

Farmit: Organic E-commerce Website

Comprehensive Documentation



By Amritanshu Raj

22BCE3427

EXECUTIVE SUMMARY

Farmit is an innovative and comprehensive e-commerce platform dedicated to promoting healthy, sustainable living by offering a wide range of organic products. This platform is thoughtfully designed to connect organic farmers and sellers directly with consumers who prioritize health, wellness, and environmental sustainability. Farmit goes beyond just selling products—it fosters a community around the principles of organic living, providing a trusted marketplace for high-quality, fresh organic produce and goods.

The Farmit platform enhances the shopping experience with features that cater to the needs of both customers and administrators. Consumers can easily browse a diverse selection of organic items, add them to their cart, and track orders—all through a user-friendly and intuitive interface. For administrators and sellers, Farmit offers robust back-end tools to manage inventory, track orders, update product listings, and communicate with customers, ensuring smooth and efficient platform operations.

The platform also emphasizes transparency and trust, allowing consumers to know the origins of their products and the standards followed by the producers. With a strong focus on organic integrity, sustainability, and customer satisfaction, Farmit not only facilitates transactions but also strengthens the relationships between conscientious consumers and ethical organic suppliers.

Source Code: [click here](#)

TABLE OF CONTENTS

1. Introduction-----	4
2. Technical Architecture-----	5
3. Database Design-----	6
4. User Interface -----	8
5. Customer Portal -----	9
6. Administrative System -----	13
7. Security Implementation -----	17
8. Installation & Setup -----	17
9. Maintenance & Support -----	18
10. Future Enhancements -----	19
11. Appendix -----	19
A. Database Queries to make tables -----	19
B. Code of ODBC Connection -----	21
C. Software Used -----	22
D. References -----	22

1. INTRODUCTION

1.1 PROJECT OVERVIEW

Farmit serves as a digital marketplace for organic products, implementing a user-friendly interface for customers while maintaining powerful backend capabilities for administrators. The platform emphasizes ease of use, security, and efficient order management.

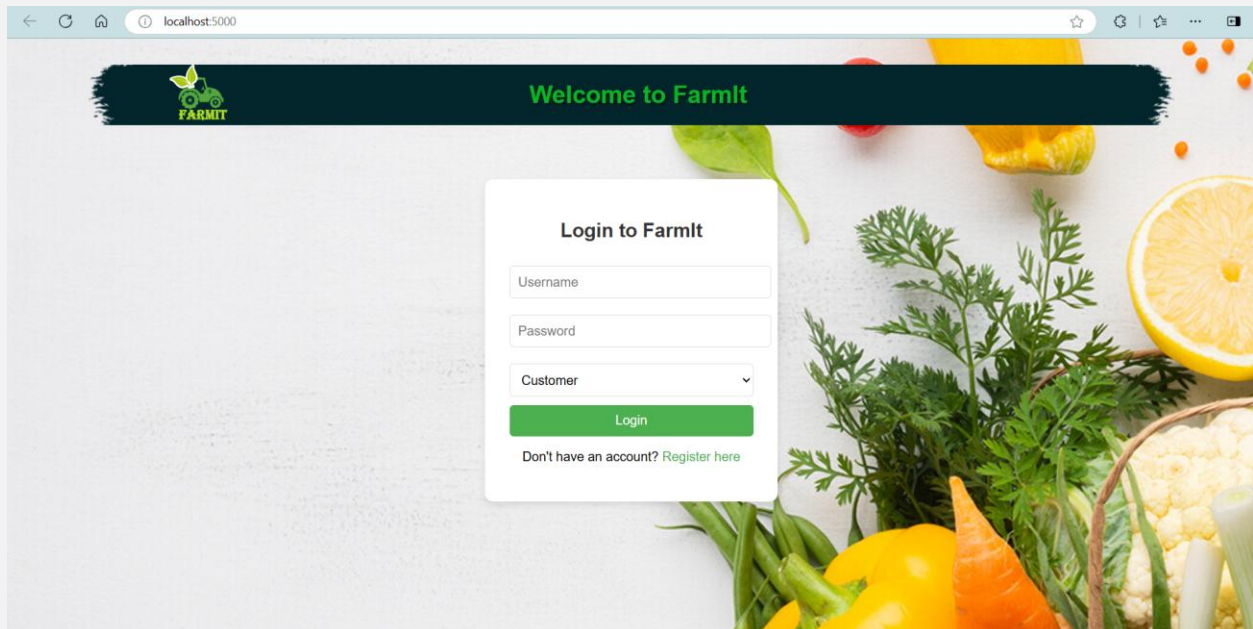
1.2 TARGET AUDIENCE

- Primary Users: Health-conscious consumers seeking organic products
- Secondary Users: Administrators managing inventory and orders
- Tertiary Users: Customer service representatives monitoring feedback

