



VIVEKANAND COLLEGE FOR ADVANCED COMPUTER AND
INFORMATION SCIENCE

VEER NARMAD SOUTH GUJARAT UNIVERSITY, SURAT

PROJECT REPORT

OF

Fresh connect

AS A PARTIAL REQUIREMENT FOR THE DEGREE

BACHELOR OF COMPUTER APPLICATION
(B.C.A.)

2024-2025

GUIDED BY:

Ms. Nidhi Mehta

SUBMITTED BY:

Ghantiwala Vivek.[1670]
Patel Aman.[1845]
Upale Rutvesh.[1977]
Patel Bhavika.[1849]



**VIVEKANAND COLLEGE FOR
ADVANCED COMPUTER & INFORMATION SCIENCE**

(Affiliated to the Veer Narmad South Gujarat University)

Near Saroli Bridge, Jahangirpura, Surat-Olpad Road, SURAT-5.

CERTIFICATE

This is to certify that GHANTIWALA VIVEK ANKUR-1670, PATEL AMANKUMAR BHARATBHAI-1845, UPALE RUTVESH SANTOSHBHAI-1977, PATEL BHAVIKAKUMARI ANILBHAI-1849 have successfully completed their project work entitled "**FRESH VEGETABLES BADGE FOR RESTAURANTS**" as partial fulfillment of the requirement of BCA sixth semester during academic year 2024-2025.

Name of Guide: **Dr. Nidhi Joshi**

Signature of Guide: Nidhi Joshi

Signature of Examiner: _____

A handwritten signature in blue ink, appearing to read "Dr. Nidhi Joshi".

Date:

Place: Surat



I/C Principal
Vivekanand College for BCA
Surat

Acknowledgement

Our self Ghantiwala Vivek, Patel Aman, Upale Rutvesh, Patel Bhavika. Have opportunity to express our knowledge. We would like to express our gratitude to all those who gave us the possibility to complete our project. We want to thanks to Vivekanand College for advanced computer and information science for giving me permission to do project work in their organization.

Success is such a comprehensive project cannot achieve singlehanded. It is team effort that sails the Ship to the coast. So we would like to express our sincere thanks to all the dignitaries who were involved in making this project the great joy and turning it in to successful piece of work.

I would like to thanks the Campus Director of Vivekanand College for B.C.A Ms. Sumathi Shenoy and I/C Principal Ms. Sandip Surati for helping attitude and encouraging us to excel in studies. Now we also thankfully to our guide Ms. Ankita Patel. He always responded us with smile for discussing the problem come during the project work. They are the person who has giving this direction to our work and the shape to our imagination.

We also like to Thanks our all the Professor who are always ready to give best guide. They are the person who give solution whenever needed. We would also like to acknowledge all the friends and colleagues, team member for the help and encouragement by them for time to time. It was the great experience of exposing and learning new things in this field. Thank You...

Ghantiwala Vivek.
Patel Aman.
Upale Rutvesh.
Patel Bhavika.

Preface

Before talking about our project work of foundation, it is quite necessary to have an exact idea of the work “project”. The project is made by joint efforts of all group members.

By Planning, Communication, Task of Working, Resource planning and operations we made our project. We worked very hard to make this project a reality.

We are happy to hand over this project to the Vivekanand College for Advanced Computer and Information Science. This project is titled **FreshConnect**, and it is focused on connecting users directly with local sellers and hotel service providers. It will be useful in providing online bookings of hotel rooms and enabling users to buy fresh products directly from sellers.

In a computer application study, partial training is very important. We can improve theoretical knowledge by reading and attending classes, but it is incomplete without gaining practical knowledge. As IT students, we should explore all aspects of technical work. It plays a vital role in solving real-life problems and creating opportunities in the software field.

Here, we are presenting a project on a different concept that we observed and experienced during our training. We have tried our best to justify this concept through the development of the **FreshConnect** system.

Index

Sr. No	Chapter Name	Page No
1.	Introduction	6
	1.1 Project Profile	7
	1.2 Project Introduction	8
2.	Environment Description	9
	2.1 Hardware And Software Requirements	10
	2.2 Tools And Technology Used	12
3.	Proposed System	18
	3.1 Scope	19
	3.2 Objective	20
	3.3 Constraint	21
	3.4 Expected Advantages	22
4.	System Planning	23
	4.1 Requirement specification	24
	4.2 Feasibility Study	28
	4.3 Life cycle Model	30

	4.4 Effort Distribution Diagram	31
	4.5 Task Dependency Diagram	32
	4.6 Project schedule Diagram	33
5.	System Model Architecture	35
	5.1 Use Case Diagram	36
	5.2 Class Diagram	41
	5.3 Uml Diagram	
6.	System Design	68
	6.2 Table Structure	73
	6.3 Screen Layout	84
7.	System Tasting	130
	7.1 Software Testing	131
	7.2 Test Case	136
8.	Limitation Of Proposed System	149
9.	References	151

Introduction

1

Chapter

1.1 Project Profile 1.2 Project Introduction

Project Profile

Project Title	Fresh connect
Project Type	e-commerce
Organization Name	Vivekanand College for Advanced computer And Information Science
Team Members	4
Technology	ReactJS, MongoDB, NodeJS, ExpressJS
Front End Tools	ReactJS
Back End	MongoDB, NodeJS
Project Guide	Ms. Nidhi Mehta
Submitted By	Ghantiwala Vivek. Patel Aman. Upale Rutvesh. Patel Bhavika.

Project Introduction

➤ Project Definition

This project presents a comprehensive Food Delivery and Hotel Management System developed as a full-stack web application. The system addresses the growing demand for efficient food delivery services while providing specialized management solutions for restaurants, hotel owners, and sellers. The application incorporates features such as user authentication, order management, inventory tracking, urgent sales promotion, and leftover food management to create a holistic platform that benefits all stakeholders in the food service ecosystem.

The system implements a modern technology stack including React.js, Node.js, Express.js, and MongoDB to create a scalable and responsive solution. Through this project, we demonstrate how technology can optimize food service operations, reduce food wastage, and enhance customer satisfaction in the competitive food delivery market. There are Five section in our Website.

1. Admin
2. Hotel owner
3. seller
4. User
5. Visitor

Environment Description 2 Chapter

2.1 Hardware and Software Requirements 2.2 Tools and Technology used

Hardware and Software Requirements

Hardware Requirement

○ Development Side:

Processor: i5 or Above

Ram: 8 GB or Above

Storage: 1 TB or Above

○ Client Side:

Processor: i3 or Above

Ram: 4 GB or Above

Storage: 512 GB or Above

Software Requirement

○ Development Side:

Front End: Visual Studio Code

Back End: Visual Studio Code, Chrome

Operating System: Windows 11 ○

Client Side:

Browser: any

Operating: Windows XP & above ○

Server Side:

Browser: any

Operating System: Windows 7 & above



Tools and Technology Used

○ Overview of React

React is JavaScript library used for building reusable UI components. According to React official documentation, following is the definition –

“React is a library for building composable user interfaces. It encourages the creation of reusable UI components, which present data that changes over time.” Lots of people use React as the V in MVC.

React abstracts away the DOM from you, offering a simpler programming model and better performance. React can also render on the server using Node, and it can power native apps using React Native. React implements one-way reactive data flow, which reduces the boilerplate and is easier to reason about than traditional data binding.

○ Features f React

- **JSX**

JSX is JavaScript syntax extension. It isn't necessary to use JSX in React development, but it is recommended.

- **Components**

React is all about components. You need to think of everything as a component. This will help you maintain the code when working on larger scale projects.

- **Unidirectional data flow and Flux**

React implements one-way data flow which makes it easy to reason about your app. Flux is a pattern that helps keeping your data unidirectional.

- **License**

React is licensed under the Facebook Inc. Documentation is licensed under CC BY 4.0.

○ Overview of MongoDB

MongoDB is a cross-platform, document-oriented database that provides, high performance, high availability, and easy scalability. MongoDB works on concept of collection and document.

- **Database**

Database is a physical container for collections. Each database gets its own set of files on the file system. A single MongoDB server typically has multiple databases.

- **Collection**

Collection is a group of MongoDB documents. It is the equivalent of an RDBMS table. A collection exists within a single database.

Collections do not enforce a schema. Documents within a collection can have different fields. Typically, all documents in a collection are of similar or related purpose.

- **Document**

A document is a set of key-value pairs. Documents have dynamic schema. Dynamic schema means that documents in the same collection do not need to have the same set of fields or structure, and common fields in a collection's documents may hold different types of data.

○ Overview of NodeJS

Node.js is a server-side platform built on Google Chrome's JavaScript Engine (V8 Engine). Node.js was developed by Ryan Dahl in 2009 and its latest version is v0.10.36

Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux

Node.js also provides a rich library of various JavaScript modules which simplifies the development of web applications using Node.js to a great extent.

- **Features of NodeJS**

Following are some of the important features that make Node.js the first choice of software architects.

- **Asynchronous and Event Driven**

All APIs of Node.js library are asynchronous, that is, non-blocking.

It essentially means a Node.js based server never waits for an API to return data.

The server moves to the next API after calling it and a notification mechanism of Events of Node.js helps the server to get a response from the previous API call.

- **Very Fast**

Being built on Google Chrome's V8 JavaScript Engine, Node.js library is very fast in code execution.

- **Single Threaded but Highly Scalable**

Node.js uses a single threaded model with event looping. Event mechanism helps the server to respond in a non-blocking way and makes the server highly scalable as opposed to traditional servers which create limited threads to handle requests.

Node.js uses a single threaded program and the same program can provide service to a much larger number of requests than traditional servers like Apache HTTP Server.

- **No Buffering**

Node.js applications never buffer any data. These applications simply output the data in chunks.

- **License**

Node.js is released under the MIT license.

○ Overview of ExpressJS

ExpressJS is considered a minimal as well as flexible web application framework of NodeJS, which gives robust features for use of the web as well as mobile applications.

ExpressJS is also considered an open-source framework, and it was developed and maintained by the foundation of NodeJS.

The developer of Express was TJ Holowaychuk and is currently been maintained by the Node.JS foundation as well as varied contributors of open source.

• Features of ExpressJS

Following are some of the important features that make ExpressJS the first choice of software architects.

• Routing

ExpressJS provides a simple and organized way to handle different HTTP requests and map them to specific actions. This allows you to define different paths for your website or API and specify what should happen when a user visits each one.

• Middleware

ExpressJS uses middleware to process incoming requests before they reach their final destination. This allows you to perform tasks such as authentication, validation, and logging in a reusable and modular manner.

• Templating

ExpressJS includes built-in support for several popular Templating engines, such as EJS and Jade, which make it easier to create dynamic and interactive pages.

• Database Integration

ExpressJS can be easily integrated with popular databases such as MongoDB and MySQL, making it simple to store and retrieve data in your web application.

• Scalable

ExpressJS is designed to be scalable and efficient, making it suitable for building large and complex web applications.

- **License**

Express.js released as free and open-source software under the MIT License.

○ Overview of Stripe

- Stripe, Inc. is an Irish-American financial services and software as a service (SaaS) company dual-headquartered in South San Francisco, California, United States and Dublin, Ireland.
- The company primarily offers payment-processing software and application programming interfaces for e-commerce websites and mobile applications.
- Irish entrepreneur brothers John and Patrick Collison founded Stripe in Palo Alto, California, in 2009.
- Stripe issue cards and simulate purchases using your own Stripe integration in test mode.
- This allows us to test your integration before we go live without having to make real purchases. We can do external purchases.

Proposed System

3

Chapter

3.1 Scope 3.2 Objective 3.3 Constraint 3.4 Expected Advantages

Scope

○ Scope of System

The scope of the project encompasses:

User Management: Registration, authentication, and profile management for different user roles

- + Order Management: Order placement, tracking, history, and fulfilment
- + Restaurant Management: Menu creation, inventory tracking, and order processing
- + Promotions Management: Urgent sales and leftover food listings
- + Payment Processing: Multiple payment method integration and transaction recording
- + Communication: Notifications and updates for order status changes

Objective

○ Main Aim of the System

- + Our main aim is to provide best Service as per the requirements of the customer.

O Objective of The System

The primary objectives of the Food Delivery and Hotel Management System are:

1. To develop an integrated platform that serves customers, restaurants, and sellers
2. To streamline the food ordering and delivery process
3. To implement efficient inventory management for restaurants
4. To provide a mechanism for promoting urgent sales and managing leftover food
5. To create comprehensive analytics dashboards for data-driven decision-making
6. To ensure secure user authentication and data protection
7. To design a responsive interface accessible across multiple devices



The constraint associated in the development of any software system include those that are common to all system. The common constraints may include lack of managerial participation tighter development schedule etc.

Some of the constraints are listed below with which we have developed this system:

- User mistake can be turned into the improper answer.
- The common constraints may include lack of managerial participation tighter development schedule frequently working for management etc.
- It has some specification need to run this software because is developed in the ReactJS.
- Net connection and other web related software are required.

Expected Advantages

High Security:

The system implements robust security mechanisms to protect user data and transactions. Only authorized personnel can access and modify data stored in the system. Multiple security layers including JWT authentication, role-based access control, and encrypted communications ensure that unauthorized users cannot tamper with sensitive information.

Time Efficiency:

All processes in our system are computerized and automated, significantly reducing the time required for transactions compared to traditional manual systems. Tasks like inventory management, order processing, and payment handling that would typically take hours can be completed in minutes or seconds.

Enhanced Accuracy:

By minimizing human intervention in critical processes, the system drastically reduces the potential for errors. Calculations and data manipulations are performed automatically by the application, ensuring that the information generated is accurate and reliable. This is especially important for financial transactions, inventory management, and order processing.

24/7 Accessibility:

The online hotel room and flight ticket booking functionality is accessible around the clock. Users can make bookings at any time that suits them, even at unusual hours like 7 a.m., without having to wait for business hours. This constant availability significantly enhances user convenience.

Improved User Experience:

Users benefit from the comfort of taking their time while browsing available options. The intuitive interface allows customers to explore various hotels and flight options, compare prices, check availability, and make informed decisions without feeling rushed, leading to a more satisfying booking experience.

Wider Selection Options:

The system provides users with a comprehensive view of available hotel rooms and flight tickets, allowing them to choose from a wider variety of options than would typically be possible through traditional booking methods. This expanded selection empowers users to find exactly what they need.

Simplified Decision Making:

With all necessary information presented clearly and comprehensively, users can easily evaluate different options based on their preferences for pricing, timing, location, and amenities. This simplifies the decision-making process and helps users feel confident in their choices.

System Planning

4

Chapter

4.1 Requirement specification 4.2 Feasibility Study 4.3 Life cycle Model 4.4 Effort Distribution Diagram 4.5 Task Dependency Diagram 4.6 Project schedule Diagram

Requirement specification

Requirement Gathering:

- Our e-Commerce Food Platform is an online food ordering system through which users can purchase regular and discounted food products.
- Our e-Commerce Food Platform offers to buy food products online rather than going to physical stores and purchasing it.
- Ease of analyzing food products is one of the biggest tools for any e-commerce food platform to approve only valid and quality-worthy products for customers.
- Our e-Commerce Food Platform not only gives information to buy food products but it also provides real time consultation with well detailed user reviews.
- This website provides facility for sellers to list their food products through the web app and manage through their account.
- The e-Commerce Food Platform provides online payment facility, by using our online payment service users can easily make their purchases.
- The e-Commerce Food Platform has also wish-list feature here user can wish-list his/her favourite food products.
- The platform includes an urgent sales section where soon-to-expire food products are offered at discounted prices to reduce food waste.
- Users can filter products by categories, price range, dietary requirements, and availability.
- The system maintains detailed product information including ingredients, nutritional values, and expiry dates.
- Sellers receive notifications for new orders and can track their inventory in real-time.

There are Five Section in our applications:

- 1)Admin
- 2)Hotel owner
- 3)Seller
- 4)User(Customer)

† Admin

- Admin can activate and deactivate client.
- Admin can activate and deactivate user.
- Admin can edit his profile.
- Admin receive payment from the users.
- Admin gives the payment to clients.
- Admin can view the bookings.
- Admin can view the feedbacks.
- Admin can view the ratings.
- Admin can change his password.
- Admin can login the website.

† Hotel owner

† Hotel Management

- Hotel can register on the platform.
- Hotel can update profile information.

- Hotel can add address details.
- Hotel can upload logo and cover image.
- Hotel can add a description.
- Hotel can submit verification documents.
- Hotel can view reviews and ratings.
- Hotel can track number of reviews.
- Hotel can manage role and hotel type.
- Hotel can see account creation and update timestamps.
- Hotel status is shown as verified/unverified.

† seller

- Seller can register on the platform.
- Seller can update name and business type.
- Seller account is marked as verified or not.
- Seller can view account creation date.
- Seller can specify type (e.g., vegetable, fruit, etc.).
- Seller can manage location details.

† user

- User can register with name, email, and password.
- User can update phone number and address.
- User can add a profile description.
- User can upload a logo and cover image.

- User account is marked as verified or not.
- User can receive and view ratings.
- User can view number of reviews received.
- User can upload verification documents.
- User can view submitted reviews.
- User can be marked as hotel (isHotel = true).
- User is assigned a specific role (e.g., admin, hotel, seller).
- User data includes account creation and update timestamps

Feasibility Study

The Food Delivery and Hotel Management System does not require advanced or cutting-edge technology to implement successfully. The system primarily necessitates database interaction and internet accessibility, both of which are widely available and well-established technologies.

Key technical feasibility aspects:

Web and Database Technologies:** The project employs established technologies (React.js, Node.js, Express.js, and MongoDB) that are mature, well-documented, and have large community support.

Online Booking Functionality: The implementation of hotel room and flight ticket booking systems is achievable using existing web technologies and payment gateways. These features will provide reliable navigation and decision-making tools for users.

User Interface: The application uses modern frontend frameworks (React.js with Material UI and Tailwind CSS) that facilitate responsive design across different devices.

Development Timeline: The core system can be developed within a three-month timeframe, excluding testing, validation, and verification periods. This timeline is realistic given the scope and the technologies chosen.

API Integration: The system can easily integrate with third-party services for features like payment processing, email notifications, and map services.

Scalability: The chosen architecture and technologies support horizontal scaling to accommodate future growth in user base and transaction volume.

Based on these factors, the project is deemed technically feasible with the available resources, knowledge, and technology stack.

† Economical Feasibility

In this project, we will require to have an internet connection for better online website.

Such at the developer end it needs the good server with high capacity of RAM and CPU processors so can it can handle lots of members at time online communications, but as it will be web-enabled we do not have any extra cost of setting up a network.

This is also feasible economically.

† Operational Feasibility:-

The new system can be beneficial only if it satisfies the organization requirements; in such a way that resource utilization and optimum outcome is justified

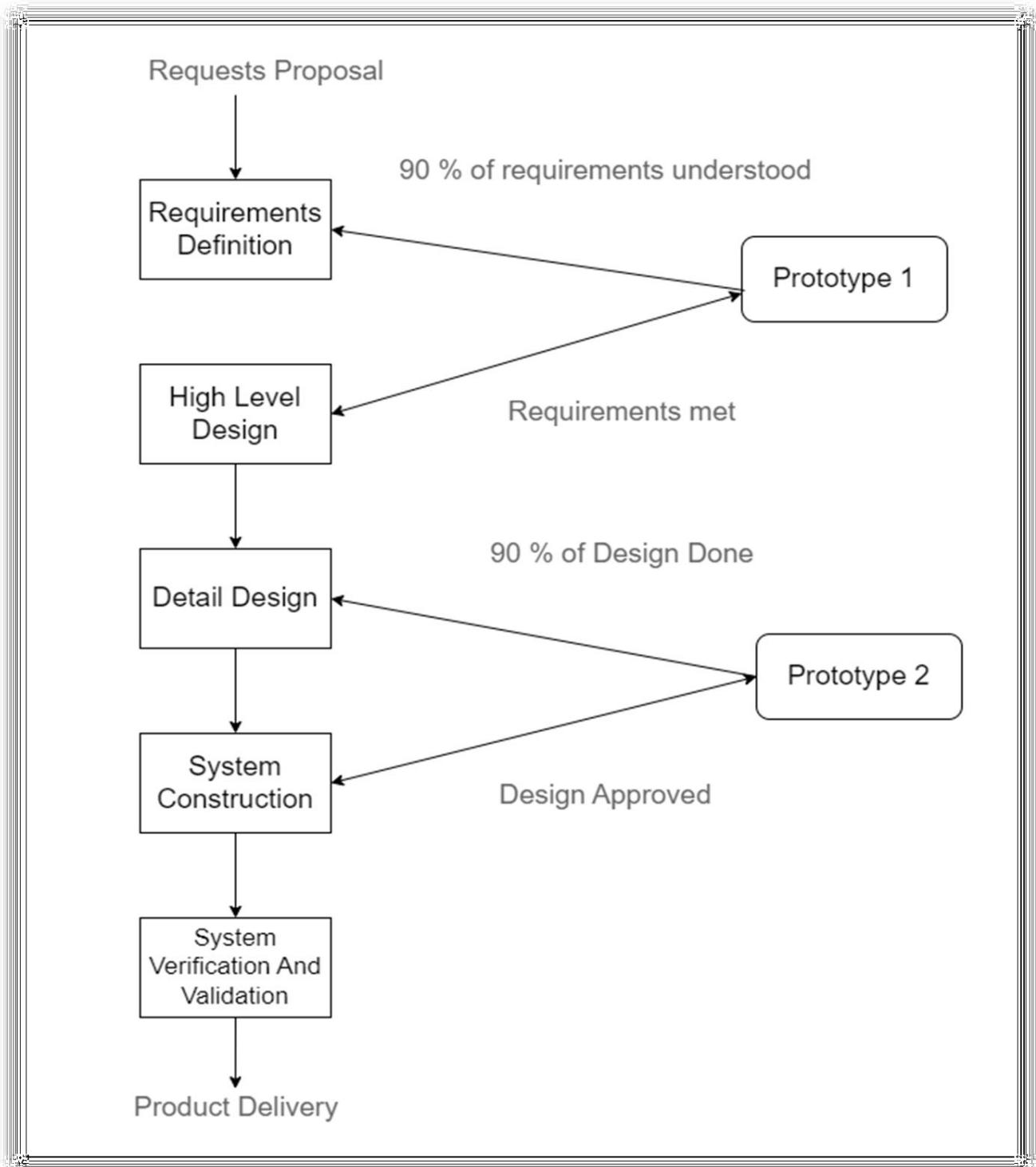
A new system should not only be robust but should also be able to work simultaneously with other systems. Operational feasibility means that new system should not affect any existing system during the development phase or even in the implementation phase.

Following are some points underlying the operational feasibility of the system- As the development proceeded many doubts got cleared out.

