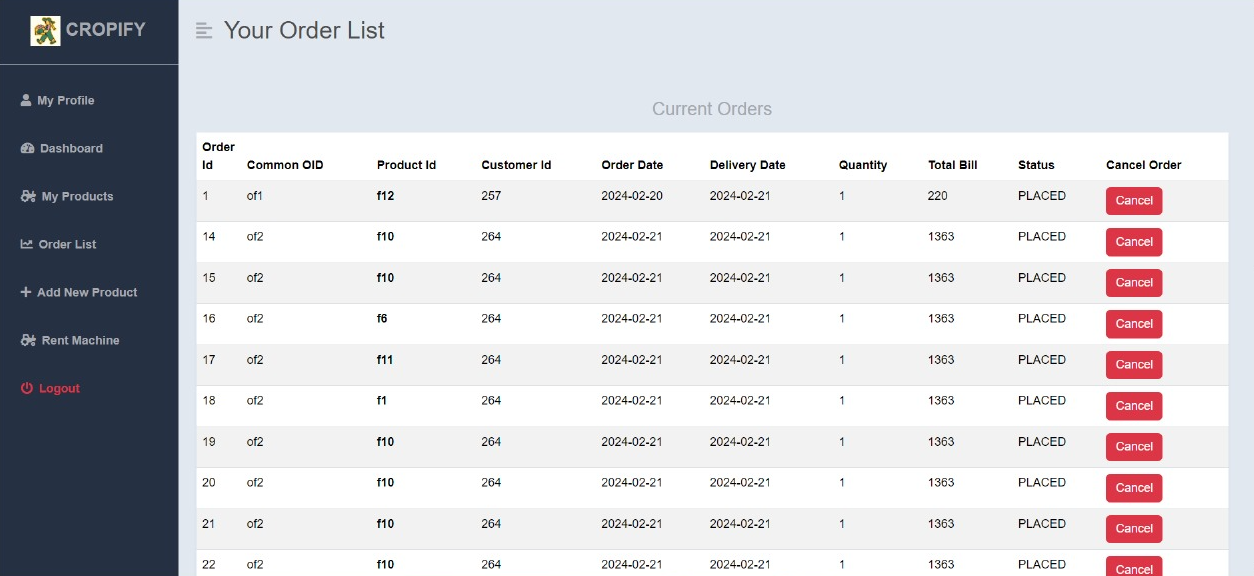


Fig 35 Order List

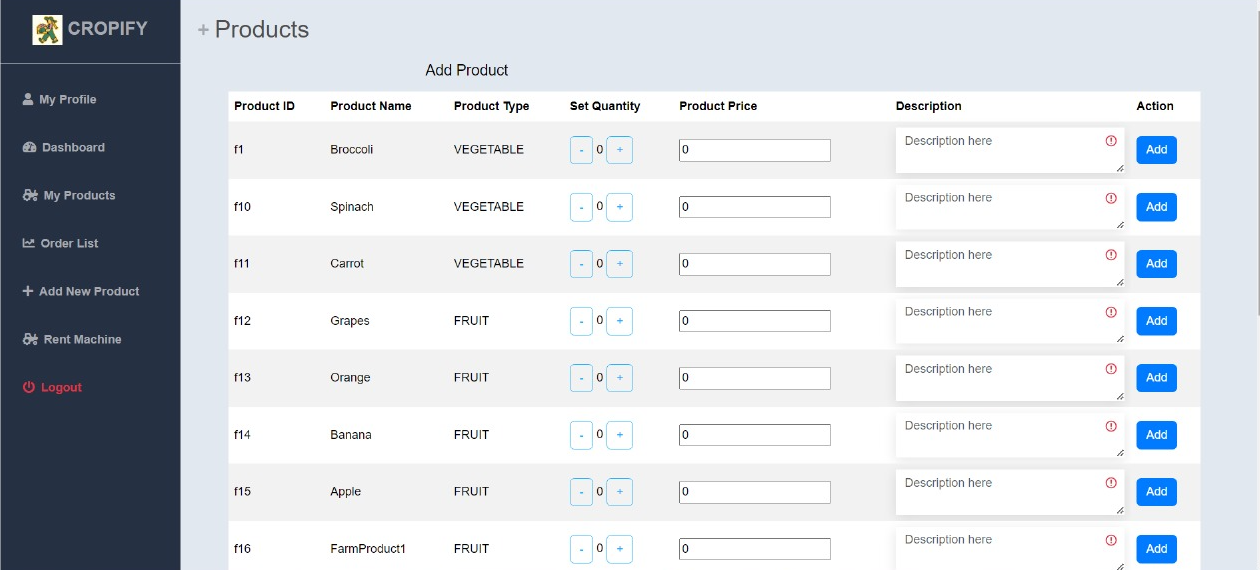


Fig 36 Add Products



Fig 37 Rent Machinery (for farmers)

# CONCLUSION

“Cropify”, an online Grocery store application, was developed by our project team to simplify the online sale and purchase of Fresh-organic merchandise and to help farmers with agricultural products.

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, “Cropify” is an application would definitely be a good choice for any fresh-food merchandise trading Farmers and seller that wishes to enter the online market. At the same time, it provides one-stop platform for Customers and Farmers to purchase their daily need of merchandise directly from authenticated Farmers and to rent or lend machinery from other farmers or sellers directly.

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.

1. Customers can upvote/downvote/report feedbacks.
2. In case the user forgets the password, a ‘reset password’ functionality can be added.
3. CAPTCHA can be added to login page.

# REFERENCES

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

* 1. <https://getbootstrap.com/docs/5.1/getting-started/introduction/>
  2. <https://reactjs.org/docs/getting-started.html>
  3. [https://www.baeldung.com/spring-boot](https://www.baeldung.com/spring-boot%20)
  4. <https://www.w3schools.com/>
  5. <https://docs.spring.io/spring-data/jpa/docs/current/reference/html>
  6. <https://github.com/javaee/javaee-spec/tree/master/javadocs>
  7. [https://javadoc.io/doc/org.springframework.data/spring-data- jpa/latest/index.html](https://javadoc.io/doc/org.springframework.data/spring-data-%20jpa/latest/index.html)
  8. <https://github.com/amaroteam/react-credit-cards>