1.3 KEY FEATURES

- Dual interface system (Customer/Admin)
- Real-time inventory management
- Secure authentication
- Newsletter subscription system
- Order tracking and management
- Customer feedback system

Take a look on Landing Page:



2. TECHNICAL ARCHITECTURE

2.1 FRONTEND TECHNOLOGIES

- **HTML5:** Structure and content
- **CSS3:** Styling and responsive design
- **JavaScript:** Client-side functionality
- **Bootstrap:** Responsive framework
- **jQuery:** DOM manipulation and AJAX calls

2.2 BACKEND TECHNOLOGIES

- **Node.js:** Server runtime environment
- **Express.js:** Web application framework
- **MySQL:** Database management
- **JWT:** Token-based authentication

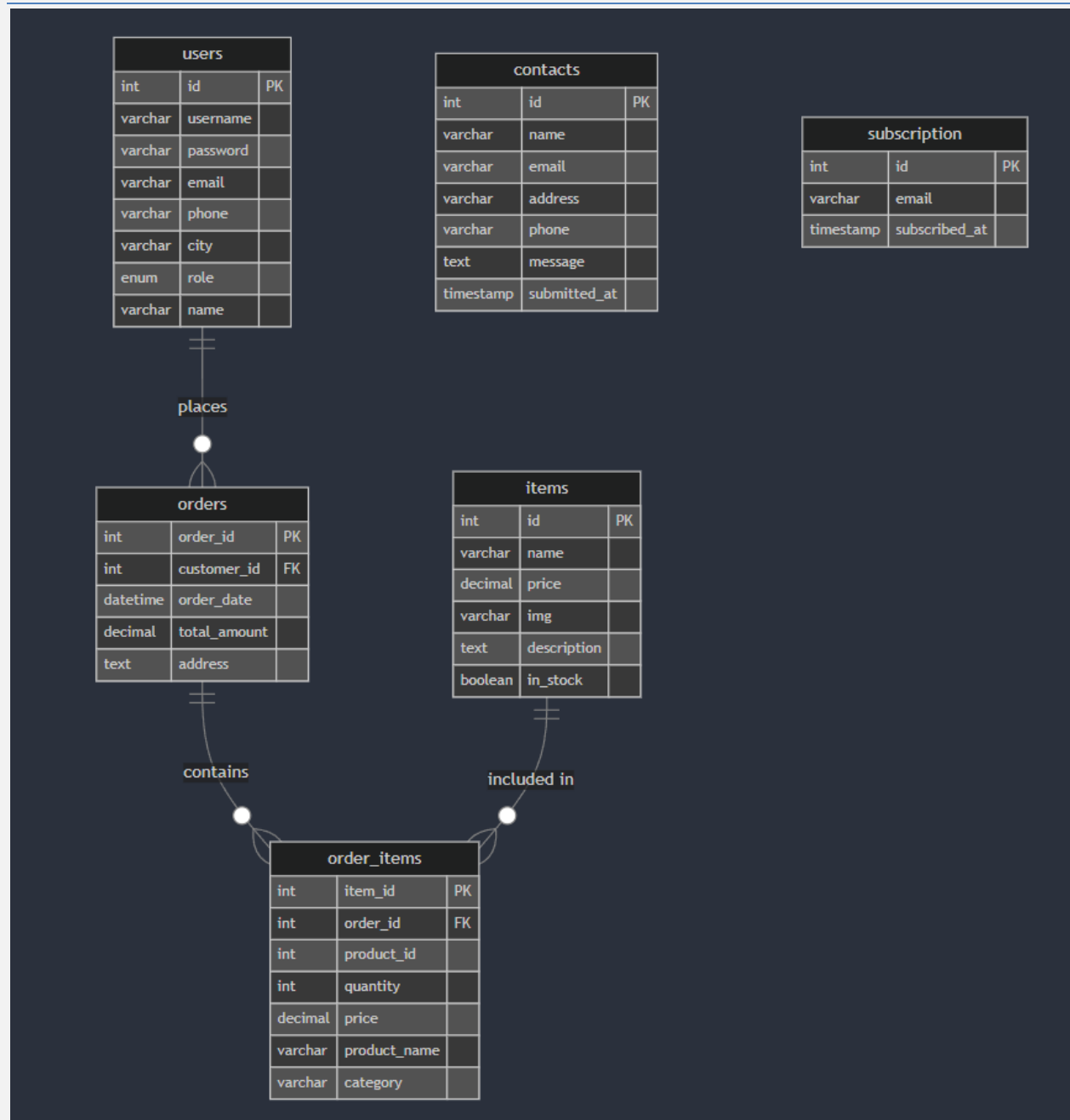
2.3 DEPENDENCIES

```
{  
  "body-parser": "^1.20.3", // Request parsing  
  "dotenv": "^16.4.5",      // Environment configuration  
  "express": "^4.21.1",     // Web framework  
  "express-fileupload": "^1.5.1", // File upload handling  
  "express-session": "^1.18.1", // Session management  
  "jsonwebtoken": "^9.0.2",  // Authentication  
  "mysql2": "^3.11.4",      // Database connector  
  "odbc": "^2.4.9"          // Database connectivity  
}
```

3. DATABASE DESIGN

3.1 SCHEMA OVERVIEW

The database architecture consists of six primary tables, each serving specific functionality within the system.