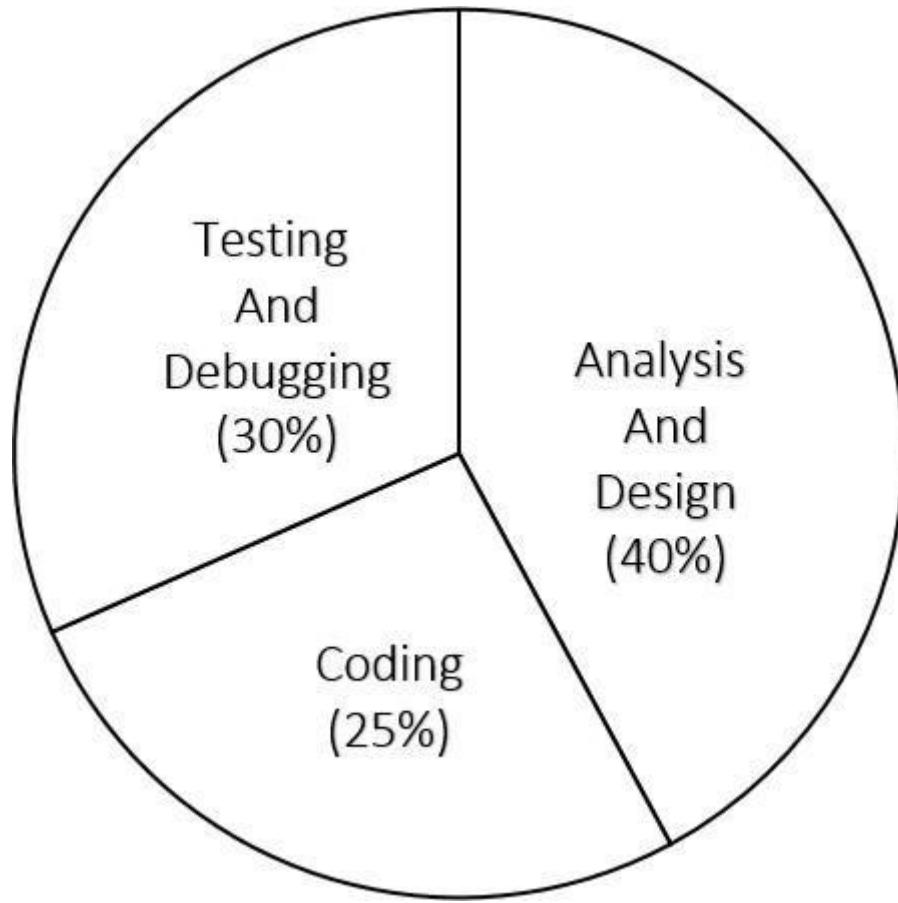
Our project guide – Ms. Nidhi Mehta guided us to take proper care and check for the operational feasibility of the system.

Efforts were made to optimize the human efforts in data collection, storage, retrieval, security and presentation.

Life cycle Model

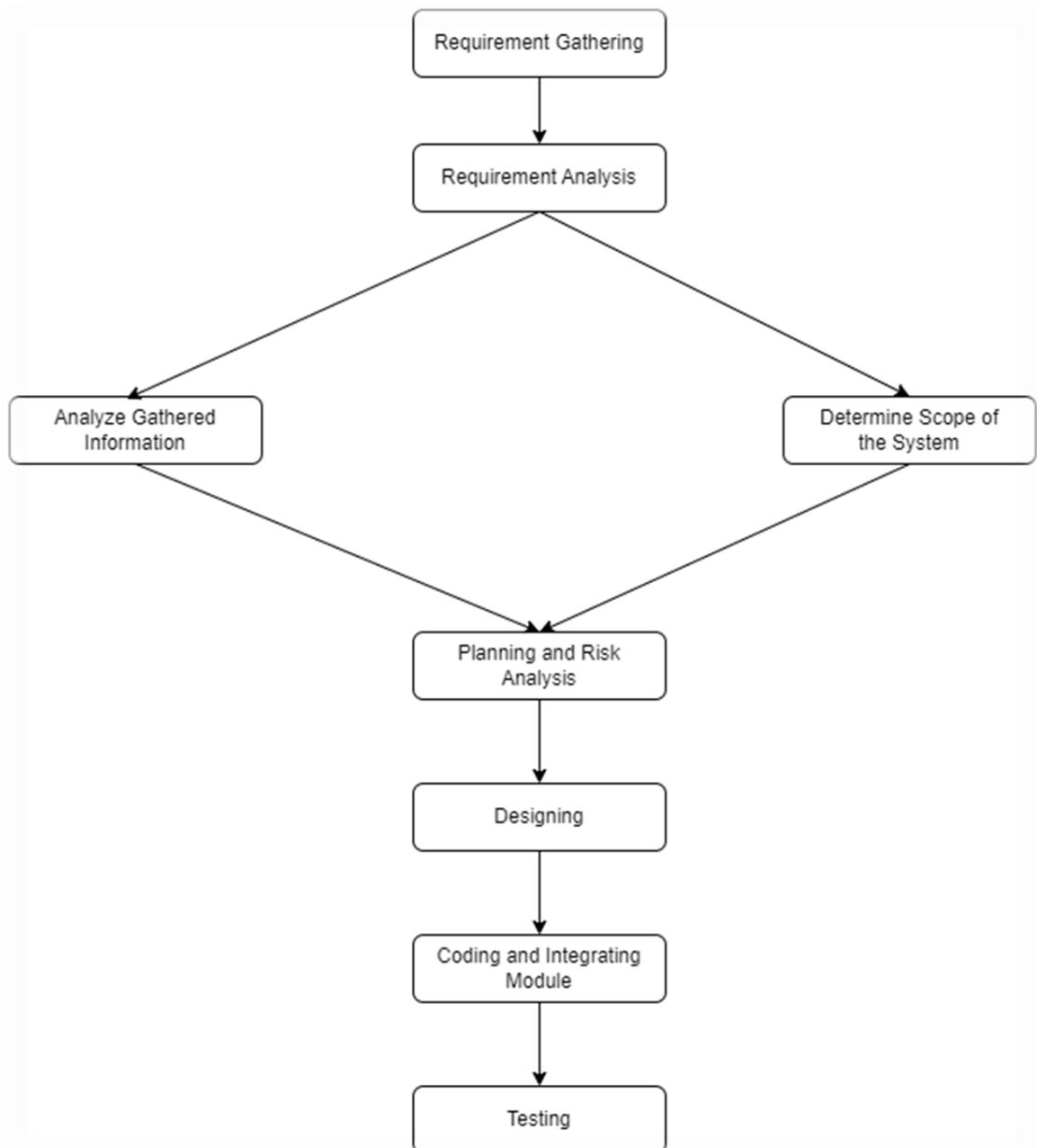


Effort Distribution



Project Planning	5%
Analysis and Design	40%
Coding	25%
Testing and Debugging	30%

Task Dependency





Project Schedule

5. Coding and Integrating modules													
Implement logic for diff modules													
Integrating code with backend													
Integrating dependent modules													
Milestone: Coding completed													
6. Testing of developed system													
Milestone: Finalized													
7. Documentation													
Milestone: Documentation completed													

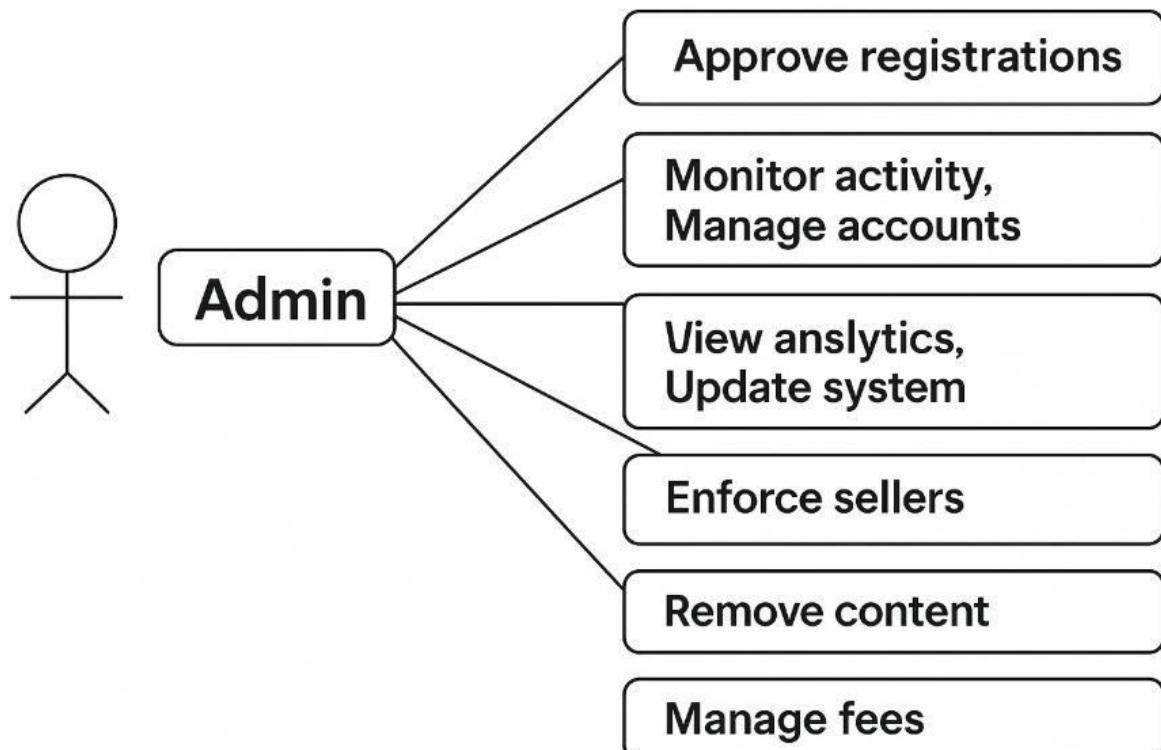
System Model Architecture 5

Chapter

5.1 Use Case Diagram 5.2 Class Diagram 5.3 Uml Diagram

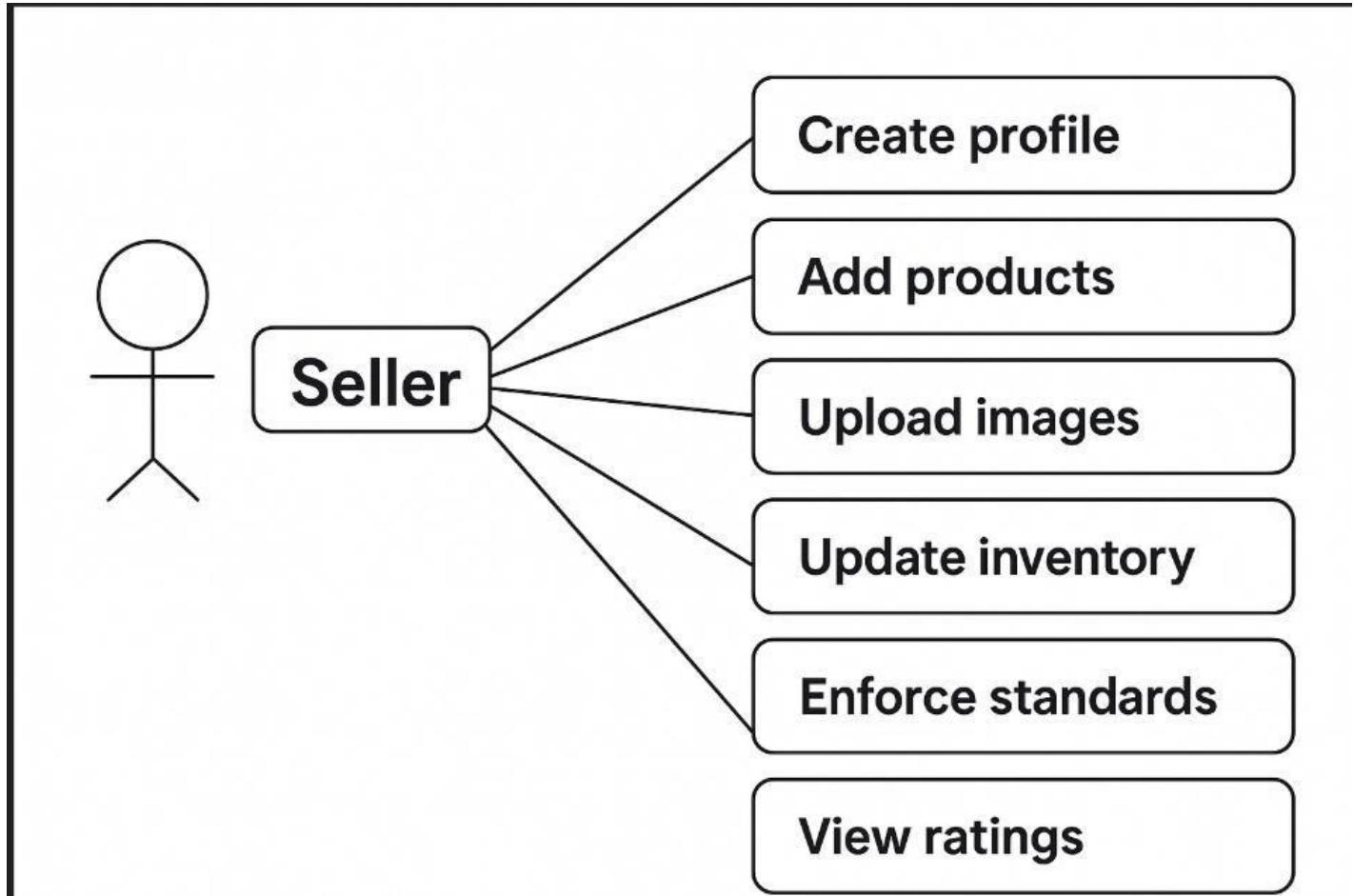
Use case Diagram

Admin

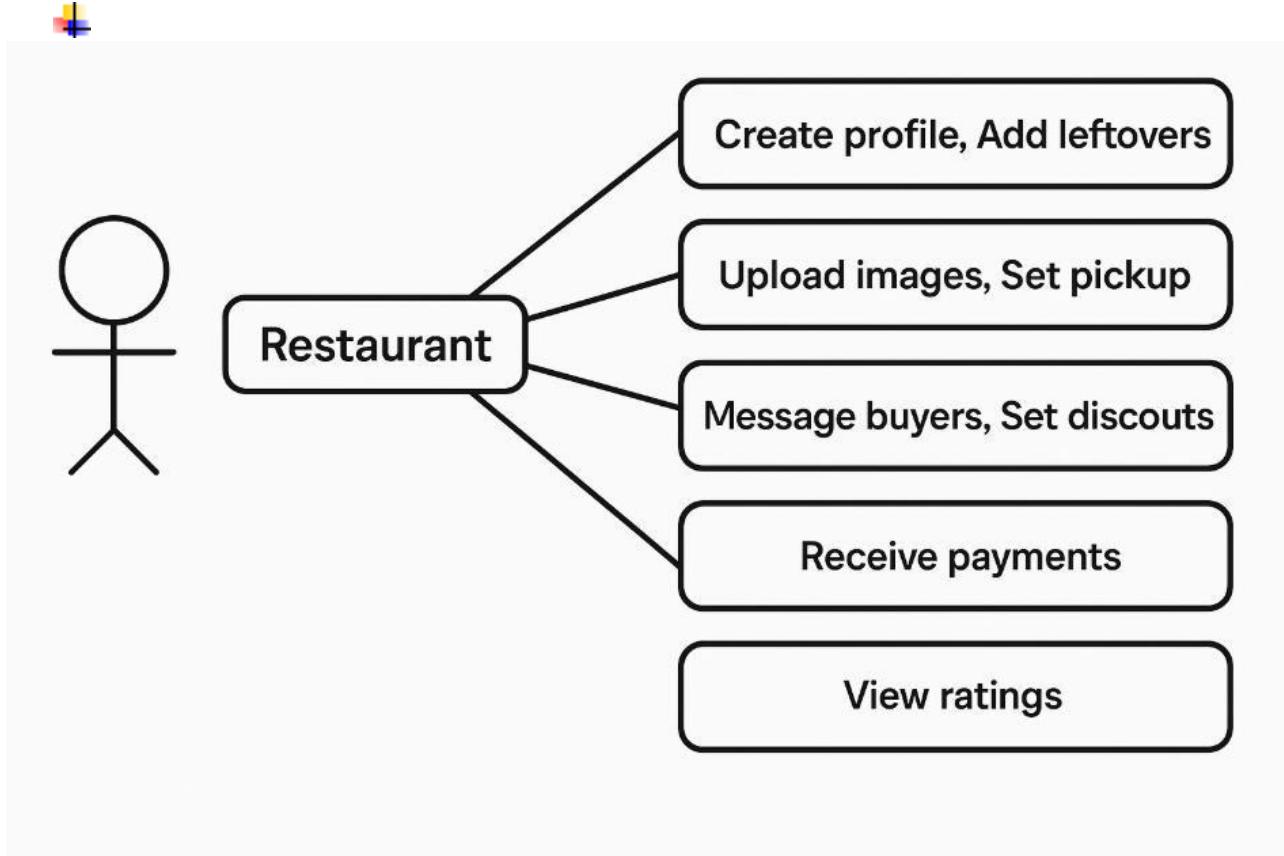




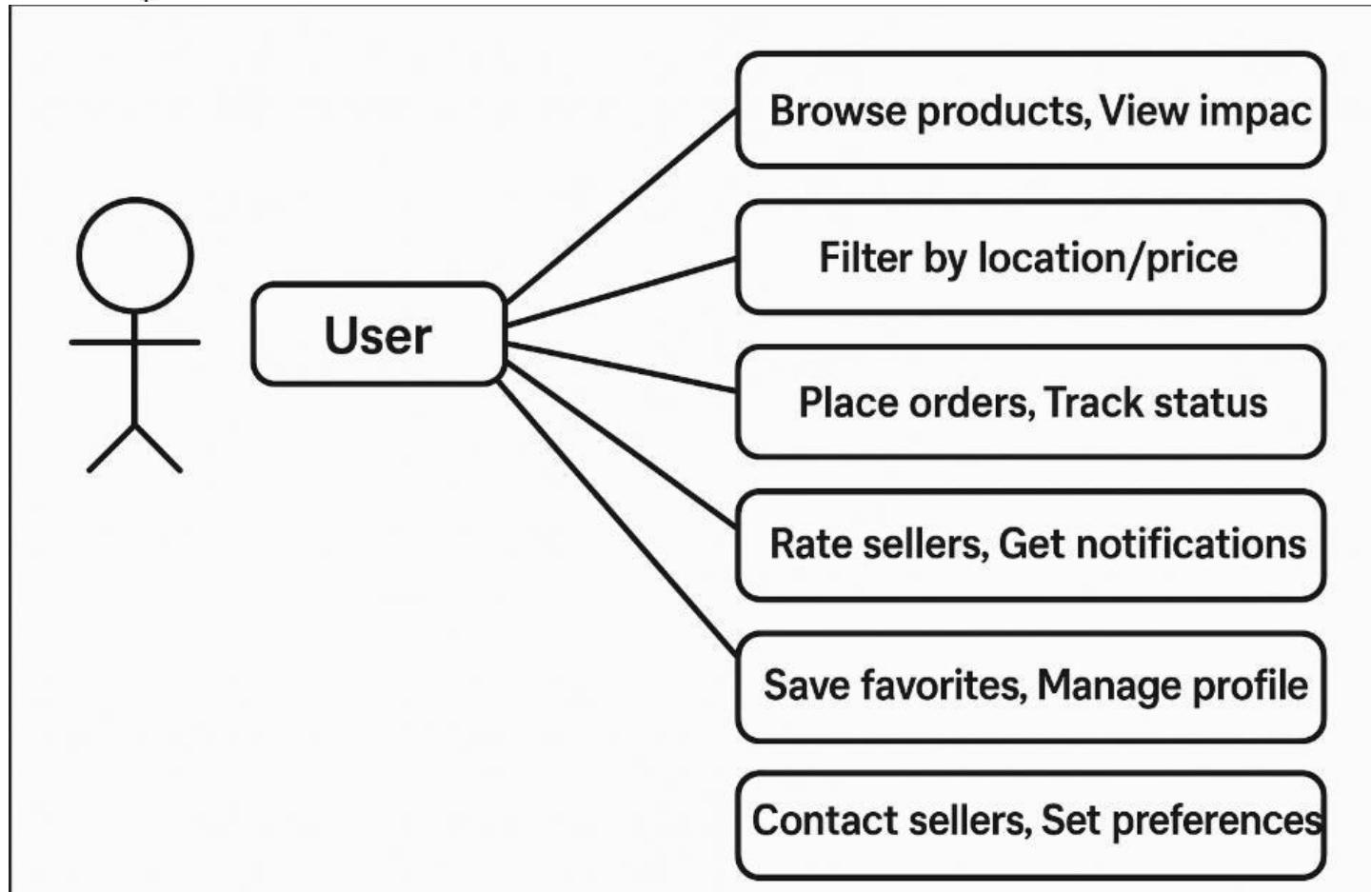
Seller



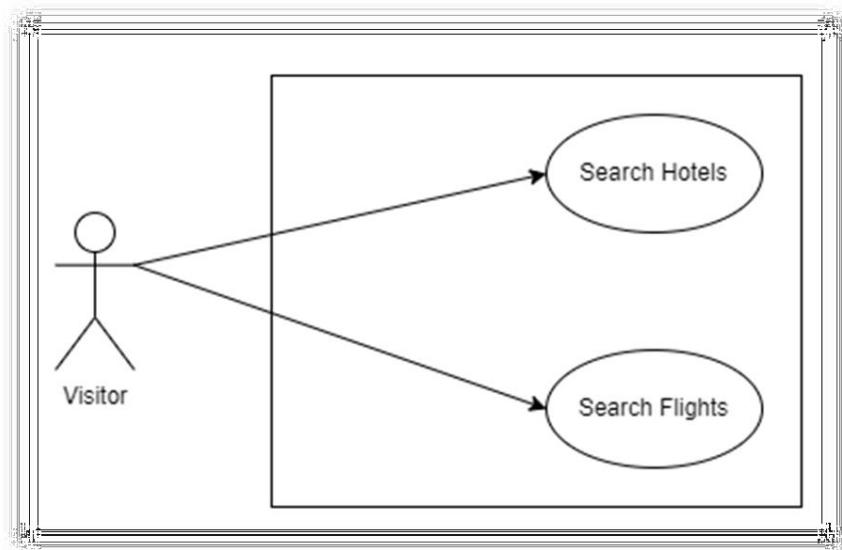
Hotel Management :



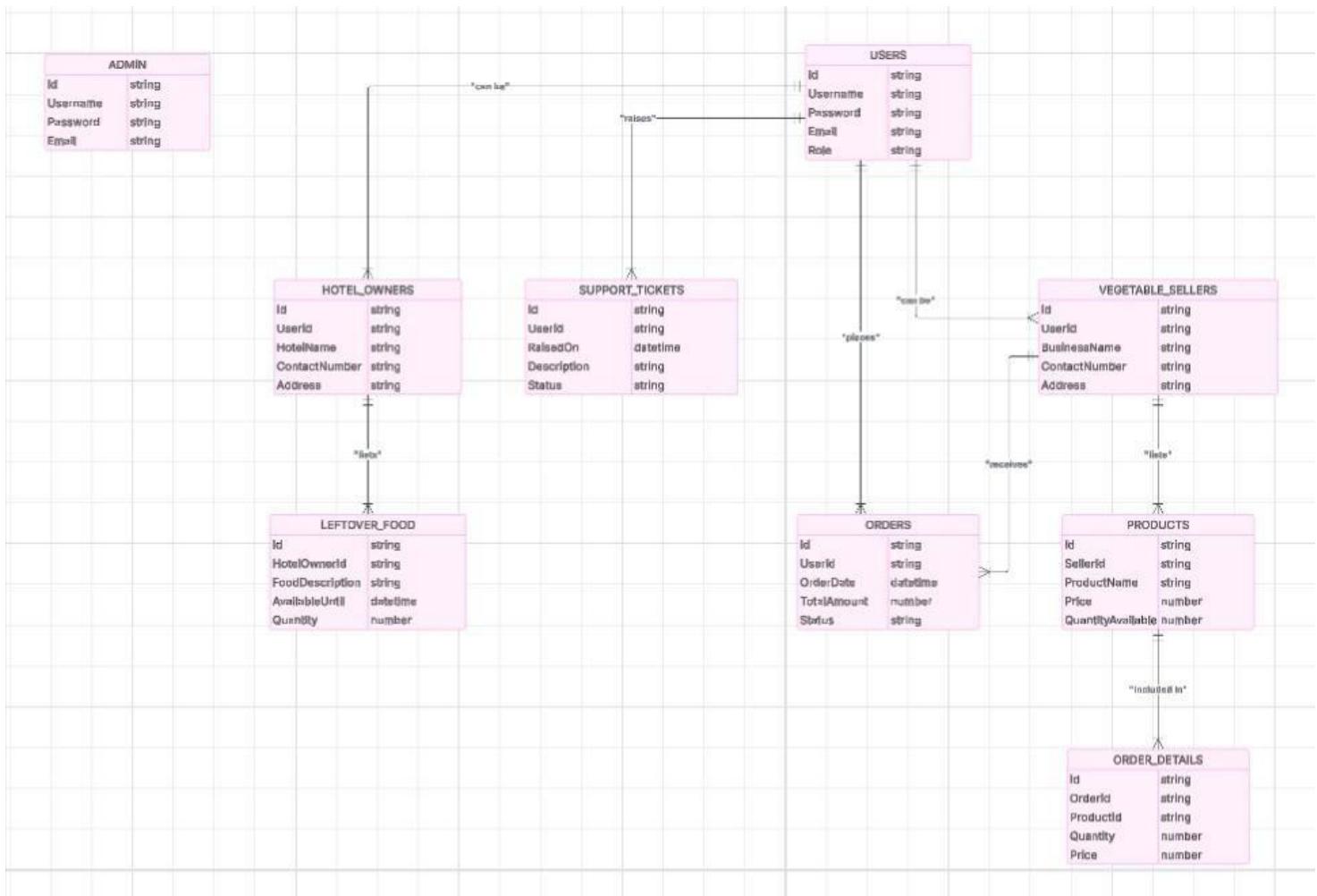
User



Visitor

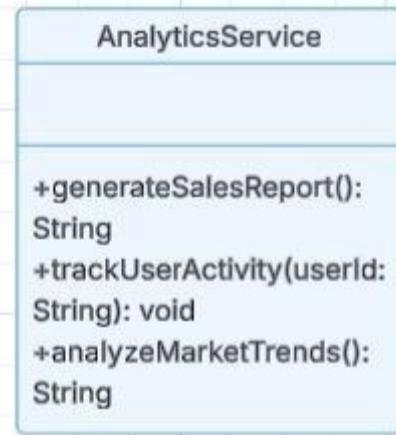


ER Diagram

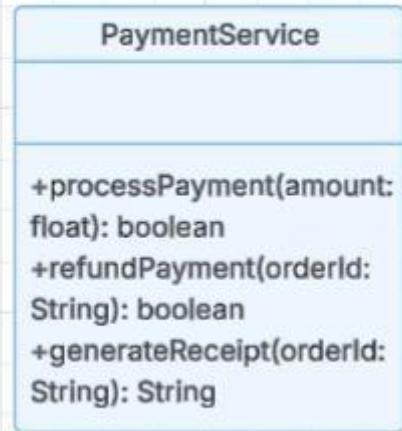


Class Diagram

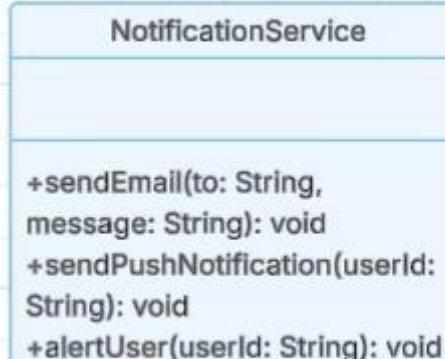
Analytics service :



Payment service:



Notification service :

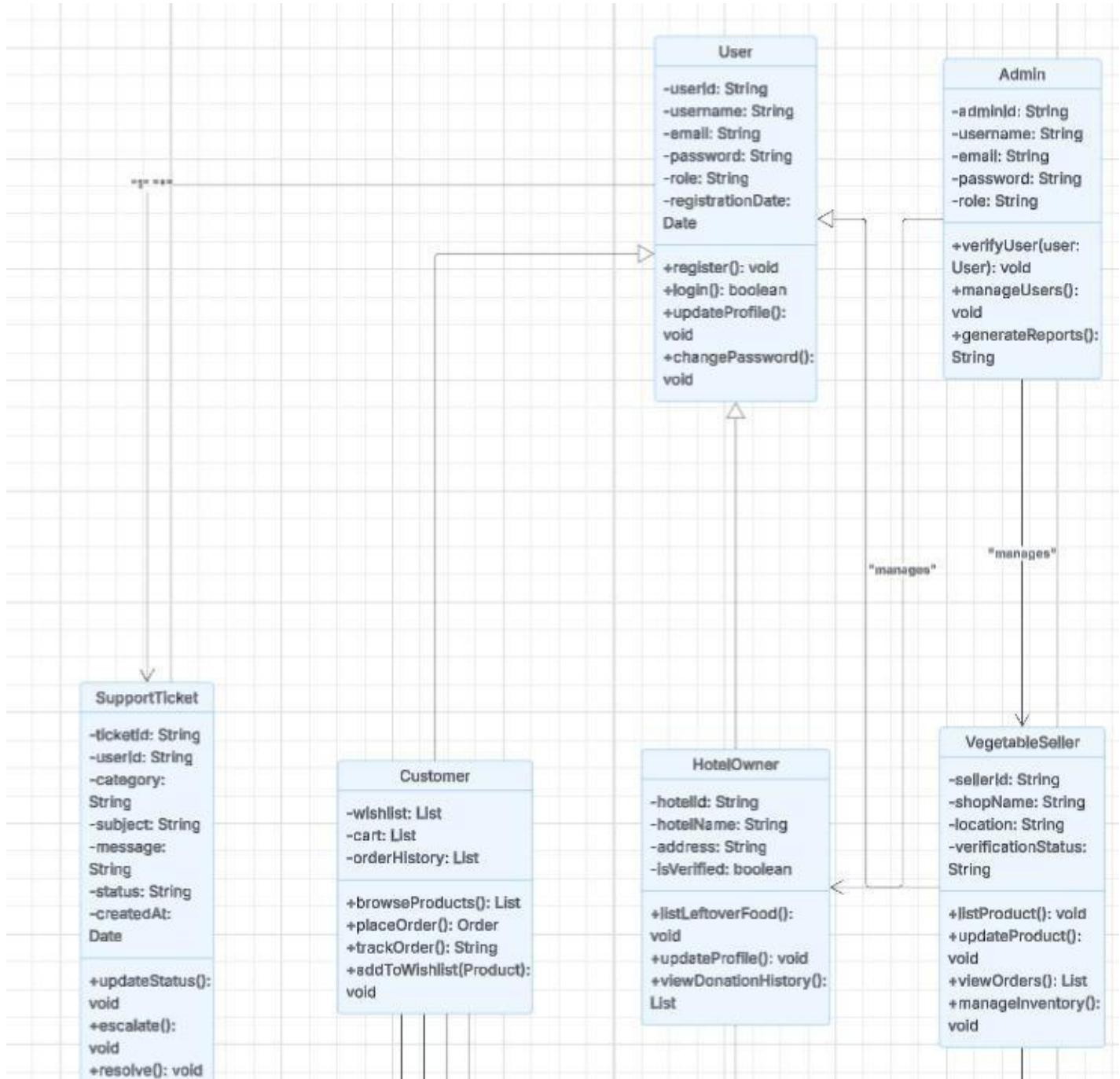


Authontication service :

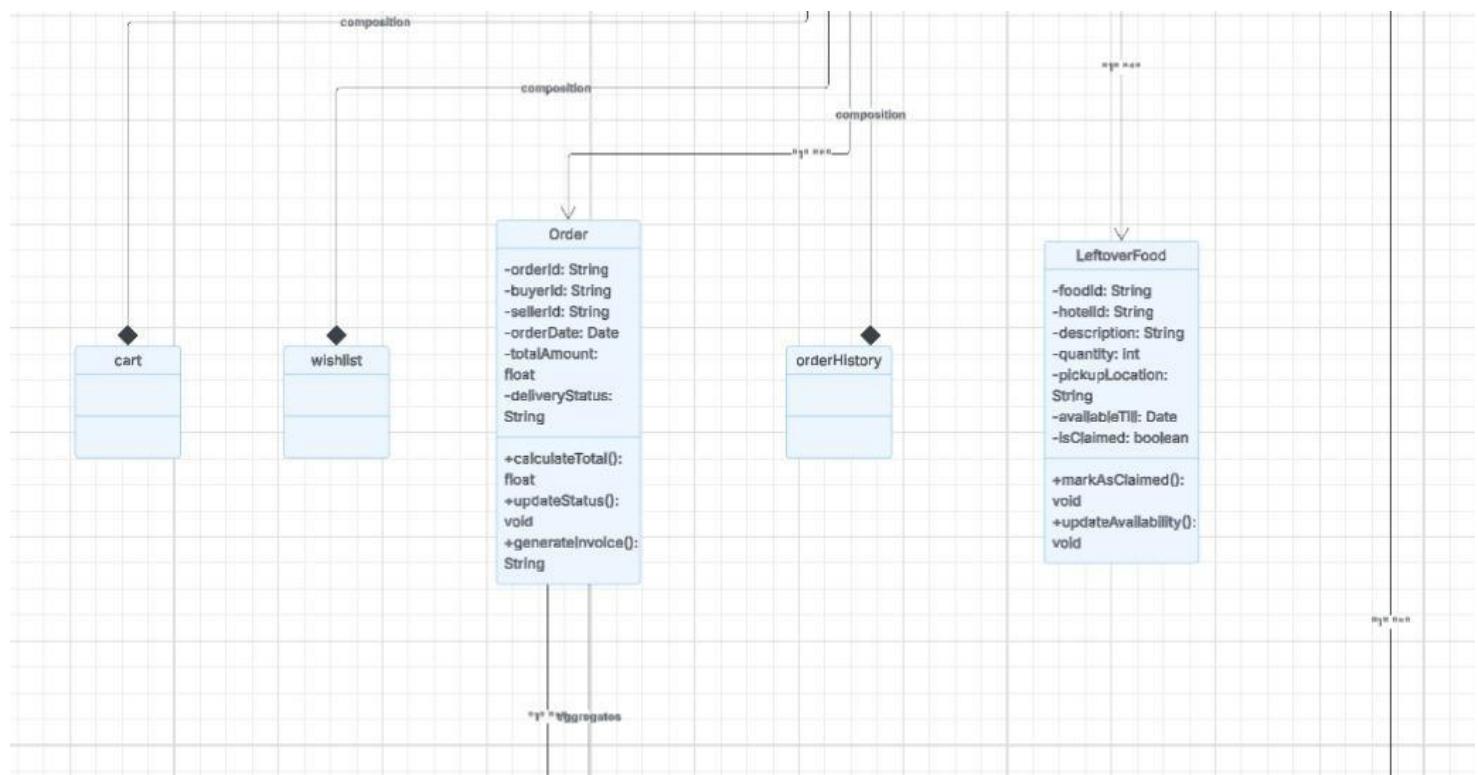


Class Diagram

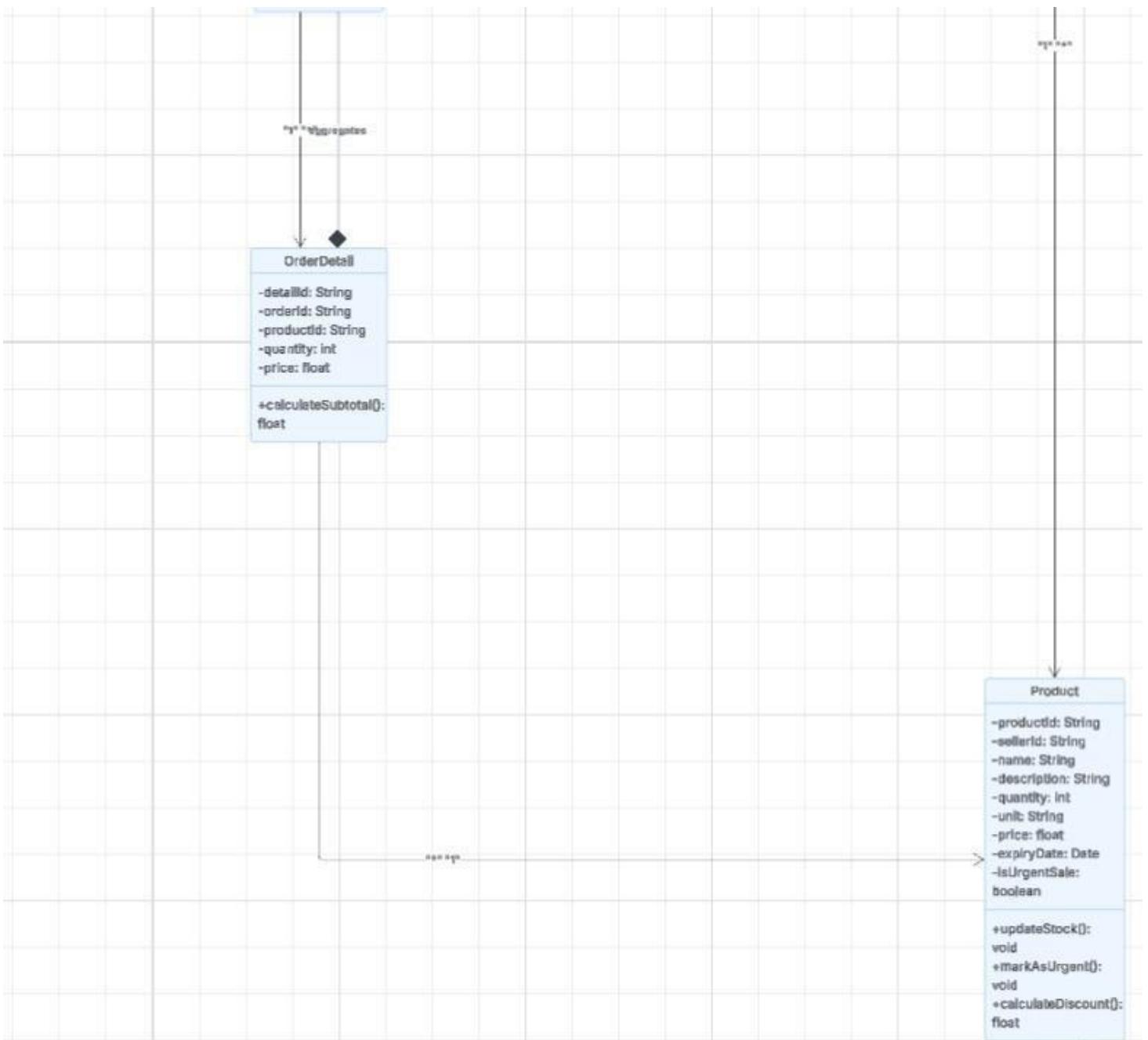
User,Admin,Support ticket,Customer,Hotel owner,Vegetable seller :



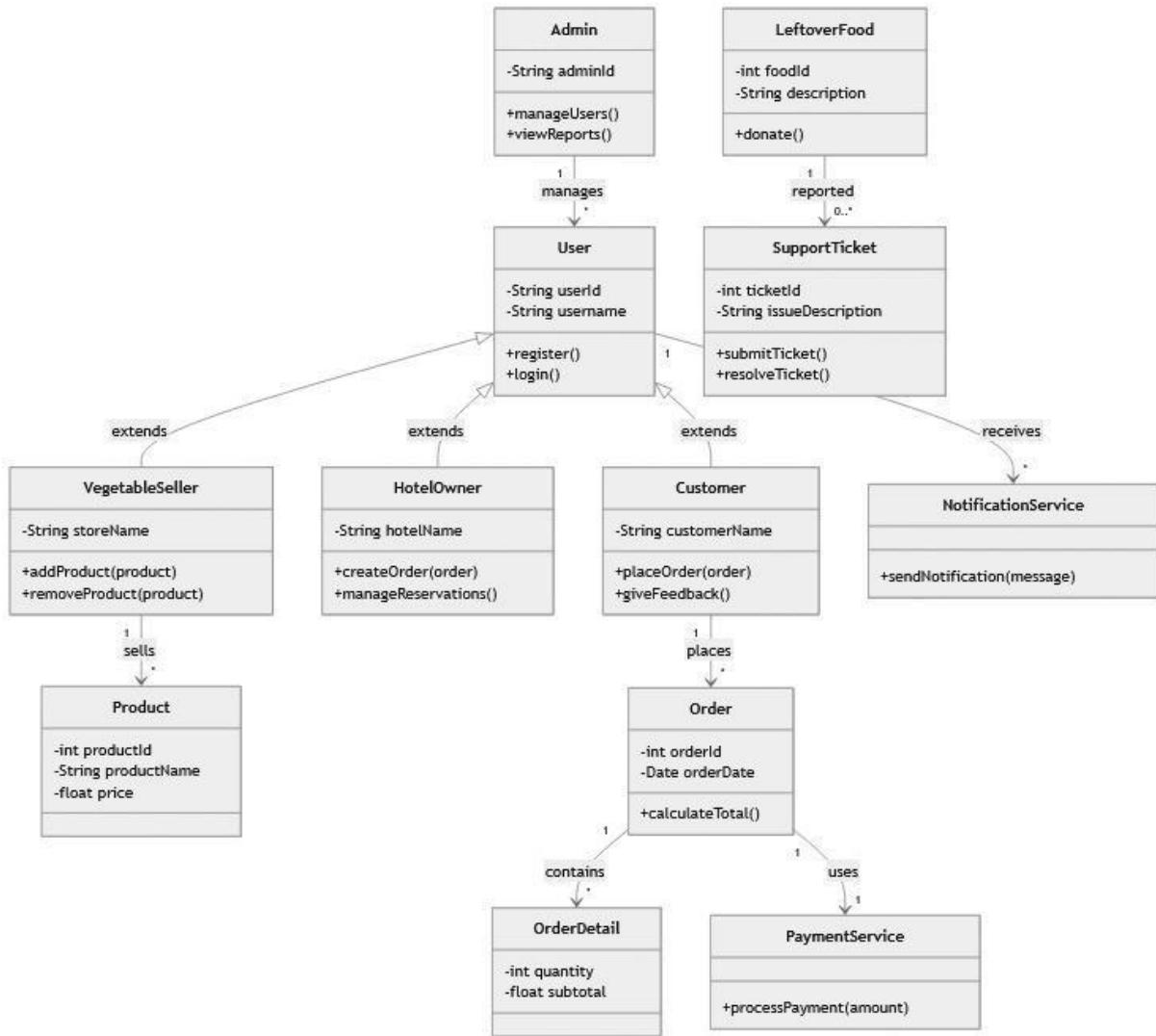
Cart,history,Wrder,Wishlist,Order history,Leftover food :



Order details,product:



Uml Diagram



System Design

6

Chapter

6.1 Table Structure 6.2 Screen Layout

Table Structure

Table Name: tblAdmin

This table contains all the information about admin like User Name, Email and Password.

Sr No.	Field Name	Data type	Constraint	Not Null	Description
1.	Admin_ID	Varchar	PK (Auto Increment)		Admin's Unique ID
2.	Username	Varchar			Admin's user name
3.	Email	Varchar			Admin's email
4.	Password	Varchar			Password
5.	Img	Varchar		Null	Admin's photo

Table Name: Production

This Collection contains all the information about Hotel management like Hotel name, Email and Password.

Field Name	Data type	Constraint	Description
_ID	Object	PK (Auto Increment)	Indicates User ID.
name	Varchar	Required	Stores the full name of the user.
imageurl	Varchar	Optional	Stores the URL of the product image.
price	Number	Required	Stores the product price
category	Varchar	Required	Indicates the category ID of the product
instock	Boolean	Default: true	Indicates the category ID of the product.
description	Varchar	Optional	Stores the description of the product.
__V	Number	System Field	Stores the versioning field managed by MongoDB.
createdAt	Object	Required	Stores the timestamp when the product was created.
discount	Number	Optional	Stores the discount percentage on the product.
image	Varchar	Optional	Stores the image of the product.
isFresh	Boolean	Default:false	Indicates if the product is fresh.
isOrganic	Boolean	Default:false	Indicates if the product is organic.
isVerified	Boolean	Default:false	Indicates if the product is verified.
numReviews	Number	Default: 0	Stores the total number of reviews.
rating	Number	Default: 0	Stores the average rating of the product.
reviews	Object	Optional	Stores the customer reviews for the product.
seller	Object	Required	Stores the seller's details or reference.
sellerType	Varchar	Required	Indicates the type of seller (e.g., individual, business).
stock	Number	Required	Stores the total stock quantity.

unit	Varchar	Required	Stores the measurement unit of the product (e.g., kg, liter).
updatedAt	Object	Required	Stores the timestamp when the product was lastupdated

urgentSale	Boolean	Default:false	Indicates if the product is on urgent sale.
------------	---------	---------------	---

Table Name: End User

This table contains all the information about User like Name, Email and Password.

Field Name	Data type	Constraint	Description
_id	Object	PK (Auto Increment)	Indicates User ID.
name	Varchar	Required	Stores the full name of the user.
phone	Varchar	Required,Unique	Stores the email address of the user.
Email	Varchar	Required	Stores the hashed password for authentication.
password	Varchar	Optional	Stores the contact phone number of the user.
role	Varchar	Required	Indicates the role of the user (e.g., admin,user).
isverified	Boolean	Default:false	Indicates whether the user's email is verified.
document	Object	Optional	Stores references to userrelated documents.
createdAt	Object	Required	Stores the timestamp when the user was created.
updatedAt	Object	Required	Stores the timestamp when the user was last updated.
__v	Number	System Field	Stores the versioning field managed by MongoDB.

Table Name:

Urgent Sale

This table contains all the information about Urgent sale food like product,price,discount price

Field Name	Data type	Constraint	Description
_id	Object	PK	Indicates Urgent Sale ID.
seller	Object	Required	Stores the seller's details or reference.
product	Object	Required	Stores the product details or reference.
Name	Varchar	Required	Stores the name of the urgent sale product
originalPrice	Number	Required	Stores the original price of the product..
discountedPrice	Number	Required	Stores the discounted price of the product.
stock	Number	Required	Stores the available stock quantity.
unit	Varchar	Required	Stores the measurement unit (e.g., kg, liter).
expiryDate	Object	Required	Stores the expiry date of the urgent sale offer.
description	Object	Optional	Stores the description of the urgent sale item.
image	Varchar	Optional	Stores the image of the product.
isActive	Boolean	Default:true	Indicates if the urgent sale is active.

createdAt	Object	Required	Stores the timestamp when the urgent sale was created.
updatedAt	Object	Required	Stores the timestamp when the urgent sale was last updated.
__v	Number	System Field	Stores the versioning field managed by MongoDB.

Table Name:

This all the information about

MenuItem

table contains

Hotel Rooms like

Category, Maximum Adults, Children and Rate.

Field Name	Data type	Constraint	Description
<code>id</code>	Object	PK	Indicates Menu Item ID.
<code>hotel</code>	Object	Required	Stores the reference to the hotel or restaurant.
<code>name</code>	Varchar	Required	Stores the name of the menu item.
<code>description</code>	Varchar	Optional	Stores the description of the menu item.
<code>price</code>	Number	Required	Stores the price of the menu item.
<code>image</code>	Varchar	Optional	Stores the image URL of the menu item.
<code>category</code>	Varchar	Required	Indicates the category of the menu item.
<code>isVegetarian</code>	Boolean	Default:false	Indicates if the menu item is vegetarian.
<code>isVegan</code>	Boolean	Default:false	Indicates if the menu item is vegan.
<code>isGlutenFree</code>	Object	Default:false	Indicates if the menu item is gluten-free.
<code>ingredients</code>	Number	Default:false	Stores the list of ingredients used.
<code>preparationTime</code>	Boolean	Optional	Stores the preparation time in minutes.
<code>isPopular</code>	Number	Required	Indicates if the menu item is popular.
<code>rating</code>	Number	Default:false	Stores the average rating of the menu item.
<code>numReviews</code>	Object	Default:0	Stores the total number of reviews
<code>createdAt</code>	Object	Default:0	Stores the timestamp when the menu item was created.
<code>updatedAt</code>	Number	Required	Stores the timestamp when the menu item was last updated
<code>IsAvailable</code>	Boolean	Required	Indicates if the menu item is available.

__v	Number	System field	Stores the versioning field managed by MongoDB
-----	--------	--------------	---

Table Name:

This table contains all the information about

Purchase

table contains Purchase details like hotel,supplier,items.

Field Name	Data type	Constraint	Description
_id	Object	Primary key	Indicates Purchase ID.
user	Object	Required	Stores the reference to the user who made the purchase.
hotel	Object	Required	Stores the reference to the hotel associated with the purchase.
supplier	Varchar	Required	Stores the name of the supplier.
items	Object	Required	Stores the list of purchased items.
totalAmount	Number	Required	Stores the total amount of the purchase.
paymentMethod	Varchar	Required	Stores the payment method used (e.g., cash, card).
paymentStatus	Varchar	Default:false	Indicates the status of the payment (e.g., pending, completed)
notes	Varchar	Optional	Stores any additional notes about the purchase.
receiptNumber	Varchar	Required	Stores the unique receipt number for the purchase.
createdAt	Object	Required	Stores the timestamp when the purchase was made.
updatedAt	Object	Required	Stores the timestamp when the purchase was last updated.

Inventories

table contains
quantity,units,price.

Inventories like

Field Name	Data type	Constraint	Description
_id	Object	PK	Indicates Inventory ID.
hotel	Object	Required	Stores the reference to the hotel.
name	Varchar	Required	Stores the name of the inventory item.
category	Varchar	Required	Indicates the category of the inventory item.
quantity	Number	Required	Stores the quantity available in stock.
unit	Varchar	Required	Stores the measurement unit (e.g., kg, liter).
price	Number	Required	Stores the price of the inventory item.
supplier	Object	Required	Stores the reference to the supplier.
purchaseDate	Object	Required	Stores the date when the item was purchased.
minStocklevel	Number	Required	Stores the minimum stock level before alert.
isLowStock	Boolean	Default:false	Indicates if the stock is below the minimum level.
isOutofstock	Boolean	Default:false	Indicates if the inventory item is out of stock.
location	Varchar	Optional	Stores the storage location of the inventory item.
notes	Varchar	Optional	Stores additional notes about the inventory item.
images	Object	Optional	Stores image references of the inventory item.
IsleftoverFood	Boolean	Default:false	Indicates if the item is leftover food.

Table Name:

This table contains all the information about

createdAt	Object	Required	Stores the timestamp when the menu item was created.
updatedAt	Object	Required	Stores the timestamp when the menu item was last updated.
v	Number	System field	Stores the versioning field managed by MongoDB.

Field Name	Data type	Constraint	Description
_id	Object	PK	Indicates Image Storage ID.
data	Object	Required	Stores the binary data of the image.
contentType	Varchar	Required	Stores the MIME type of the image (e.g., image/png,).
filename	Varchar	Required	Stores the original name of the uploaded file.
user	Object	Required	Stores the reference to the user who uploaded the image.
storageType	Varchar	Required	Indicates the type of storage used (e.g., local, cloud).
uploadDate	Object	Required	Stores the timestamp when the image was uploaded.

__v	Number	System Field	Stores the versioning field managed by MongoDB.
-----	--------	--------------	--

Table Name: ImageStorage

This table contains all the information about imagesStorage

Table Name:

Task

This table contains all the information about Task.

Field Name	Data type	Constraint	Description
_id	Object	PK	Indicates Inventory ID.
hotel	Object	Required	Stores the reference to the hotel.
title	Varchar	Required	Stores the name of the inventory item.
description	Varchar	Required	Indicates the category of the inventory item.
status	Varchar	Required	Stores the quantity available in stock.
priority	Varchar	Required	Stores the measurement unit (e.g., kg, liter).
dueDate	Object	Required	Stores the price of the inventory item.
assignedTo	Varchar	Required	Stores the reference to the supplier.
category	Varchar	Optional	Stores the date when the item was purchased.
progress	Number	Default:0	Stores the minimum stock level before alert.
notes	Varchar	Optional	Indicates if the stock is below the minimum level.
attachments	Object	Optional	Indicates if the inventory item is out of stock.
createdAt	Object	Required	Stores the timestamp when the inventory item was created.

updatedAt	Object	Required	Stores the timestamp when the inventory item was last updated.
__v	Number	System Field	Stores the versioning field managed by MongoDB.

Table Name: Hotel

This table contains all the information about Hotel like address, phone, logo, rating.

Field Name	Data type	Constraint	Description
_id	Object	Pk	Indicates Hotel ID.
name	Object	Required	Stores the name of the hotel.
email	Varchar	Required	Stores the hotel's email address.
password	Varchar	Required	Stores the hashed password for authentication.
address	Varchar	Required	Stores the hotel's address details.
phone	Varchar	Required	Stores the hotel's contact phone number.
description	Object	Required	Stores a brief description of the hotel.
logo	Varchar	Required	Stores the URL of the hotel's logo image.
coverimage	Varchar	Optional	Stores the URL of the hotel's cover image.
isVerified	Number	Default:0	Indicates whether the hotel is verified.

rating	Varchar	Optional	Stores the average rating of the hotel.
numReviews	Object	Optional	Stores the total number of reviews.

verificationDoc	Number	Required	Stores references to verification documents.
reviews	Object	Default:false	Stores customer reviews for the hotel.
updatedAt	Object	Required	Stores the timestamp when the hotel was last updated.
createdAt	Object	Required	Stores the timestamp when the hotel was created.
__v	Number	System Field	Stores the versioning field managed by MongoDB.
isHotel	Boolean	Default:true	Indicates whether the entity is a hotel.
role	Varchar	Required	Indicates the role of the hotel in the system.

Table Name: FeedBack

This table contains all the information about FeedBack like user, rating, comment.

Field Name	Data type	Constraint	Description
_id	Object	PK	Indicates Feedback ID
user	Object	Required	Stores the reference to the user who provided the feedback.
rating	Number	Required	Stores the rating given by the user (e.g., 1-5).
comment	Varchar	Optional	Stores the user's feedback comment.
createdAt	Object	Required	Stores the timestamp when the feedback was submitted.
status	Varchar	Default:Pending	Indicates the feedback status (e.g., pending, reviewed).
response	Object	Optional	Stores the response to the feedback from the hotel or admin.
hotel	Object	Required	Stores the reference to the hotel associated with the feedback.

Table Name: Orders

This table contains all the information about Orders.

Field Name	Data type	Constraint	Description
_id	Object	PK	Indicates Order ID.
buyer	Object	Required	Stores the reference to the user who placed the order.
items	Varchar	Required	Stores the list of ordered items.
shippingAddress	Varchar	Required	Stores the shipping address details.
totalAmount	Varchar	Required	Stores the total order amount.
status	Varchar	Default:Pending	Indicates the current order status (e.g., pending, shipped).
paymentMethod	Object	Required	Stores the payment method (e.g., cash, credit card).
paymentStatus	Varchar	Default:unpaid	Indicates the payment status (e.g., unpaid, paid, refunded)
createdAt	Object	Required	Stores the timestamp when the order was created.
updatedAt	Object	Required	Stores the timestamp when the order was last updated.
seller	Object	Required	Stores the reference to the seller processing the order.

Screen Layout

† Admin

Dashboard:

The dashboard features a top navigation bar with a search bar and user icons. Below is a summary section with counts for Verified Sellers (5), Verified Hotels (5), Urgent Listings (3), and Free Food Listings (3). The main area includes sections for Verification Requests, Admin Activity, Urgent Sales Listings, and Free Food Listings.

- Verification Requests:** Pending approvals and notifications. A message says "Need to approve the verification request for Green Farms vegetable seller".
- Admin Activity:** Pending tasks for today. A timeline shows "New Seller Verification" at 09:00 and "Hotel Badge Approval" at 11:00.
- Urgent Sales Listings:** A table with three items, all marked as Active.
- Free Food Listings:** A table with three items, all marked as Active.

Dashboard:

The dashboard includes a sidebar with navigation icons. The main area displays a list of verified users (Sellers and Hotels) and their status. It also shows Hotel Partners (187) and End Users (2,014) with a progress bar.

Quick Actions:

- Verify Seller
- Verify Hotel
- Urgent Sales
- Free Food
- User Reports
- Sales Analytics

Hotel page:

The screenshot shows the FreshConnect User Management interface. At the top, there's a search bar and navigation icons. Below it, a summary section displays user counts: 46 Total Users, 26 Verified, and 20 Pending. A large table lists individual users with columns for Name, Email, Type, Role, Status, Last Login, Joined, Verification, and Actions. Each user entry includes a small profile icon and a three-dot menu.

User	Email	Type	Role	Status	Last Login	Joined	Verification	Actions
AU Admin User	admin@example.com	Admin	Admin	Active	2025-03-21T20:31:28.807Z	2025-03-21T20:31:28.813Z	Verified	⋮
A Aman	amanpatef4108@gmail.com	Hotel	Customer	Suspended	Never	2025-03-22T03:25:15.914Z	Verified	⋮
AK Amit Kumar	amit.kumar@example.com	Seller	Seller	Active	Never	2023-12-05T00:00:00.000Z	Unverified	⋮
AD Amita Desai	amita@organicveggies.com	Vegetable Seller	Seller	Active	2023-05-17 16:20	2022-08-15	Verified	⋮
AR Anil Reddy	anil.reddy@example.com	Seller	Seller	Active	Never	2023-12-25T00:00:00.000Z	Unverified	⋮
AP Anjali Patel	anjali@greatfoods.com	Hotel	Manager	Active	2023-05-16 11:45	2022-09-20	Unverified	⋮
AM Arjun Mehta	arjun@freshfarms.in	Vegetable Seller	Seller	Active	2023-02-28 10:30	2022-06-18	Unverified	⋮
CM Content Moderator	moderator@example.com	Admin	Moderator	Active	2025-03-20T20:31:28.846Z	2024-11-21T20:31:28.846Z	Verified	⋮

Add user:

The screenshot shows the 'Add New User' form. It includes fields for 'Full Name *', 'Email Address *', 'User Type' (set to 'Customer'), 'Role' (set to 'End Customer'), 'Status' (set to 'Active'), a toggle for 'Verified Account' which is off, and fields for 'Password *' and 'Confirm Password *'. At the bottom are 'Cancel' and 'Add User' buttons.

Vegetable seller verification page:

The screenshot shows the 'Vegetable Seller Verification' page. It displays statistics: Total Vegetable Sellers (72), Pending Verification (24), Verified Sellers (24), and Organic Farms (24). Below these are filters for search, status, farming type, certification, and inspection. A navigation bar at the top includes 'Verification Checklist' and 'Export List'. The main table lists sellers with columns for Details, Category, Submission Date, Document Status, Verification Status, and Actions (Verify, Products). The table shows three entries: Neha Gupta, Meera Singh, and Anil Reddy.

Vegetable Seller Verification					
Verify sellers to ensure quality vegetables for hotel partners					
Total Vegetable Sellers 72		Pending Verification 24		Verified Sellers 24	
All applications	Status All Status	Farming Type All Types	Certification All Certifications	Inspection All Statuses	
Search by name, owner, location					Clear Refresh
ALL SELLERS	PENDING 24	APPROVED 24	REJECTED 24	PREMIUM SELLERS 15	
Seller Details	Category	Submission Date	Document Status	Verification Status	Actions
Neha Gupta Neha Gupta neha.gupta@example.com	seller	2024-01-15T00:00:00.000Z	<input checked="" type="checkbox"/> Business Certificate <input checked="" type="checkbox"/> Tax Identification <input checked="" type="checkbox"/> Identity Proof <input checked="" type="checkbox"/> Address Proof <input checked="" type="checkbox"/> Organic Certification <input checked="" type="checkbox"/> Pesticide Usage Declaration	Active	Verify Products
Meera Singh Meera Singh meera.singh@example.com	seller	2024-01-05T00:00:00.000Z	<input checked="" type="checkbox"/> Business Certificate <input checked="" type="checkbox"/> Tax Identification <input checked="" type="checkbox"/> Identity Proof <input checked="" type="checkbox"/> Address Proof <input checked="" type="checkbox"/> Organic Certification <input checked="" type="checkbox"/> Pesticide Usage Declaration	Active	Verify Products
Anil Reddy localhost:3000/seller-verification	seller	2023-12-25T00:00:00.000Z	<input checked="" type="checkbox"/> Business Certificate <input checked="" type="checkbox"/> Tax Identification <input checked="" type="checkbox"/> Identity Proof <input checked="" type="checkbox"/> Address Proof	Active	Verify Products

Hotel verification:

The screenshot shows the FreshConnect Hotel Verification interface. At the top, there are six summary cards: Total Hotels (24), Pending Verification (15), Approved (7), Rejected (2), Verified Suppliers (7), and Food Waste Programs (9). Below these are search and filter options. A main table lists four hotels: aman@123, NA, aman@123, and Grand Plaza Hotel, each with details like location, owner, vegetable program status, and actions.

Hotel Details	Location	Owner	Vegetable Program	Status	Actions
aman@123 0 images • 0 facilities	📍	abc@gmail.com	0/0 organic items 0 surplus listings	Pending	👁️ ✅ 🚫 ⏮
NA 0 images • 0 facilities	📍	aman@gmail.com	0/0 organic items 0 surplus listings	Pending	👁️ ✅ 🚫 ⏮
aman@123 0 images • 0 facilities	📍	Aman aman@gmail.com	0/0 organic items 0 surplus listings	Pending	👁️ ✅ 🚫 ⏮
Grand Plaza Hotel 0 images • 0 facilities	📍	John Peterson manager@grandplaza.com	0/0 organic items 0 surplus listings	Pending	👁️ ✅ 🚫 ⏮

Order monitoring:

The screenshot shows the FreshConnect Order Monitoring interface. At the top, there are four summary cards: Total Orders (20), Hotel Orders (5), Total Revenue (₹7,282.80), and Sustainability Impact (11). Below these are filter options and a main table listing five recent orders with details like customer name, date, status, amount, payment, and actions.

Order ID	Customer	Date	Status	Amount	Payment	Actions
ORD-2310010 0 items	Arjun Mehta arjun.mehtha@gmail.com	9/8/2023	Processing	₹87.98	completed	🔗 ⏮
ORD-2310005 0 items	Raj Malhotra raj.malhotra@gmail.com	9/5/2023	shipped	₹183.99	completed	🔗 ⏮
ORD-2310009 0 items	Meera Joshi meera.joshi@gmail.com	9/15/2023	pending	₹81.49	pending	🔗 ⏮
ORD-2310007 0 items	Anand Iyer anand.iyer@gmail.com	9/12/2023	Delivered	₹93.99	completed	🔗 ⏮
ORD-2310003 0 items	Vikram Singh vikram.singh@gmail.com	9/10/2023	processing	₹1,444.99	completed	🔗 ⏮

Payment management:

The screenshot shows the FreshConnect Payment Management dashboard. At the top, there are four summary boxes: Total Transactions (15), Total Revenue (₹10,100.20), Platform Commission (₹NaN), and Food Waste Reduced (0 transactions). Below these are search and filter options. The main area displays a table of transactions with columns for Transaction ID, Date, Buyer, Seller, Type, Amount, Status, and Action. The table lists several completed transactions, including ones from Anil Sharma, Radisson Blue, Marriott International, Grand Hyatt, Karthik Rajan, Arjun Nair, Meera Joshi, and Priya Singh.

ALL TRANSACTIONS	HOTEL PURCHASES	END USER PURCHASES	URGENT & WASTE REDUCTION	REFUNDS			
Transaction ID	Date ↓	Buyer	Seller	Type	Amount	Status	Action
67dc5ad8d4e288ba99d69dfa		Anil Sharma	Unknown	Payment	₹75.25	completed	⋮
67dc5ad8d4e288ba99d69dfd		Radisson Blue	Unknown	Payment	₹499.50	completed	⋮
67dc6ad8d4e288ba99d69dfc		Marriott International	Unknown	Payment	₹1,899.99	completed	⋮
67dc6ad8d4e288ba99d69dfb		Grand Hyatt	Unknown	Payment	₹2,750.00	completed	⋮
67db3fa31dd08348b210c100		Karthik Rajan	Unknown	Payment	₹320.00	pending	⋮
67db3fa31dd08348b210c102		Arjun Nair	Unknown	Payment	₹480.00	failed	⋮
67db3fa31dd08348b210c101		Meera Joshi	Unknown	Payment	₹340.00	completed	⋮
localhost:3000/payments d4e288ba99d69dfa		Priya Singh	Unknown	Payment	₹99.99	completed	⋮

complaint page:

FreshConnect

Complaint Handling

Total Complaints: 10

Pending: 3

In Progress: 4

Resolved: 3

ALL OPEN IN PROGRESS RESOLVED

ID	Date ↓	Customer	Subject	Priority	Status	Assigned To	Actions
COMP-2025-001	7/8/2023	Sanjay Kapoor	Damaged Packaging	medium	pending	Unassigned	🔗
COMP-2025-002	7/2/2023	Neha Gupta	Wrong Item Delivered	high	in-progress	Unassigned	🔗
COMP-2025-003	6/30/2023	Divya Sharma	Refund Not Processed	urgent	in-progress	Unassigned	🔗
COMP-2025-004	6/25/2023	Rahul Mehta	Late Delivery	medium	resolved	Unassigned	🔗
COMP-2025-005	6/20/2023	Mohan Reddy	Quality Issue	high	resolved	Unassigned	🔗
COMP-2025-006	6/15/2023	John Doe	Delayed Delivery	high	pending	Unassigned	🔗
COMP-2025-007	6/13/2023	Michael Brown	Account Access Issue	low	in-progress	Unassigned	🔗
COMP-2025-008	6/12/2023	Emma Wilson	Refund Not Processed	urgent	pending	Unassigned	🔗
COMP-2025-009	6/10/2023	Alice Smith	Damaged Product	medium	in-progress	Unassigned	🔗

localhost:3000/complaints

Content management page:

The screenshot shows the FreshConnect Content Management interface. At the top, there's a navigation bar with a search bar and user icons. Below it, a sidebar on the left contains various icons for navigation. The main area is titled "Content Management" and has a subtitle "Create and manage your website content". There are three tabs at the top: "PAGES" (selected), "BLOG POSTS", and "MEDIA". Below these are filters for "Status" (set to "All") and "Sort By" (set to "Last Updated"). A search bar and an "Add Page" button are also present. The main content area is a table listing six pages:

Title	Slug	Status	Last Updated	Author	Actions
FAQ	faq	Draft	2023-05-18	Sarah Wilson	⋮
Home Page	home	Published	2023-05-15	Mike Johnson	⋮
About Us	about-us	Published	2023-04-20	Sarah Wilson	⋮
Services	services	Published	2023-04-10	Mike Johnson	⋮
Contact	contact	Published	2023-03-22	John Smith	⋮
Terms of Service	terms	Published	2023-02-15	Mike Johnson	⋮

Market analytics page:

The screenshot shows the FreshConnect Marketplace Analytics interface. At the top, there's a navigation bar with a search bar and user icons. Below it, a sidebar on the left contains various icons for navigation. The main area is titled "Marketplace Analytics" and has a subtitle "Track sales, orders, and marketplace performance". There are three main sections: "Total Sales" (₹0, -0% vs yesterday), "Marketplace Activity" (0 Orders Today, ₹0 Avg. Order, 0% Cancellation), and "Buyer Types" (Hotels & Restaurants 55%, End Consumers 23%, Food Processors 12%, Retailers 8%, Charity & Others 2%). Below these are sections for "Sales Overview" (with tabs for SALES REVENUE, NUMBER OF ORDERS, QUANTITY SOLD (KG)) and a placeholder for a line chart visualization.

System setting Profile:

 FreshConnect

System Settings

GENERAL SECURITY SELLER SETTINGS HOTEL SETTINGS EMAIL NOTIFICATIONS STORAGE & BACKUP DELIVERY

Site Information

Site Name: FreshConnect Admin

Site URL: https://admin.freshconnect.com

Admin Email: admin@freshconnect.com

Maintenance Mode: When enabled, the site will be inaccessible to regular users.

Display Settings

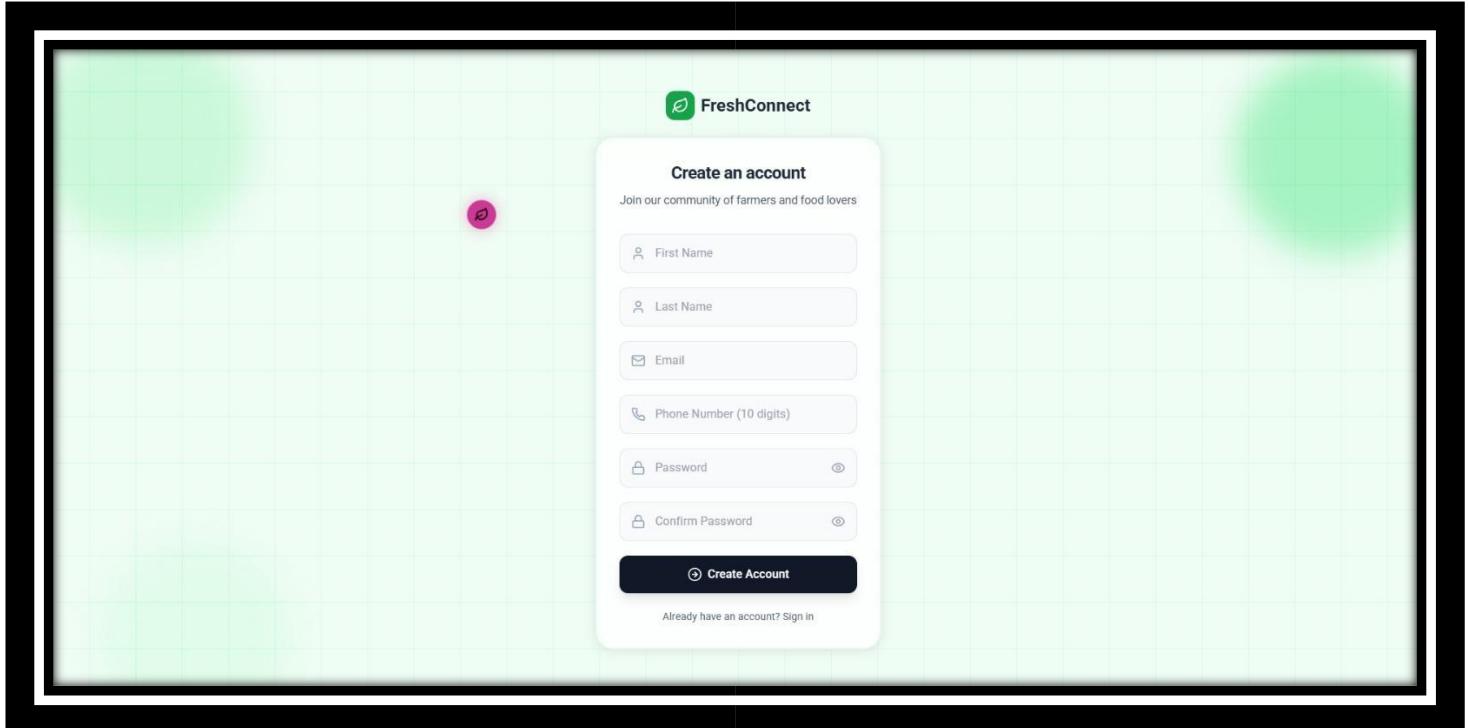
Items Per Page: 20 Date Format: MM/DD/YYYY

Time Format: 12 Hour (AM/PM) Timezone: UTC

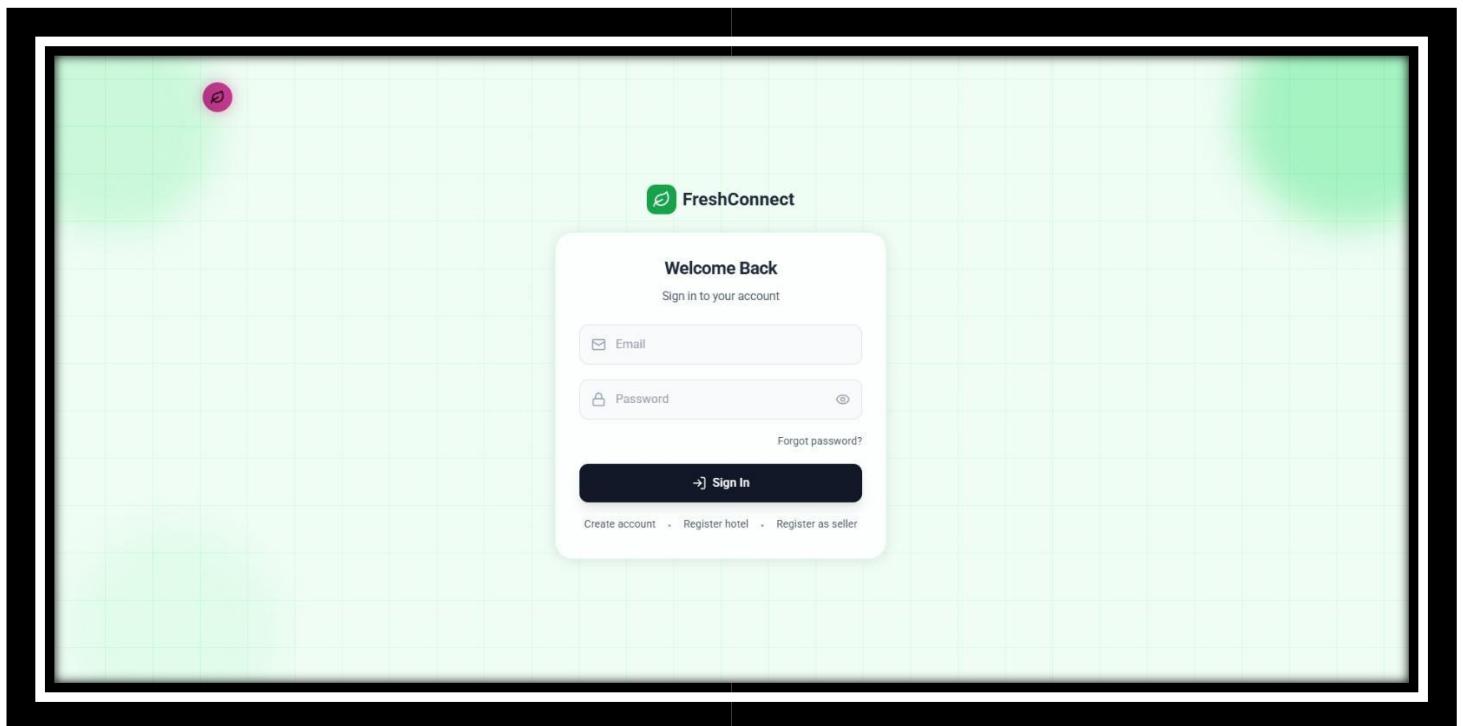
Refresh Save Changes

† Seller Management

Sing up Page:



Login Page:



Dashboard:

The Seller Dashboard provides a quick overview of performance. Key metrics include:

- Total Revenue: ₹6,339.78 (↑ +10.0% from last week)
- Total Orders: 17 (↓ -0% from last week)
- Monthly Sales: ₹1,584.945 (↑ +10.0% from last week)
- Pending Orders: 0 (↓ -0% from last week)

Quick Actions:

- Add Product
- View Orders
- Create Urgent Sale
- Manage Inventory

Weekly Sales chart (This Week vs Last Week):

Day	This Week (₹)	Last Week (₹)
Mon	~20	~20
Tue	~4500	~20
Wed	~20	~20
Thu	~20	~20
Sat	~20	~20
Sun	~1500	~20

Top Selling Items:

- Premium T-Shirt (35 sales, ↑ +8% from last week)
- Designer Jeans (28 sales, ↑ +1% from last week)
- Running Shoes (25 sales, ↑ +5% from last week)

Analytics page:

The Analytics Dashboard provides a deeper look at the business. Key metrics include:

- Total Revenue: ₹1,583.91 (Based on recent orders, ↑ +0% from last period)
- Total Orders: 17 (Complete orders, ↑ +8.3% from last period)
- Avg. Order Value: ₹93.17 (Per order average, ↑ +3.2% from last period)
- Active Products: 3 (1 low stock, ↓ -11.1% from last period)

Navigation tabs: Sales Overview, Product Performance, Customer Analytics, Geographic Analysis.

Revenue Trends chart (Line and Bar):

The chart displays Revenue (Blue Line) and Profit (Green Line) over 30 days. The revenue starts high on Day 1 (~₹4800), drops sharply on Day 2 (~₹200), and then fluctuates between ₹200 and ₹1500 for the rest of the period. Profit follows a similar pattern, starting high and dropping on Day 2.

Product page :

Seller Dashboard Rahul

Products

Manage your product inventory, prices, and details

Total Products: 6 Active in your inventory

Low Stock: 1 Products with less than 5 units

Out of Stock: 1 Products to restock soon

+ Add New Product

Product Categories Overview

Search products... Category: All Categories Stock Status: All Status Sort By: Name (A-Z)

Select all products

Low Stock

Image	Product Name	Status
	Cartoon Character Headphones	Low Stock

Orders page:

Seller Dashboard Rahul

Orders

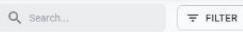
Debug - Orders: 17 | Pending: 0 | Processing: 3 | Delivered: 9

Total Orders: 17 Pending Orders: 0 Processing: 3 Delivered: 9

Search orders... Status: All Status Filters

Order ID	Customer	Date	Total	Status	Action
67ee54b2b05a3433be1670ff	Aman Patel 2.0	4/3/2025	₹54.00	Cancelled	
67da265797b1af1c7087a054	Unknown Customer	4/3/2025	₹89.97	Processing	
67da265797b1af1c7087b003	Unknown Customer	4/3/2025	₹119.98	Processing	
67da265797b1af1c7087a053	Unknown Customer	4/3/2025	₹119.98	Processing	
67ee6476a0f590046754cb21	Unknown Customer	4/1/2025	₹1199.98	Delivered	
67ee6466a0f590046754cb1e	Unknown Customer	4/1/2025	₹1199.98	Delivered	
67da265797b1af1c7087b002	Unknown Customer	4/1/2025	₹89.99	Cancelled	

Sales dashboard page:

Seller Dashboard     

Urgent Sales Dashboard

⚠️ Your products that need to be sold quickly at discounted prices

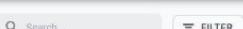
Total Sales 66 No trend data available	Revenue ₹1056 No trend data available	Avg. Discount 66.8% Effective pricing	Expiring Soon 2 Needs attention
---	--	--	--

ALL PRODUCTS  FEATURED  EXPIRING SOON  INACTIVE  ANALYTICS

Search products... Category All Categories Sort by Expiry Date 

+ ADD NEW LISTING

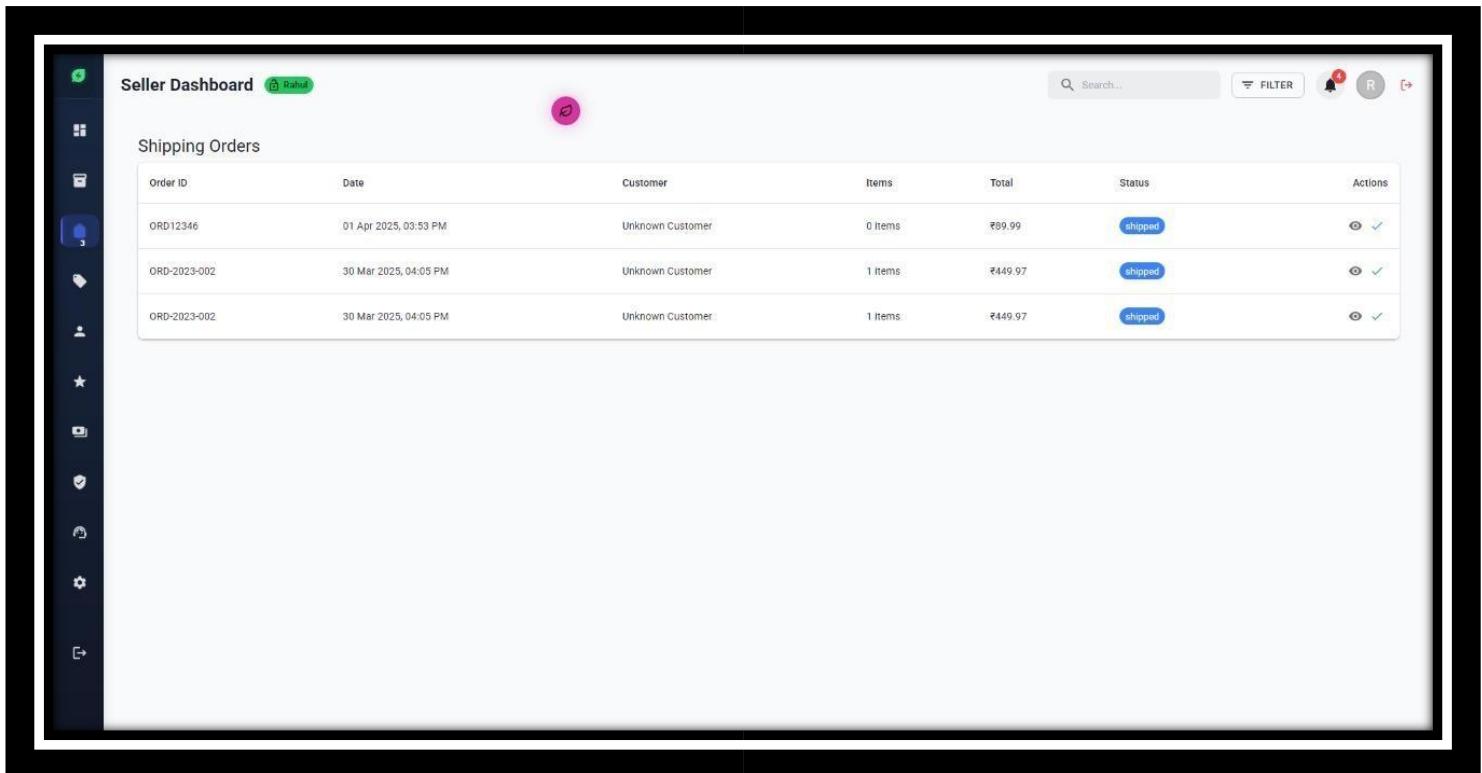


Seller Dashboard     

Shipping Orders

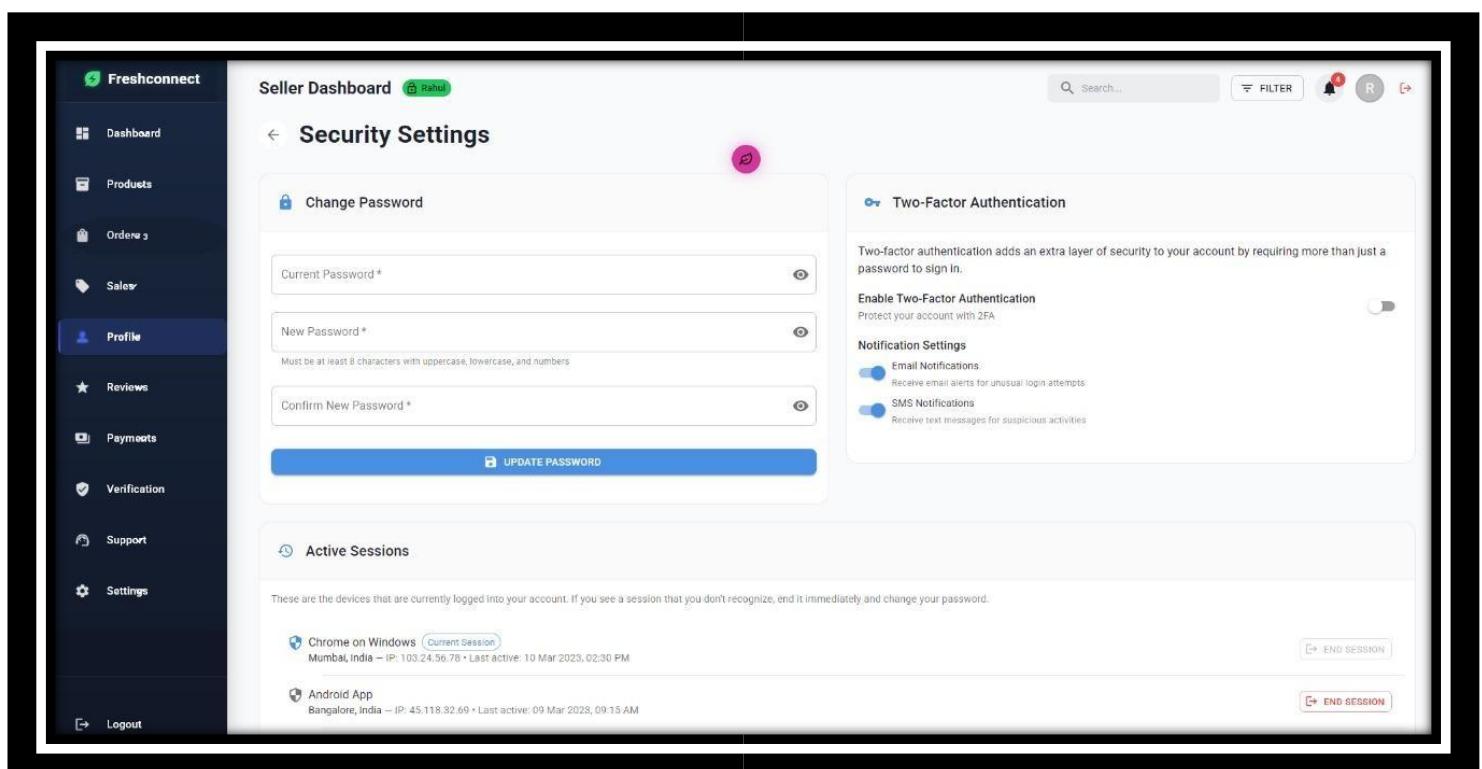
Order ID	Date	Customer	Items	Total	Status	Actions
ORD12346	01 Apr 2025, 03:53 PM	Unknown Customer	0 Items	₹89.99	shipped	 
ORD-2023-002	30 Mar 2025, 04:05 PM	Unknown Customer	1 Items	₹449.97	shipped	 
ORD-2023-002	30 Mar 2025, 04:05 PM	Unknown Customer	1 Items	₹449.97	shipped	 

Profile page:



The screenshot shows the Seller Dashboard interface. On the left is a dark sidebar with various icons for navigation. The main area is titled "Seller Dashboard" and shows a table of "Shipping Orders". The table has columns for Order ID, Date, Customer, Items, Total, Status, and Actions. There are three entries in the table:

Order ID	Date	Customer	Items	Total	Status	Actions	
ORD12346	01 Apr 2025, 03:53 PM	Unknown Customer	0 Items	₹89.99	shipped		
ORD-2023-002	30 Mar 2025, 04:05 PM	Unknown Customer	1 Items	₹449.97	shipped		
ORD-2023-002	30 Mar 2025, 04:05 PM	Unknown Customer	1 Items	₹449.97	shipped		



The screenshot shows the "Security Settings" page within the Seller Dashboard. The left sidebar is identical to the previous dashboard, with the "Profile" option selected. The main content area is titled "Security Settings". It contains two main sections: "Change Password" and "Two-Factor Authentication".

Change Password

This section contains three input fields: "Current Password *", "New Password *", and "Confirm New Password *". Below these fields is a blue "UPDATE PASSWORD" button.

Two-Factor Authentication

This section includes a toggle switch for enabling Two-Factor Authentication and a note about its benefits. It also features "Notification Settings" with two toggle switches: "Email Notifications" and "SMS Notifications", each with a descriptive subtitle.

Active Sessions

This section lists active sessions on the account. It shows a session on "Chrome on Windows" (Current Session) from Mumbai, India, and another on "Android App" from Bangalore, India. Each session entry includes an "END SESSION" button.

Reviews page:

Seller Dashboard (Rahul)

Reviews & Ratings

4.5

Based on 13 reviews

Rating	Count
5★	7
4★	6
3★	0
2★	0
1★	0

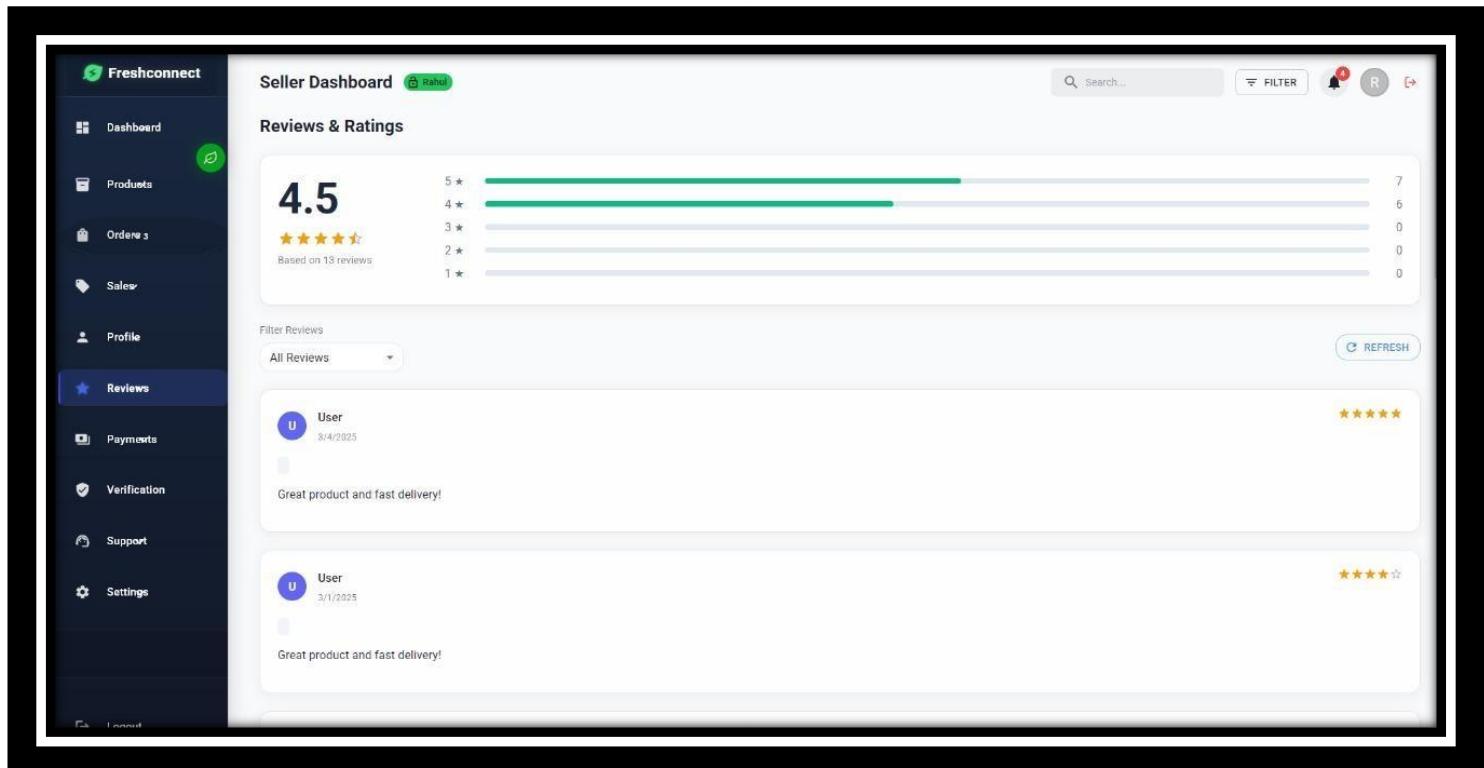
Filter Reviews

All Reviews

REFRESH

User 3/4/2025 ★★★★★
Great product and fast delivery!

User 3/1/2025 ★★★★★
Great product and fast delivery!



Seller Dashboard (Rahul)

Respond to Reviews

Customer Reviews

Respond to customer feedback and build trust with your customers

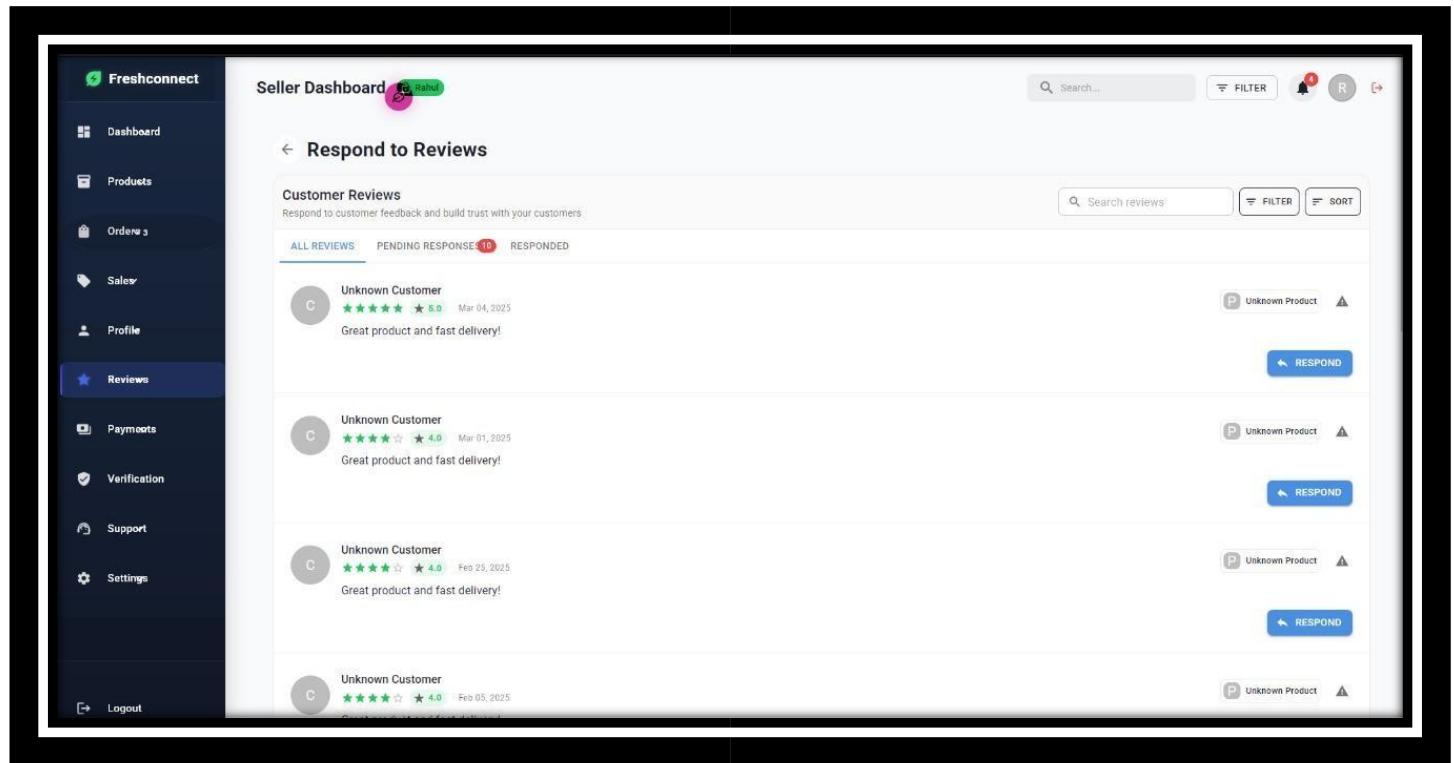
ALL REVIEWS PENDING RESPONSE (10) RESPONDED

Unknown Customer ★★★★★ 5.0 Mar 04, 2025
Great product and fast delivery! RESPOND

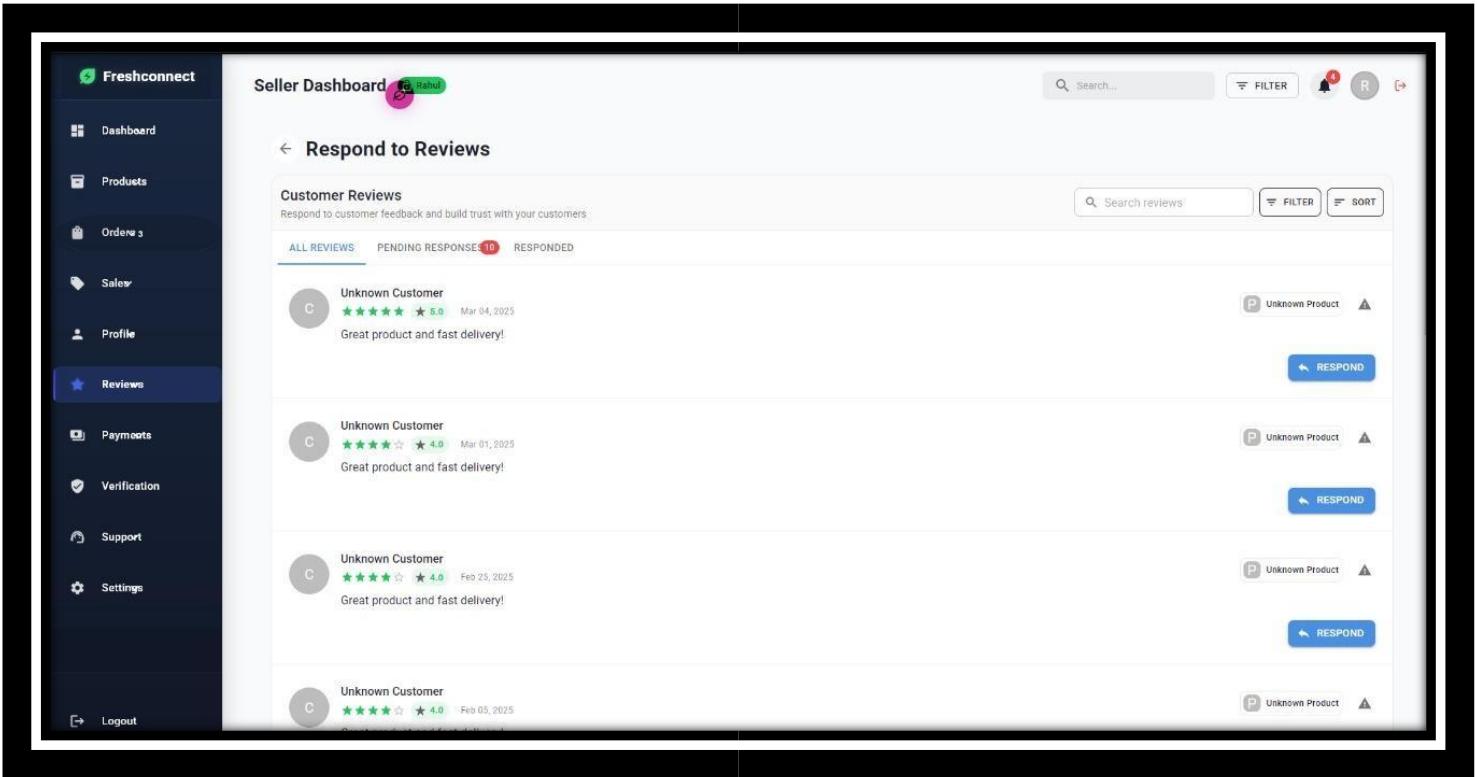
Unknown Customer ★★★★★ 4.0 Mar 01, 2025
Great product and fast delivery! RESPOND

Unknown Customer ★★★★★ 4.0 Feb 25, 2025
Great product and fast delivery! RESPOND

Unknown Customer ★★★★★ 4.0 Feb 05, 2025
Great product and fast delivery! RESPOND

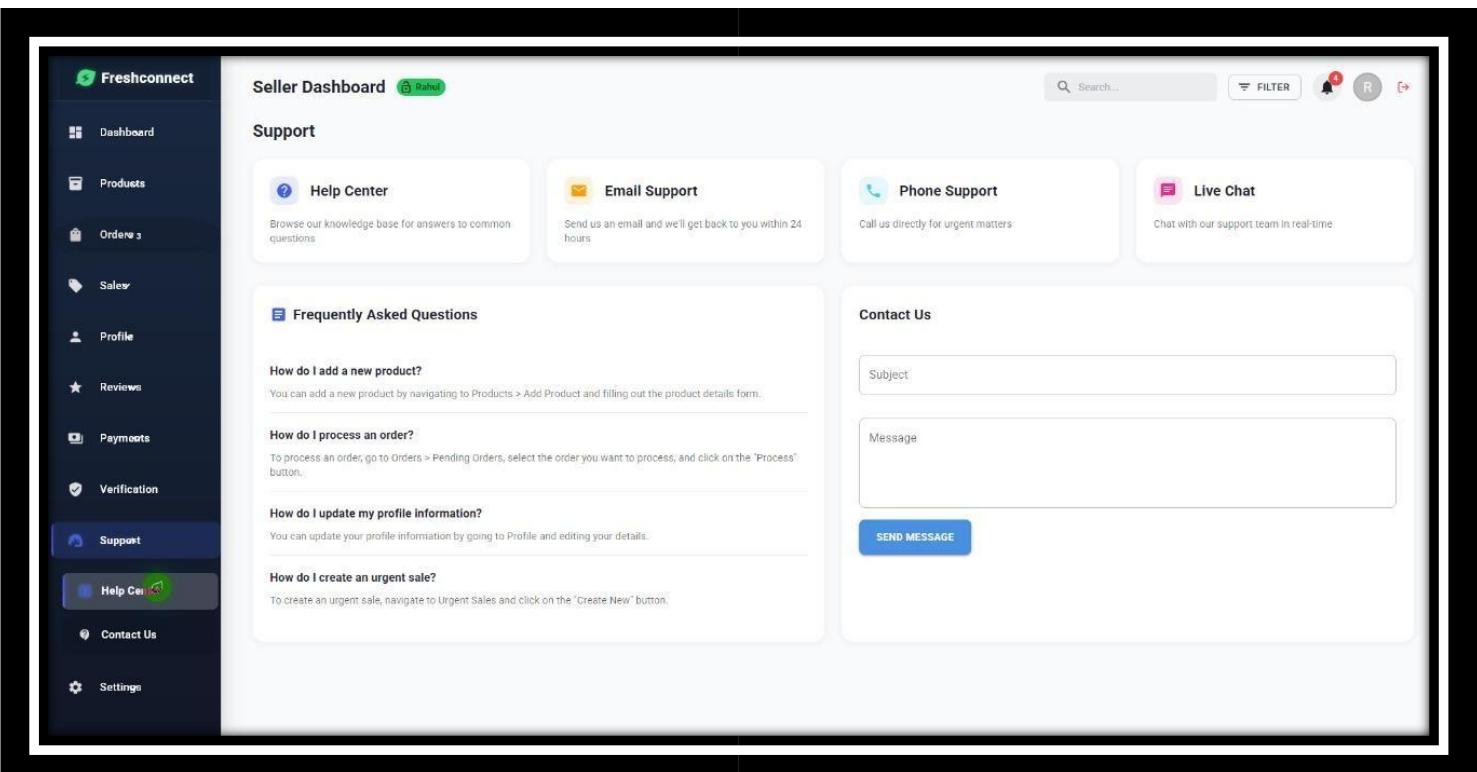


Payment page:

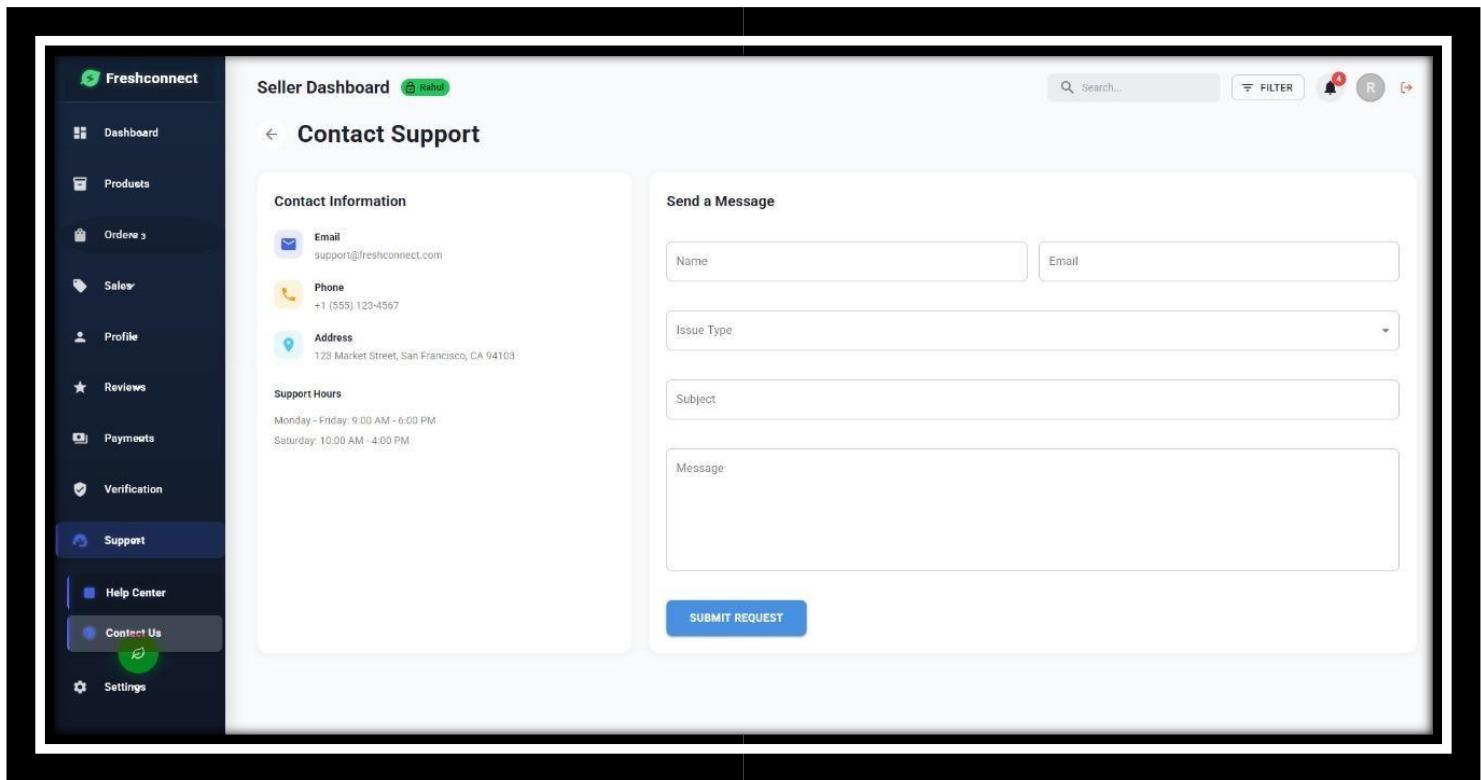


The screenshot shows the 'Seller Dashboard' interface for Freshconnect. On the left, a dark sidebar menu lists various sections: Dashboard, Products, Orders, Sales, Profile, Reviews (which is currently selected), Payments, Verification, Support, and Settings. At the bottom of the sidebar is a 'Logout' button. The main content area is titled 'Seller Dashboard' and 'Seller ID: rahul'. It features a header with a search bar, filter, and notification icons. Below this, a section titled 'Customer Reviews' encourages responding to customer feedback to build trust. It displays four review entries from 'Unknown Customer' with 5.0 star ratings on March 04, 2025. Each entry includes a snippet of the review text: 'Great product and fast delivery!' and a 'RESPOND' button. A navigation arrow '← Respond to Reviews' is located above the reviews section.

Support page:

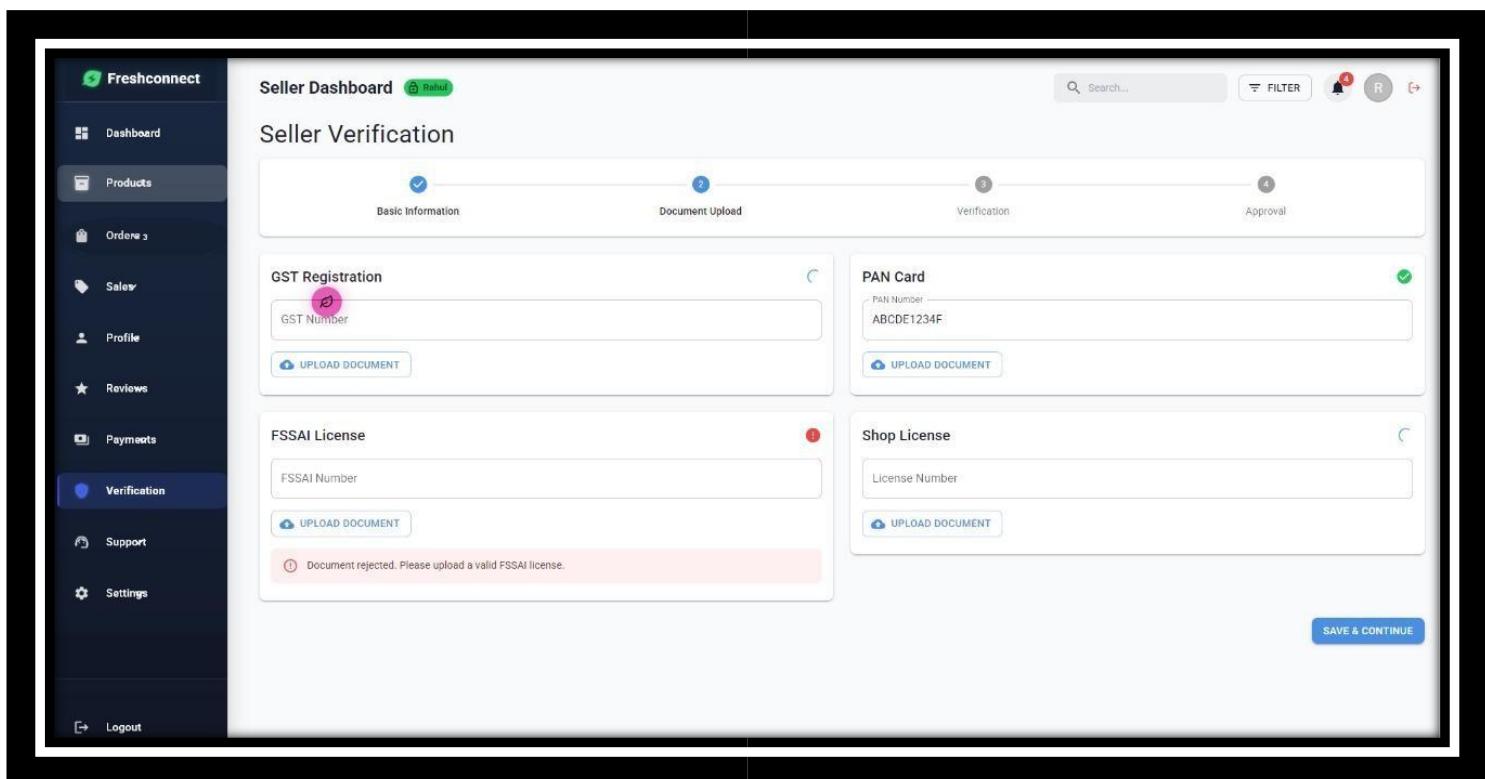


The screenshot shows the 'Seller Dashboard' interface for Freshconnect, specifically the 'Support' section. The left sidebar remains the same as the previous screenshot, with 'Reviews' selected. The main content area is titled 'Seller Dashboard' and 'Seller ID: rahul'. It features a header with a search bar, filter, and notification icons. The 'Support' section is highlighted in blue. It contains four support options: 'Help Center' (Browse our knowledge base for answers to common questions), 'Email Support' (Send us an email and we'll get back to you within 24 hours), 'Phone Support' (Call us directly for urgent matters), and 'Live Chat' (Chat with our support team in real-time). Below these options is a 'Frequently Asked Questions' section with several questions and their answers. For example, 'How do I add a new product?' is answered with instructions to navigate to Products > Add Product. The 'Contact Us' section allows users to enter a subject and message, with a 'SEND MESSAGE' button. A navigation arrow '← Support' is located above the support options.

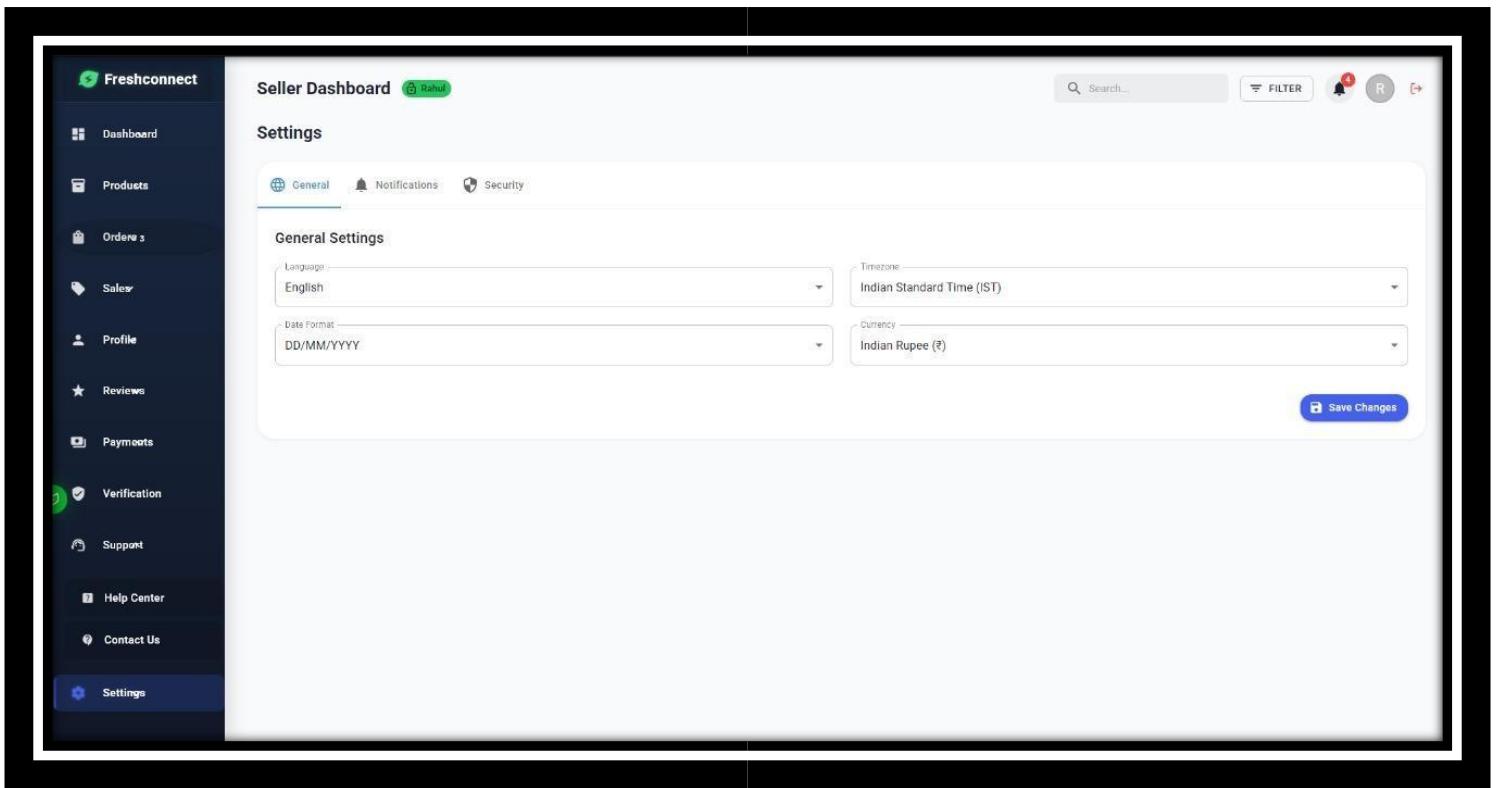


The screenshot shows the Freshconnect Seller Dashboard with the 'Contact Support' section selected. The left sidebar includes links for Dashboard, Products, Orders, Sales, Profile, Reviews, Payments, Verification, Support (highlighted), Help Center, and Contact Us (highlighted). The main content area displays contact information (Email: support@freshconnect.com, Phone: +1 (555) 123-4567, Address: 123 Market Street, San Francisco, CA 94103), support hours (Monday - Friday: 9:00 AM - 6:00 PM; Saturday: 10:00 AM - 4:00 PM), and a 'Send a Message' form. The message form fields include Name, Email, Issue Type (dropdown), Subject, and Message, with a 'SUBMIT REQUEST' button at the bottom.

Seller verification page:



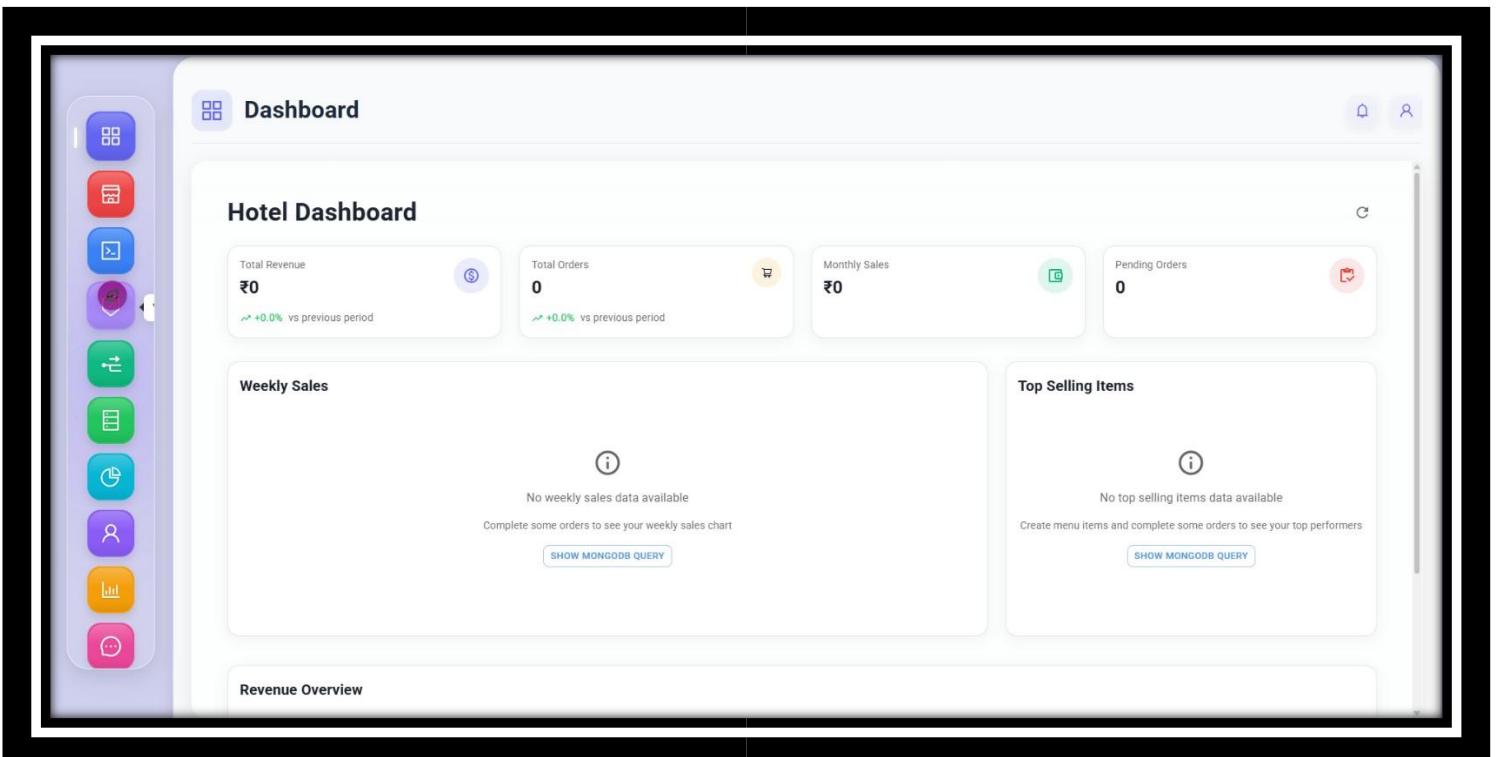
The screenshot shows the Freshconnect Seller Dashboard with the 'Verification' section selected. The left sidebar includes links for Dashboard, Products, Orders, Sales, Profile, Reviews, Payments, Verification (highlighted), Support, and Settings. The main content area displays the 'Seller Verification' process with four steps: Basic Information (checkmark), Document Upload (blue question mark), Verification (grey circle), and Approval (grey circle). The 'Basic Information' step shows a 'GST Registration' section with a placeholder 'GST Number' and a 'UPLOAD DOCUMENT' button. The 'Document Upload' step shows a 'PAN Card' section with a placeholder 'PAN Number' containing 'ABCDE1234F' and a 'UPLOAD DOCUMENT' button. The 'Verification' step shows an 'FSSAI License' section with a placeholder 'FSSAI Number' and a 'UPLOAD DOCUMENT' button, accompanied by a red error message: 'Document rejected. Please upload a valid FSSAI license.' The 'Approval' step is currently not visible. A 'SAVE & CONTINUE' button is located at the bottom right.



The screenshot shows the Seller Dashboard settings page for Freshconnect. The left sidebar includes links for Dashboard, Products, Orders, Sales, Profile, Reviews, Payments, Verification, Support, Help Center, Contact Us, and Settings. The main content area is titled 'Settings' and has tabs for General, Notifications, and Security. Under 'General Settings', there are fields for Language (English), Date Format (DD/MM/YYYY), Timezone (Indian Standard Time (IST)), and Currency (Indian Rupee (₹)). A 'Save Changes' button is at the bottom right.

⊕ Hotel Management :

Dashboard :

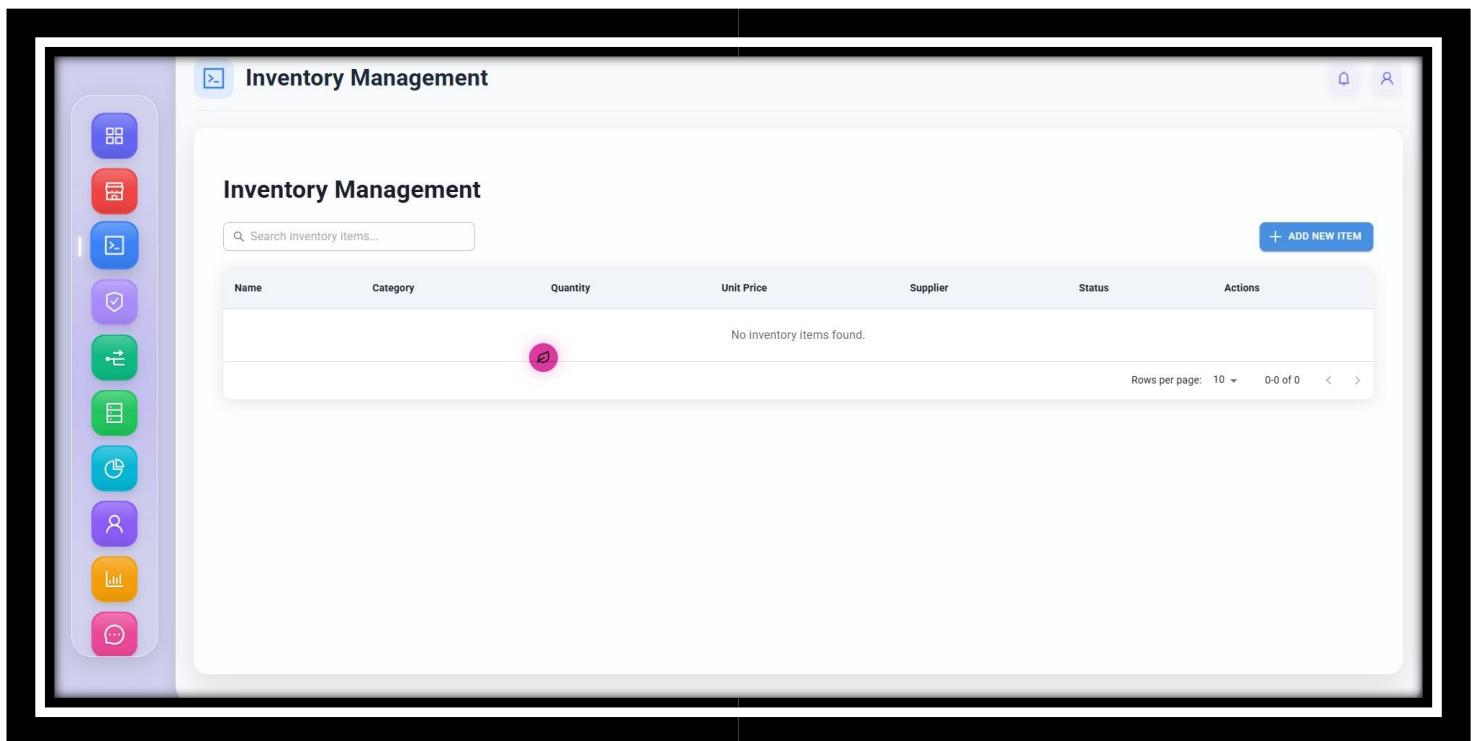


The screenshot shows the Hotel Management Dashboard. The left sidebar features a vertical stack of colorful icons for navigation. The main dashboard is titled 'Hotel Dashboard' and displays four key metrics: Total Revenue (₹0, +0.0% vs previous period), Total Orders (0, +0.0% vs previous period), Monthly Sales (₹0), and Pending Orders (0). Below these are sections for 'Weekly Sales' (No weekly sales data available) and 'Top Selling Items' (No top selling items data available). At the bottom, there's a 'Revenue Overview' section with a 'SHOW MONGODB QUERY' button.

Purchase page :

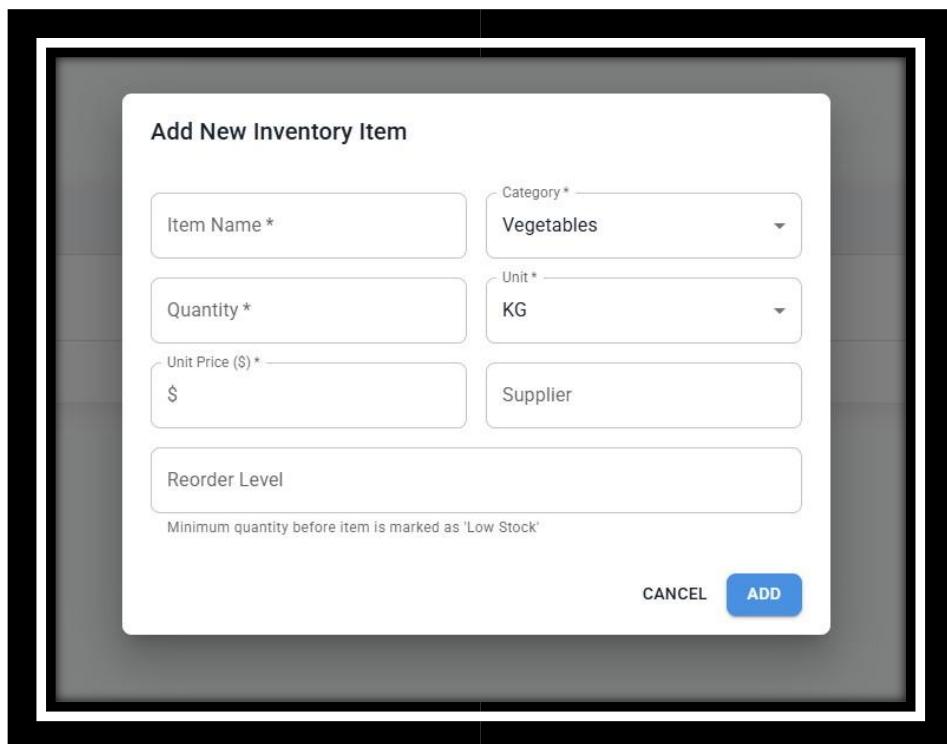
The screenshot shows a software interface titled "Purchase History". On the left, there is a vertical sidebar containing ten colorful icons with corresponding labels: Home (purple), Grid View (blue), Document (red), Map (green), Shield (purple), Refresh (teal), List (green), Clock (blue), User (purple), and Chat (orange). The main area has a header "Purchase History" with a refresh button. Below the header is a table with columns: Date, Supplier, Items, Total Amount, Payment Method, and Status. A message at the bottom states "No purchase history found."

Inventory page :



The screenshot shows a mobile application interface for 'Inventory Management'. On the left is a vertical sidebar with ten icons: a grid, a document, a magnifying glass, a shield, a double arrow, a list, a clock, a person, a bar chart, and a message bubble. The main screen has a header 'Inventory Management' with a back arrow and search icons. Below is a search bar and a blue button '+ ADD NEW ITEM'. A table header with columns 'Name', 'Category', 'Quantity', 'Unit Price', 'Supplier', 'Status', and 'Actions' is followed by a message 'No inventory items found.' and a circular progress bar. At the bottom are pagination controls: 'Rows per page: 10', '0-0 of 0', and navigation arrows.

Add item :



The screenshot shows a modal dialog titled 'Add New Inventory Item'. It contains fields for 'Item Name *' (with a dropdown for 'Category' set to 'Vegetables'), 'Quantity *' (with a dropdown for 'Unit' set to 'KG'), 'Unit Price (\$) *' (with a dropdown for 'Supplier' showing '\$'), and a 'Reorder Level' input field. A note at the bottom states 'Minimum quantity before item is marked as 'Low Stock''. At the bottom right are 'CANCEL' and 'ADD' buttons.

Verified badge page:

The screenshot shows the 'Verified Badge Management' page. On the left is a vertical sidebar with icons for Home, Reports, Analytics, Settings, and Help. The main header is 'Verified Badge Management'. Below it, a section titled 'Verified Badge' says 'Get verified to build trust with customers and access premium features'. A horizontal progress bar shows three steps: 'Apply' (Submit application), 'Upload' (Upload documents), and 'Review' (Under review). To the right of the bar is a 'Verified' button with the text 'Get verified badge'. A message box indicates 'Verification In Progress' with the note 'Your verification request is currently being reviewed by our team. This process typically takes 2-3 business days.' Below this is a 'Submitted Documents' section listing four items: 'Business License' (uploaded on 2023-06-01), 'Food Safety Certificate' (uploaded on 2023-06-01), 'Restaurant Photos' (uploaded on 2023-06-01), and 'Owner ID' (uploaded on 2023-06-01). Each document entry includes a green 'Uploaded' status indicator. At the bottom are 'Contact Support' and 'Cancel Application' buttons.

Leftover food listing page:

The screenshot shows the 'Leftover Food Listing' page. The left sidebar has an icon for Leftover Food. The main header is 'Leftover Food Management'. A blue button at the top right says '+ ADD NEW LISTING'. Below it is a search bar with 'Search listings...' placeholder text, and dropdown menus for 'Category' and 'Status'. A message 'No listings found' is displayed above a central area. This area features a trash bin icon and the text 'No food listings yet' followed by the instruction 'Create your first leftover food listing by clicking the Add New Listing button'. A blue '+ ADD NEW LISTING' button is located here. At the bottom, there are pagination controls: 'Rows per page: 6', '0-0 of 0', and navigation arrows. A large blue '+' button is in the bottom right corner.

Add food listing :

Add New Food Listing

Title *

Description *

Price

Category *

All leftover food is provided for free

Quantity *

Unit *

Expiry Time *

Dietary Info

Images (Required)

No images uploaded yet

 UPLOAD IMAGE

Images smaller than 10MB are supported.

CANCEL CREATE

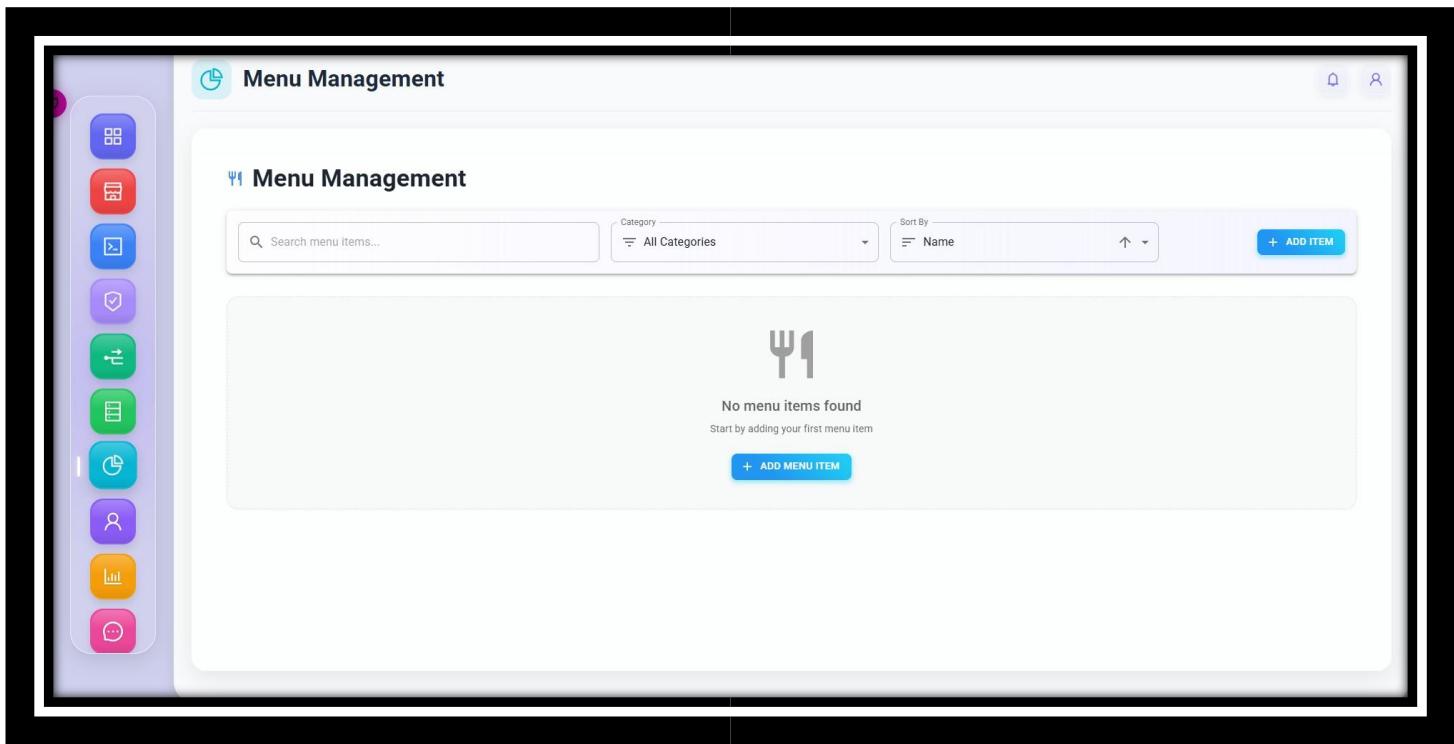
Urgent food sale page :

The screenshot shows the 'Urgent Food Sale' application interface. On the left is a vertical sidebar with a purple header containing a user icon and a list of colorful icons for navigation. The main area has a light gray header with the title 'Urgent Food Sale' and a search bar. Below the header is a section titled 'Urgent Sales Management' with buttons for 'REFRESH', '+ CREATE NEW SALE', 'SHOW FILTERS', and 'SORT'. The main content area is titled 'My Urgent Sales' and displays two items: 'Fresh Vegetables Mix' and 'Dessert Special'. Each item card includes a small image, a discount percentage (e.g., 47% OFF, 41% OFF), a brief description, original price, sale price, stock information, and an expiration date. Below each card are 'EDIT' and 'DELETE' buttons.

Add food sale item :

The screenshot shows the 'Create New Urgent Sale' form. The top bar is blue with the title 'Create New Urgent Sale'. The form contains several input fields: 'Name *' (text input), 'Description *' (text area), 'Original Price *' (text input with ₹ symbol) and 'Discounted Price *' (text input with ₹ symbol), 'Stock *' (text input), 'Unit' (dropdown menu set to kg), 'Category' (dropdown menu set to Vegetables), and 'Expiry Date & Time *' (text input showing 04/08/2025 15:10 with a calendar icon). To the right is a 'Product Image *' section with a placeholder 'No image selected' and a 'SELECT IMAGE' button. At the bottom are 'CANCEL' and 'CREATE' buttons. A warning message at the bottom states '⚠ Product will expire in less than 24 hours'.

Menu management :



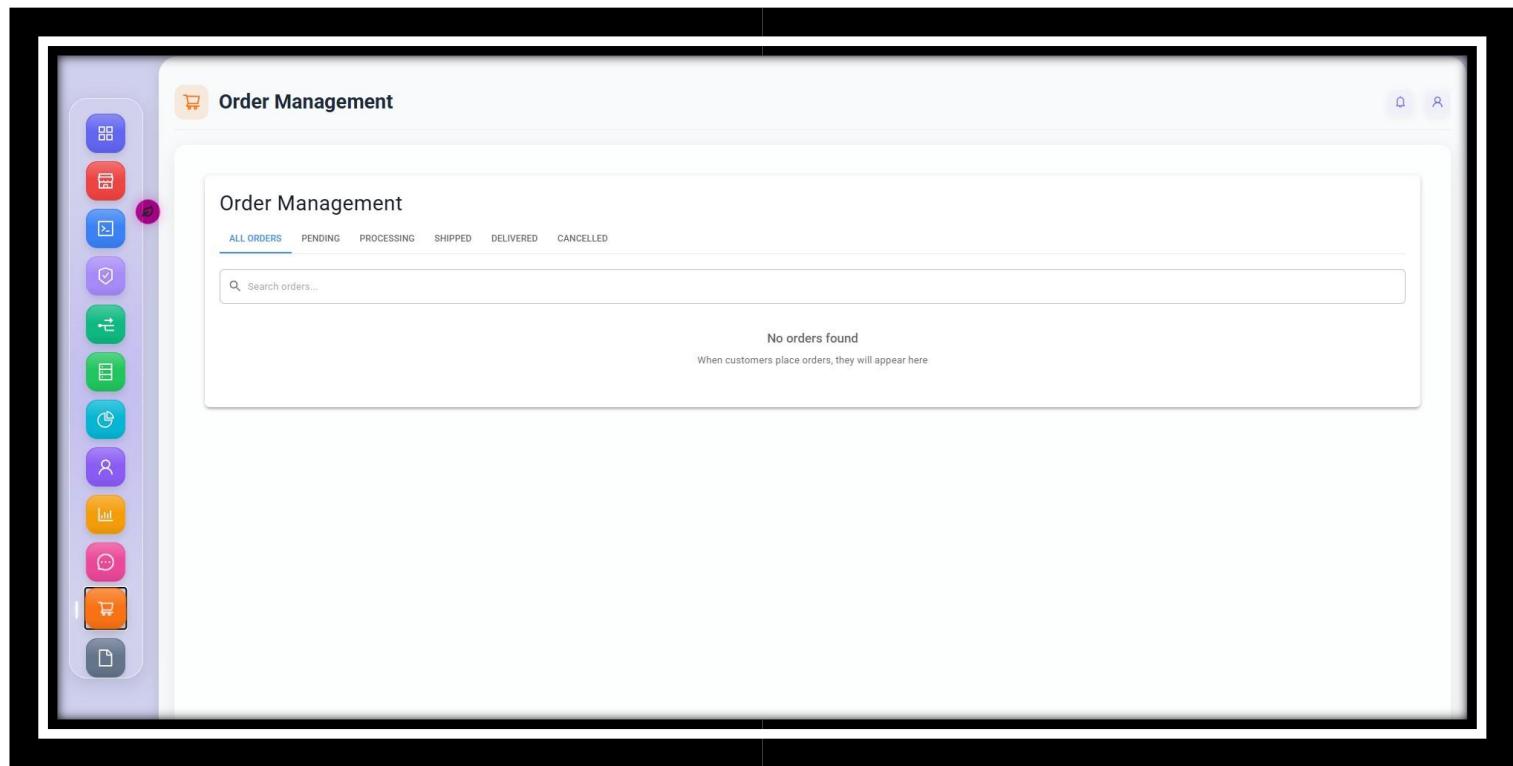
Profile page :

The screenshot shows the 'Profile Settings' page for a business profile. On the left, there's a vertical sidebar with various icons for different features like dashboard, reports, security, and users. The main area has a header 'Profile Settings' with a user icon. Below it is a circular placeholder for a profile picture with a camera icon. The business name 'Taj Hotel & Restaurant' is displayed with a green 'Verified Hotel' badge. It also says 'Member since 1995'. There are tabs for 'PROFILE', 'SECURITY', and 'NOTIFICATIONS'. A blue 'EDIT PROFILE' button is at the top right. The profile form includes fields for 'Hotel Name' (Taj Hotel & Restaurant), 'Owner Name' (Rajan Patel), 'Email' (info@tajhotel.com), 'Phone' (+91 9876543210), 'Address' (123 Main Street, Mumbai, Maharashtra 400001), 'Website' (www.tajhotel.com), and 'Founded Year' (1995). A 'Description' field contains the text: 'A luxury hotel and restaurant offering authentic Indian cuisine with a modern twist. We specialize in both vegetarian and non-vegetarian dishes prepared by our expert chefs.'

Feed back page :

The screenshot shows the 'Customer Feedback' page. The left sidebar has the same set of icons as the previous page. The main header is 'Customer Feedback' with a three-dot icon. Below it is a search bar with the placeholder 'Search feedbacks...'. A message 'No feedbacks found' is centered, with a smaller note below it stating 'When customers leave feedback, they will appear here'. At the bottom, there's a large empty white space.

Order management page:



+ User

Home Page

Welcome to FreshConnect

Connecting Restaurants with Consumers

Find restaurants, order urgent sales, or access free food listings to reduce waste and save money.

Join Our Platform

Whether you're a restaurant looking to reduce food waste or a consumer seeking quality food, our platform connects you with what you need.

[Register Now →](#)[Join as Restaurant →](#)**Hotel Sunshine Surplus**

Hotel Sunshine

4.9 • 1.2 km

Why Join FreshConnect?

 Verified Restaurants
150+ verified restaurants Fast Delivery
45 min average delivery time Community Support
Reducing waste, saving money Verified Quality
100% Chemical Free

 FreshConnect

Market Restaurants Urgent Sales Free Food Login Register

1000+
Verified Farmers

200+
Partner Restaurants

5000+
Happy Customers

10000+
Meals Shared

Verified Restaurants

Discover restaurants that prioritize quality and use verified fresh ingredients



Green Farm Restaurant
Specializing in farm-to-table cuisine and organic ingredients
★ 4.8 (230+ reviews) 100% Fresh Ingredients



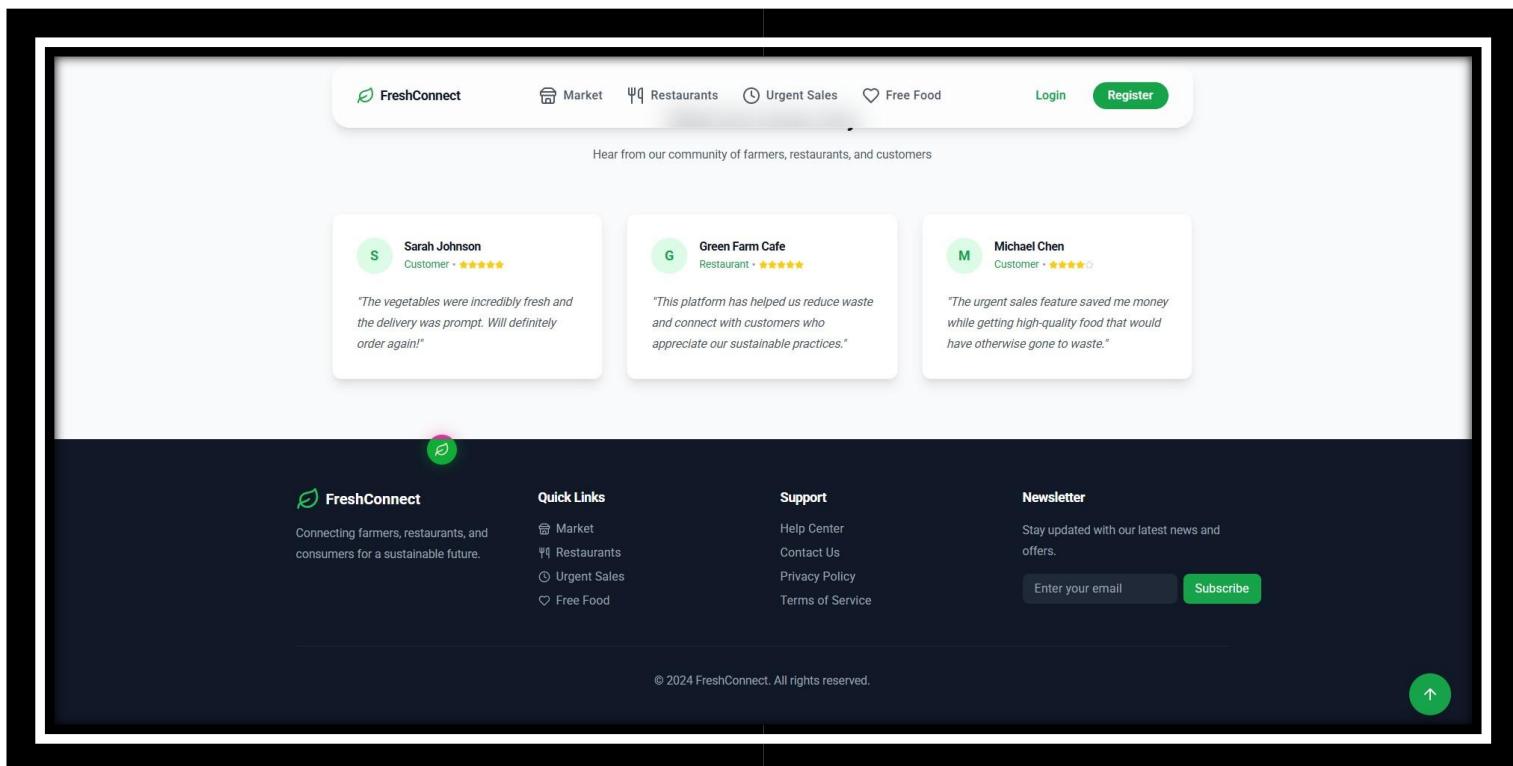
Spice Garden
Authentic Indian cuisine with fresh ingredients
★ 4.6 (185+ reviews) 100% Fresh Ingredients



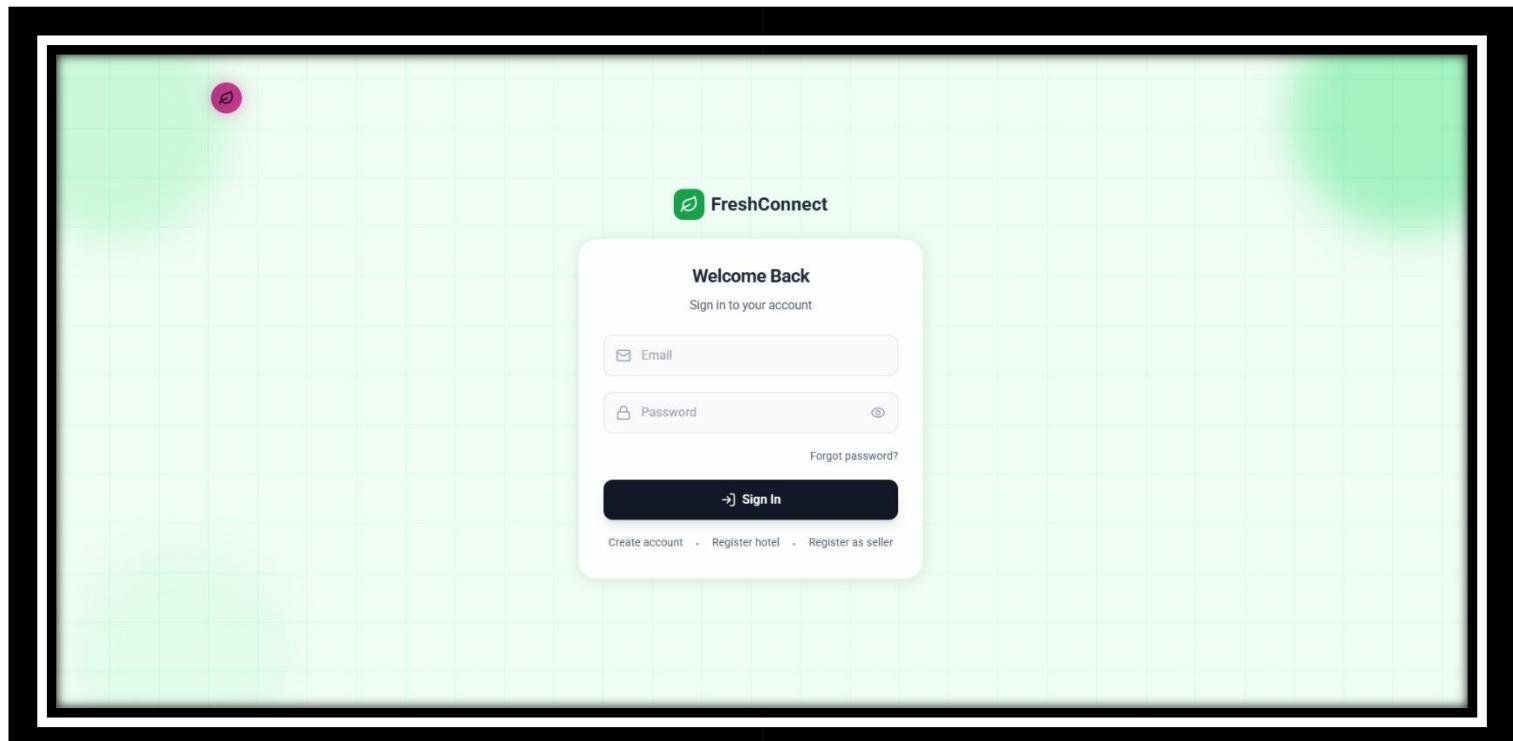
Ocean Breeze
Fresh seafood restaurant with daily specials
★ 4.9 (310+ reviews) 100% Fresh Ingredients

↑

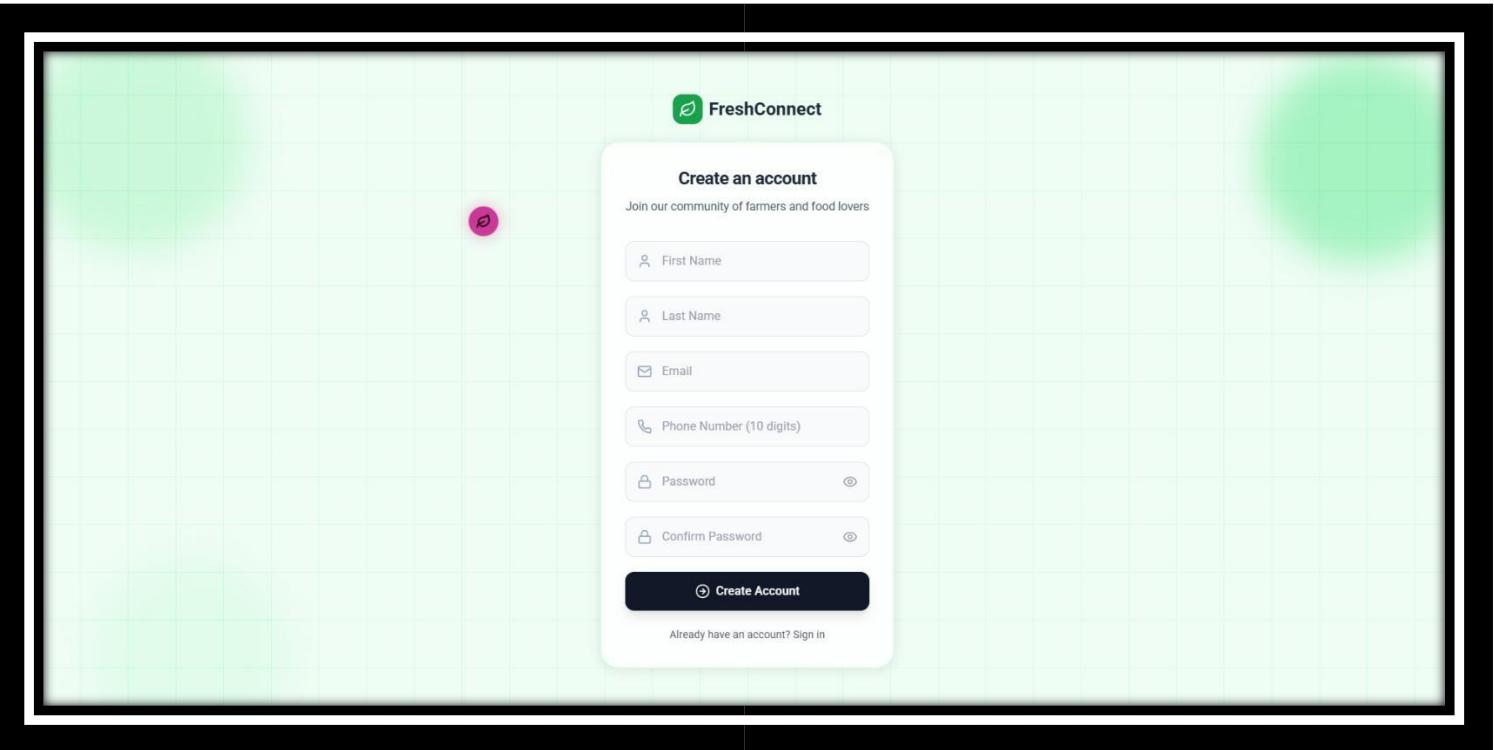
Footer page:



log in:



Sign up page :



Product page:

The screenshot displays the 'Browse Products' section of the FreshConnect website. It features a search bar, filters (Categories, Price Range, Product Type, Seller Type, Rating, Sort By), and a list of 22 products found. The products are shown in a grid format with images, names, ratings (all 0 stars), and prices (e.g., ₹54.00/kg). The filters sidebar lists various categories like 'Root Vegetables', 'Fruit Vegetables', 'Herbs', and 'Remediosdgdf'. The top navigation bar includes Market, Restaurants, Urgent Sales, Free Food, and a user profile icon.

Restaurant page :

FreshConnect

Market Restaurants Urgent Sales Free Food

Restaurants

Reviews & Feedback

Search restaurants... Filters

All Restaurants Verified Partners Nearby Trending Favorites

1 restaurants found

Sort by: Rating (High to Low)

King Palace (Verified)

King Palace

Urgent sale :

FreshConnect

Market Restaurants Urgent Sales Free Food

Urgent Food Sales

Help reduce food waste by purchasing these items at discounted prices before they expire.

Showing 10 of 10 products

Search for products, categories, or sellers Filters

All Products Expiring Soon High Discount Free Delivery Expired Reserved Sold

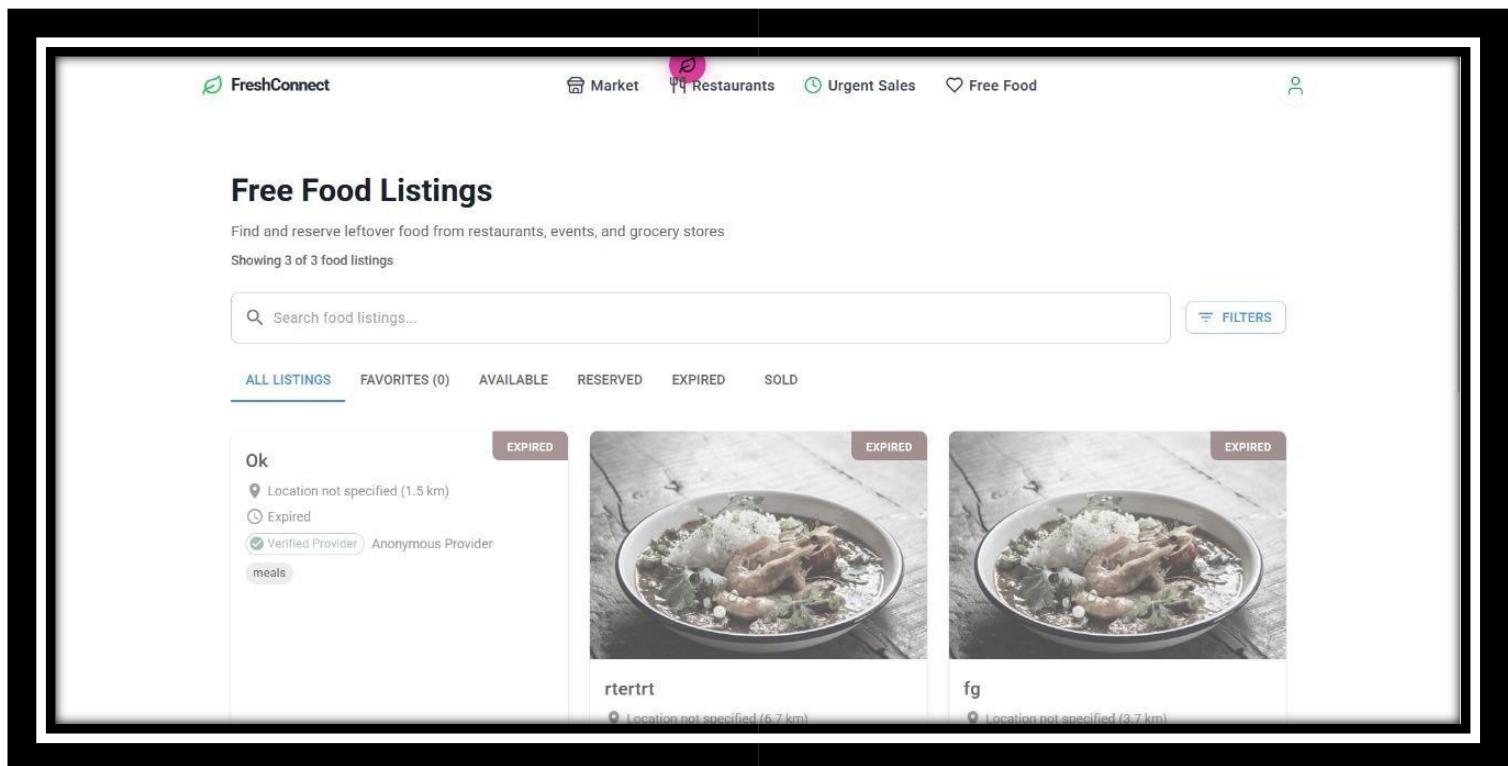


scjbhsv N/A

₹56.07

Unknown Seller

Free food listing page :



System Testing

7

Chapter

7.1 Software Testing 7.2 Test Case

Software Testing

○ Software Testing

Software testing is a critical element of software quality assurance and represents the ultimate review of specification design and coding. Testing is an exposure of a system to trial input to see whether it produces correct output. Testing cannot be determined whether software meets user's needs, only whether it appears to confirm to requirements. Testing can show that a system is free of errors, only that it contains error. Testing finds errors, it does not correct errors. Software success is a quality product, on time and within cost. Though testing can reveal critical (costly) mistakes. Testing should therefore,

- 1) Validate Performance
- 2) Detects Error
- 3) Identify Inconsistencies

○ Test Objective

There is strong evidence that effective requirement management leads to overall project cost savings. The three primary reasons for this are:

- Requirement errors typically cost well over 10 times more to repair than other errors.
- Requirement errors typically comprise over 40% of all errors in a software project.
- Small reduction in the number of requirement errors pays big dividends in avoided rework costs and schedule delays.

The testing procedure should care for all of these, as well as, in order to attain a flawless, error-free and efficient functioning system; too, software testing is an important phase of any software development life cycle. The system presented here is titled as --travelbuddy.com. Various reports and data used for

the same are the core of the system. The testing, therefore, becomes important in order to maintain the cost as well as improve performance and consistency. The testing procedure for the system has been divided in to various parts ranging from single unit testing to entire system testing.

○ Test Planning

Test planning evolves the following steps:

- Estimate the size and cost of the testing efforts.
- May allocate test resources.
- Schedule test activities in accord with other development work.
- Generate test cases.

○ Generating Test Case

A test case is a unit of testing activity. test cases have three parts:

- Goal- the aspect of the system being tested.
- Input and system state- data provided to the system under stated conditions.
- Expected Behaviour- the output or action the system should take according to these requirements.

○ Test Case Selection

The presented system --travlebuddy.com is tested using white- box testing which is based on the internal structure of the system, may also called as testing to code, path-oriented testing, or logical-driven testing. A common goal of white- box testing is to have a test case to exercise every path through a program. This may become exhaustive.

○ White Box Testing Case Selection

Because exhaustive white-box testing is infeasible, test or code coverage metrics are used to select test cases, which measure the fraction of code exercised by test cases. Test cases are selected to achieve target test coverage levels.

○ Test Coverage Metrics

- **Statement Coverage:** The fraction of statements executed at least once in running a collection of test cases.
- **Branch Coverage:** The fraction of total branch directions that have been taken at least once in running a collection of test cases.
- **Path Coverage:** The fraction of paths taken at least once in running a collection of test cases, where a Path is defined in some way to limit the totals of feasible collection.

○ Unit Testing

The presented system is tested first using unit testing. Unit testing is test of code written by a single programmer. A unit is a portion of a system implemented by a multi programmer. Unit testing is exercising a unit in isolation from the rest of the system. The system, then, is tested using integrating testing.

○ Integration Testing

- **Top-Down:** Combine, test and debug top-levelled routines that become the integration test that harms for lower-level units.
- **Bottom-Up:** Combine, test low-level routines into progressively larger modules and subsystems.
- **Sandwich:** Mainly top-down with bottom-up integration and testing applied to certain widely used components.

○ Failure and Recovery

The system is tested using sandwich testing so as it has integrated advantages of both the top-down integration as well as bottom-up integration testing.

O Advantages of using Top-Down Integration

- The top levels of the system are tested the most.
- Distributes integration and testing throughout the coding phase.
- Tends to make bug location easier.
- Minimizes test harness creation.

O Advantages of using Bottom-Up Integration

Works well early in the integration process; thorough testing is possible and bugs are easy to be found.

O Test Completion

- White-box test coverage targets are met.
- Rate of error discovery reduced the target value.
- Measured reliability of the system achieves its target value.

Testing Case

User module :

1. User Module (Common to all types)

Test Case			
ID	Description	Input	Expected Output
TC_U_01	Register a new user with valid details	Valid email, password, username	Success message, user added
TC_U_02	Register with existing email	Duplicate email	Error: Email already exists
TC_U_03	Login with correct credentials	Valid email/password	Logged in successfully
TC_U_04	Login with incorrect password	Wrong password	Error: Invalid credentials
TC_U_05	Change password with correct old password	Old, new password	Password updated
TC_U_06	Update profile info	New username/email	Profile updated
TC_U_07	Register user with invalid email format	Invalid email	Error message
TC_U_08	Login while account is inactive	Valid login but status inactive	Error: Account deactivated

Continue...

Vegetable seller :

2. Vegetable Seller

Test Case

ID	Description	Input	Expected Output
TC_VS_01	Add new product	All fields valid	Product listed successfully
TC_VS_02	Update product info	Valid changes	Product info updated
TC_VS_03	View orders as seller	Seller ID	List of received orders
TC_VS_04	Manage inventory	Reduce product quantity	Quantity updated correctly
TC_VS_05	List product with expiry in past	Past expiryDate	Error: Invalid expiry
TC_VS_06	Mark product as urgent	Product ID	Urgent tag added
TC_VS_07	Try listing product without verification	Not verified	Error: Verify first

Hotel owner :



3. Hotel Owner

Test Case ID	Description	Input	Expected Output
TC_HO_01	List leftover food	Valid food data	Food listed for pickup
TC_HO_02	View donation history	Hotel ID	List of donations
TC_HO_03	Update leftover food availability	Food ID, status	Status updated
TC_HO_04	Mark leftover food as claimed	Claimed = true	UI reflects claimed status

Customer :

4. Customer

Test Case ID	Description	Input	Expected Output
TC_C_01	Browse all products	None	Product list shown
TC_C_02	Add product to cart	Product ID	Product added to cart
TC_C_03	Add product to wishlist	Product ID	Wishlist updated
TC_C_04	Place valid order	Cart with items	Order placed, order ID generated
TC_C_05	Track an order	Valid order ID	Order status displayed
TC_C_06	Place order with empty cart	Empty cart	Error: Cart is empty
TC_C_07	Add expired product to cart	Product with past expiry	Error: Cannot add

Order detail :

6. Order / Order Detail

Test Case ID	Description	Input	Expected Output
TC_O_01	Calculate total for order	Items with quantity & price	Correct totalAmount
TC_O_02	Update delivery status	Delivered	Status updated
TC_O_03	Generate invoice	Order ID	PDF or text receipt
TC_OD_01	Calculate subtotal	2 items × ₹100	₹200 returned

Leftover food :



7. Leftover Food

Test Case ID	Description	Input	Expected Output
TC_LF_01	Mark as claimed	Valid food ID	Status changed
TC_LF_02	Update availability time	Valid date	AvailableTill updated
TC_LF_03	Claim expired leftover food	Date passed	Error: No longer available

Support ticket:

8. Support Tickets

Test Case ID	Description	Input	Expected Output
TC_ST_01	Create support ticket	Valid form input	Ticket created
TC_ST_02	Update ticket status	ticketId + status	Updated successfully
TC_ST_03	Escalate unresolved ticket	ticketId	Escalated to admin
TC_ST_04	Resolve a ticket	ticketId	Ticket closed
TC_ST_05	Create ticket with empty message	Empty form	Error: Message required

Admin: **9. Admin**

Test Case ID	Description	Input	Expected Output
TC_A_01	Verify vegetable seller	sellerId	Status set to verified
TC_A_02	Ban a hotel owner	hotelId	Access revoked
TC_A_03	Generate usage report	Date range	Report displayed
TC_A_04	Manage all users	None	List of users displayed

Service:

10. Services

◆ **PaymentService**

Test Case ID	Description	Input	Expected Output
TC_PS_01	Process payment	Valid payment info	Payment successful
TC_PS_02	Refund a payment	Order ID	Refund processed
TC_PS_03	Generate receipt	Order ID	Receipt returned

Authontication service :

◆ AuthenticationService

Test Case ID	Description	Input	Expected Output
TC_AS_01	Validate correct user	email & password	True
TC_AS_02	Generate token	userId	Token returned
TC_AS_03	Verify expired token	Expired token	Error: Invalid token

Analytics service:

◆ **AnalyticsService**

Test Case ID	Description	Input	Expected Output
TC_ANA_01	Generate sales report	Seller ID	Graph/report output
TC_ANA_02	Track user activity	userId	Activity logs
TC_ANA_03	Analyze trends	Date range	Trend insights

8. Limitation

Limitation

Sellers cannot delete their products once listed on the platform.

Hotel managers cannot remove hotel listings or room details after submission.

Users cannot directly cancel bookings through the website; they must contact the hotel or seller via email.

Users cannot live chat or message with sellers or hotel managers through the platform.

Admin cannot revert verification status once approved — needs backend support.

Users cannot give feedback without completing a booking

References

9

Chapter

9. References

References

- <https://nodejs.org/>
- <https://mongoosejs.com/>
- <https://app.diagrams.net/>
- <http://www.reactjs.org/>
- <http://www.stackoverflow.com/>
- <http://www.npmjs.com/>
- <http://www.youtube.com>