3.2 TABLE DESCRIPTIONS

USERS TABLE

- Primary storage for user information
 - Handles both customer and admin accounts
 - Includes essential contact information
 - Role-based access control implementation
-

ORDERS TABLE

- Tracks all customer purchases
 - Links to user accounts
 - Maintains order history
 - Stores delivery information
-

ORDER ITEMS TABLE

- Details of products in each order
 - Quantity and pricing information
 - Category classification
 - Product status tracking
-

ITEMS TABLE

- Product inventory management
 - Stock status tracking
 - Pricing information
 - Product descriptions and images
-

CONTACTS TABLE

- Customer feedback storage
 - Support ticket tracking
 - Timestamp for response management
 - Contact information verification
-

SUBSCRIPTION TABLE

- Newsletter subscription management
 - Email verification
 - Subscription timestamp tracking
 - Unique email enforcement
-

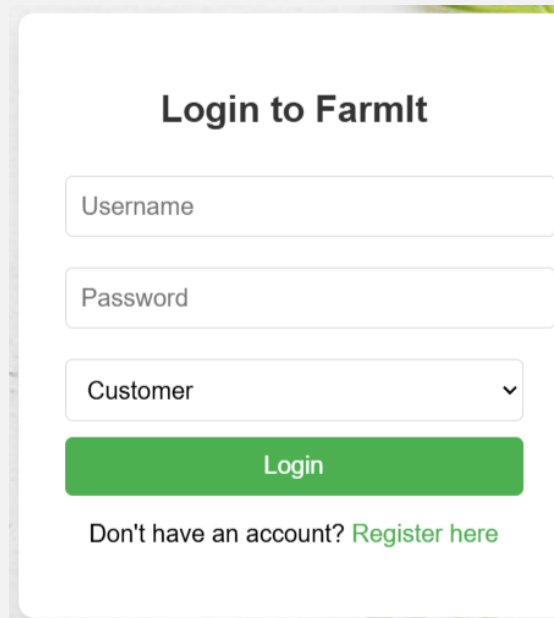
4. USER INTERFACE

4.1 LOGIN SYSTEM

The application implements a secure login system with role-based access control.

FEATURES:

- User role selection (Customer/Admin)
- Session management
- Password encryption

A login form titled "Login to Farmlt" with a white background and rounded corners. It contains three input fields: "Username", "Password", and a dropdown menu currently showing "Customer". Below these fields is a green "Login" button. At the bottom, there is a link that says "Don't have an account? Register here".

Login to Farmlt

Username

Password

Customer ▼

Login

Don't have an account? [Register here](#)

4.2 REGISTRATION PROCESS

New user registration captures essential information while implementing necessary validations.

REQUIRED INFORMATION:

- Username (unique)
- Password (with strength requirements)
- Email verification
- Contact details
- City
- Full Name

Create an Account

City

Register

Already have an account? [Login here](#)

5. CUSTOMER PORTAL

5.1 NAVIGATION STRUCTURE

- Home: Landing page with featured products
- About: Company information and values
- Products: Categorized product listings
- Contact: Customer support access
- Cart: Shopping cart management

5.2 PRODUCT CATEGORIES

- Fruits: Fresh organic fruits
- Vegetables: Organic vegetables
- Organic Products: Additional organic items

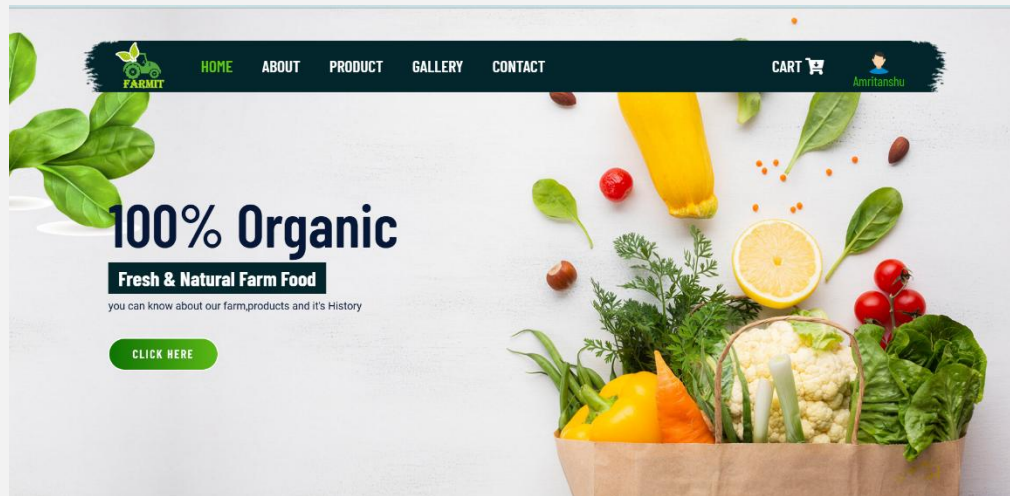
5.3 SHOPPING EXPERIENCE

Step-by-step process:

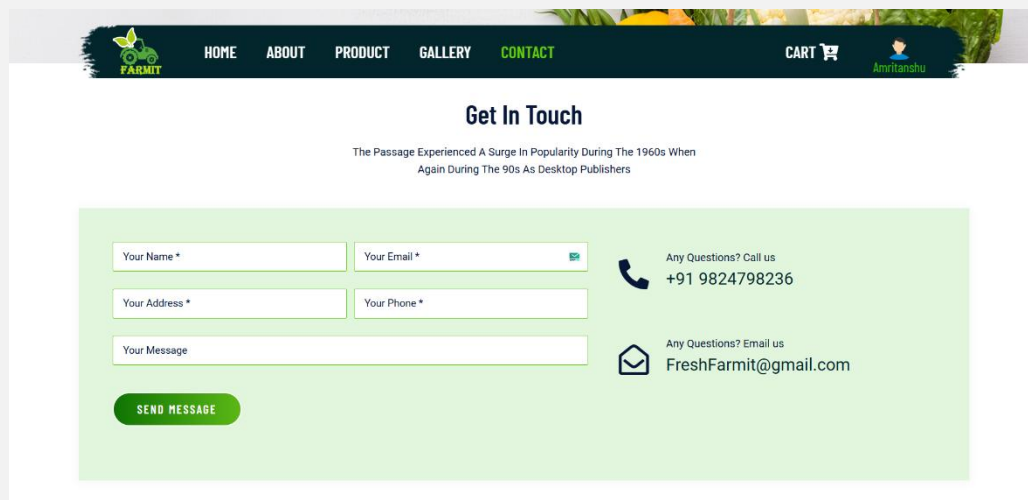
1. Product browsing
2. Cart addition
3. Quantity adjustment
4. Address confirmation
5. Order placement
6. Order confirmation

5.4 SCREENSHOTS

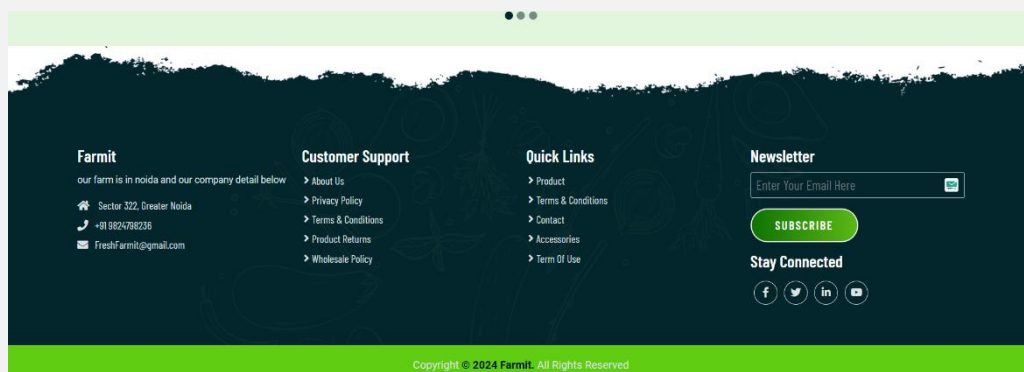
Homepage:



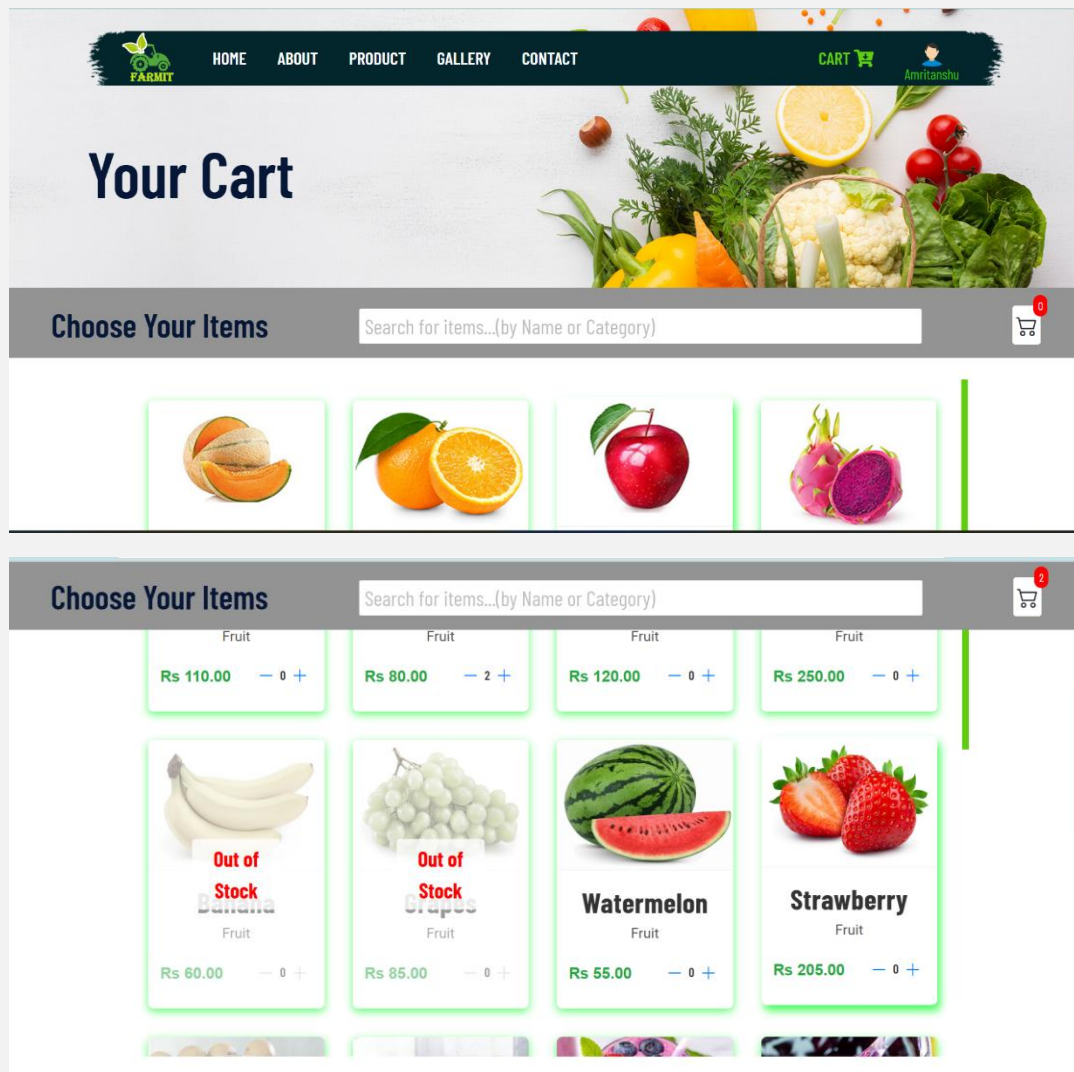
Contact-form:



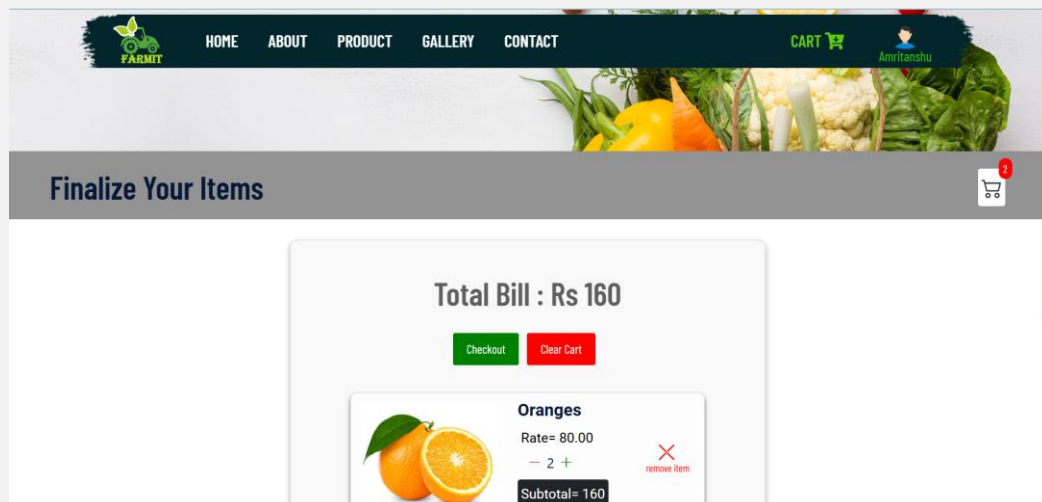
Footer:



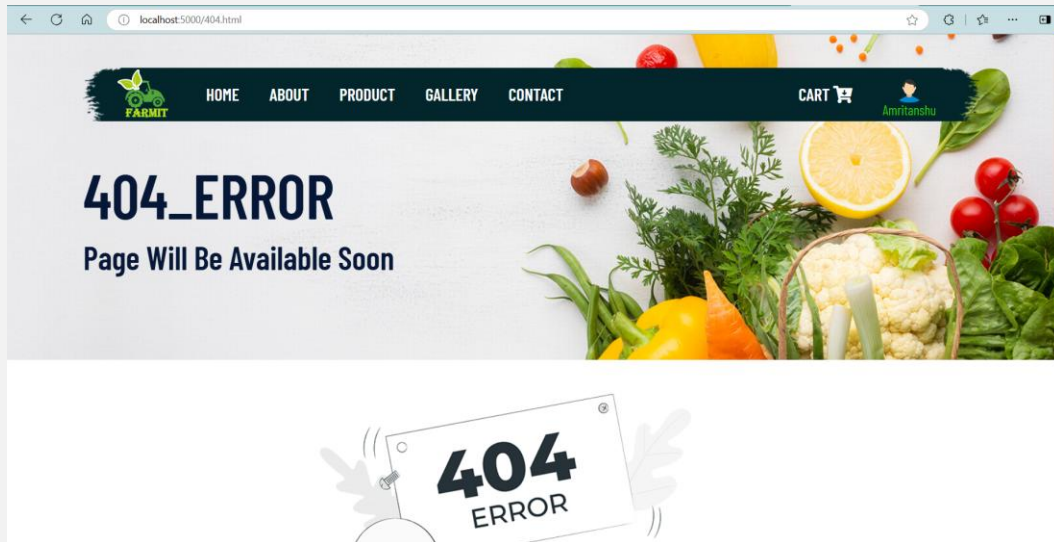
Cart:



Checkout-Page:



Error page:



Shipping Details form:

SHIPPING DETAILS

House Name/Number*

Enter house name/number

Street Name*

Enter street name

City*

Enter city

Pincode*

Enter pincode

State*

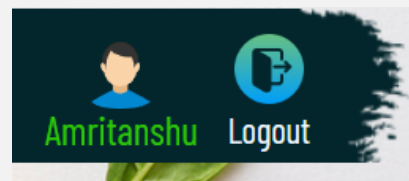
Enter state

Phone Number*

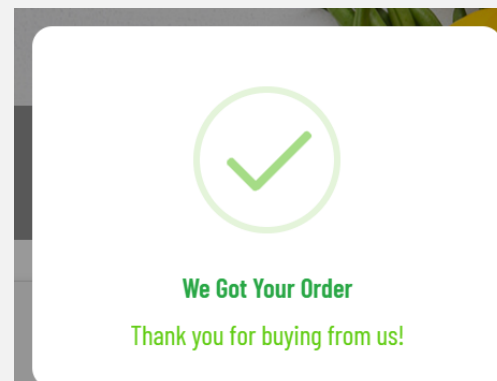
Enter phone number

PROCEED TO CHECKOUT CANCEL

logout Option:



Order-confirmation:



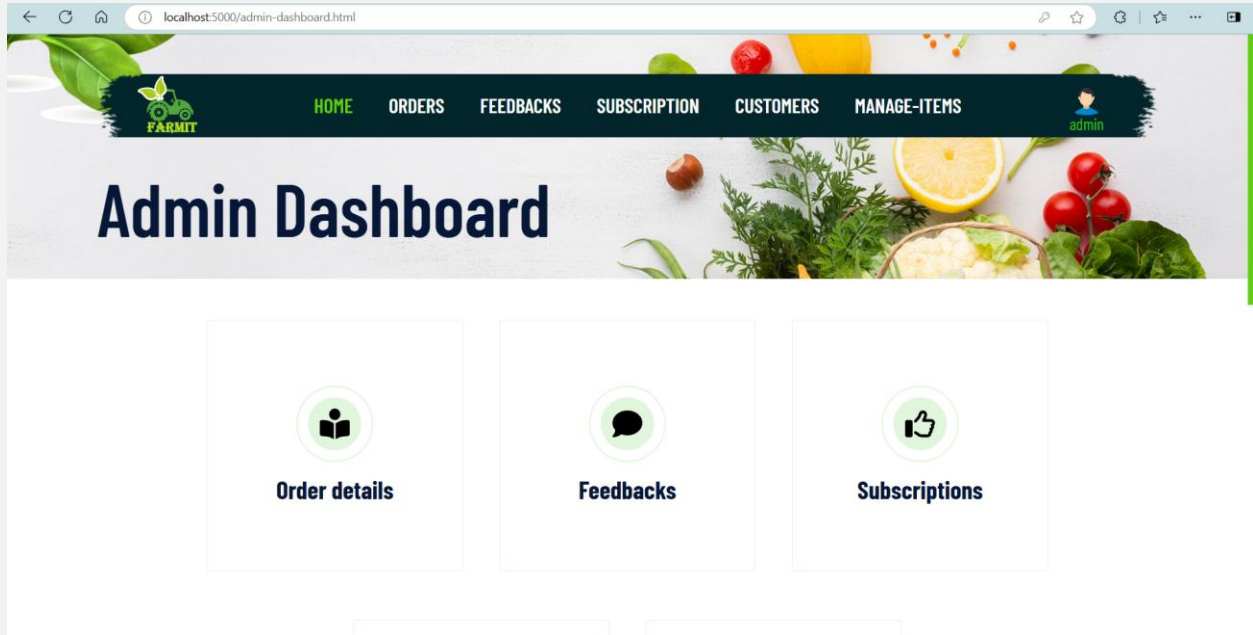
5.4 ADDITIONAL FEATURES

- Newsletter subscription
- Account management
- Feedback submission

6. ADMINISTRATIVE SYSTEM

6.1 DASHBOARD OVERVIEW

Comprehensive admin control panel with real-time updates.



6.2 KEY FUNCTIONALITIES

ORDER MANAGEMENT

- Order tracking
- Status updates
- Export capabilities
- Customer communication

INVENTORY CONTROL

- Stock management
- Product addition/removal
- Price updates
- Category management

USER MANAGEMENT

- Customer account oversight
- Order history tracking
- Account status control
- Admin account creation

REPORTING SYSTEM

- Sales reports
- Inventory reports
- Customer analytics
- Subscription tracking

SCREENSHOTS

Orders:

Order Details

Customer Orders [EXPORT AS XLSX](#)

Order #6

Customer Name: nitish kumar
Address: vit men's hostel,c block, tiruvalam road, Vellore- 632014, TAMIL NADU, Phone: 6203419519
Order Date: 13/11/2024
Order Time: 5:48:47 pm
Total Amount: ₹435.00

Product ID	Product Name	Category	Quantity	Price per Item	Subtotal
------------	--------------	----------	----------	----------------	----------


Feedbacks:

Feedbacks

Customer Feedbacks [EXPORT AS XLSX](#)

#	Name	Email	Address	Phone	Message	Submitted At
1	shubham Ghimire	shubham12@gmail.com	vit vellore men's hostel	8932729323	Excellent service and high-quality product! Iâ€™m very satisfied with the attention to detail and prompt support provided. Truly a delightful experienceâ€”thank you for exceeding expectations. Will definitely recommend to others!	13/11/2024, 4:53:17 am

Subscription:



HOMEORDERSFEEDBACKSSUBSCRIPTIONCUSTOMERSMANAGE-ITEMSadmin

Subscriptions

anshuraj26@gmail.com
Subscribed on: 13/11/2024, 7:05:03 pm

nitish47@gmail.com
Subscribed on: 13/11/2024, 5:35:42 pm

anshuraj2840@gmail.com
Subscribed on: 13/11/2024, 4:53:30 am


FarmIt

Customer Support

Quick Links

Stay Connected

Customers:



HOMEORDERSFEEDBACKSSUBSCRIPTIONCUSTOMERSMANAGE-ITEMSadmin

Customer Details

Customer List

EXPORT AS XLSX

Customer ID	Name	Email	Phone	City	Total Orders
4	Himanshu Raj	himanshu123@gmail.com	6254363356	prayagraj	2
5	nitish kumar	nitish@gmail.com	4738837483	patna	2


localhost:5000/customer.html


Footer of admin-dashboard:


REGISTER ADMIN/CUSTOMER

FarmIt

our farm is in noida and our company detail below

 Sector 322, Greater Noida

 +91 9824782336

 FreshFarmIt@gmail.com

DELETE MY ACCOUNT

Customer Support

> About Us

> Privacy Policy

> Terms & Conditions

> Product Returns

> Wholesale Policy

Quick Links

> Product


> Terms & Conditions


> Contact


> Accessories


> Term Of Use

Stay Connected





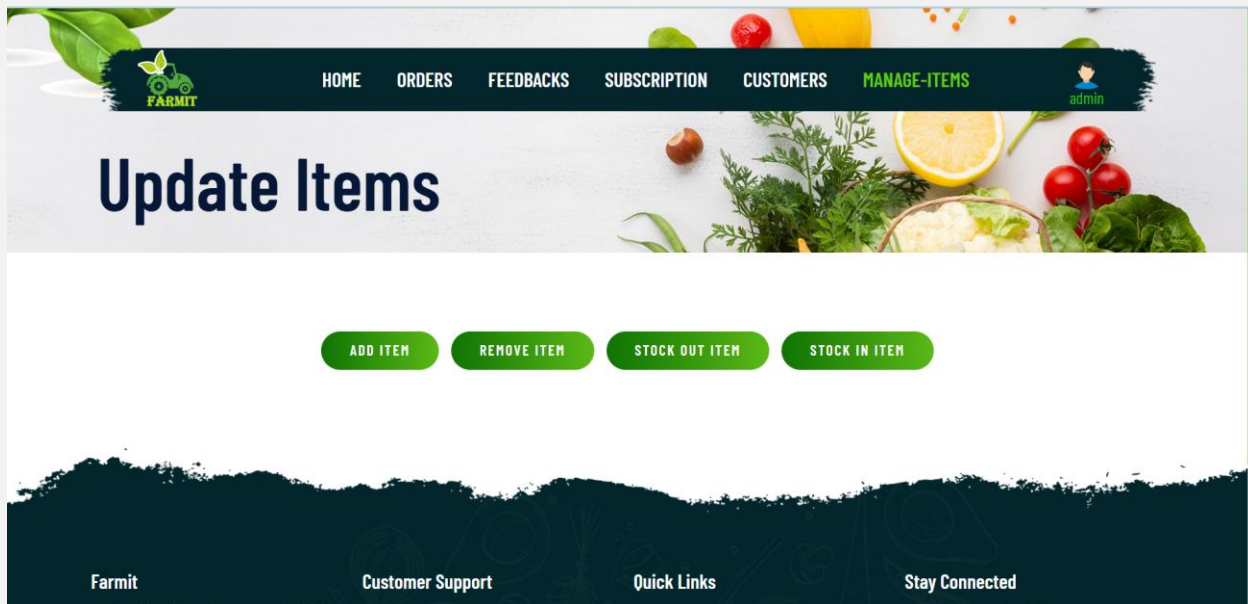




localhost:5000/admin-dashboard.html

Copyright © 2024 FarmIt. All Rights Reserved

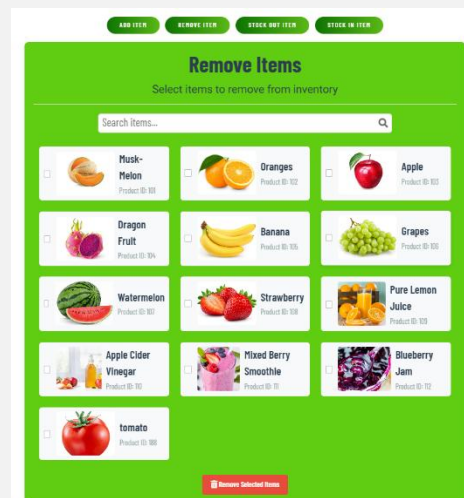
Manage-Items:



Add-item:

The screenshot shows the 'Add New Item' form. At the top, there are four green buttons: 'ADD ITEM' (highlighted), 'REMOVE ITEM', 'STOCK OUT ITEM', and 'STOCK IN ITEM'. Below these buttons, the form is titled 'Add New Item'. It contains four input fields: 'Name:' with the placeholder 'Enter the item name', 'Price:' with the placeholder 'Enter the item price', 'Category:' with the placeholder 'Enter the category', and 'Image:' with a 'Choose File' button and the text 'No file chosen'. At the bottom of the form is a green 'Add Item' button.

Remove-items:



Search feature in admin side:



Search feature in Customer side:



7. SECURITY IMPLEMENTATION

7.1 SESSION MANAGEMENT

- Single active session enforcement
- Session timeout handling
- Secure session storage
- Cross-tab protection

7.2 ACCESS CONTROL

- Role-based permissions
 - Route protection
 - API security
 - Data encryption
-

8. INSTALLATION GUIDE

8.1 PREREQUISITES

- Node.js (v14+)
- MySQL (v8+)
- npm package manager

8.2 SETUP PROCESS

```
# Clone repository
git clone [AmritanshuRaj45/Farmit_ecommerceWeb]

# Install dependencies
npm install

# Configure environment
cp .env.example .env

# Initialize database
mysql -u root -p < schema.sql

# Start application
npm start
```

8.3 CONFIGURATION

Required environment variables:

DB_HOST=localhost

DB_USER=root

DB_PASSWORD=your_password

DB_NAME=farmit

JWT_SECRET=your_secret

SESSION_SECRET=your_session_secret

9. MAINTENANCE & SUPPORT

9.1 REGULAR MAINTENANCE

- Performance optimization
- Bug fixes

9.2 TROUBLESHOOTING

Common issues and solutions:

- Session conflicts
- Database connectivity
- File upload issues
- Authentication problems

9.3 SUPPORT CHANNELS

- Technical support email
 - Bug reporting system
 - Feature request process
 - Documentation updates
-

10. FUTURE ENHANCEMENTS

10.1 PLANNED FEATURES

- OTP-based registraion
- Advanced analytics

10.2 SCALABILITY PLANS

- Load balancing implementation
- Database optimization
- Caching system
- Performance monitoring

APPENDIX

A. DATABASE QUERIES TO MAKE TABLES

create database farmit;

use farmit;

CREATE TABLE users (

id INT AUTO_INCREMENT PRIMARY KEY,

username VARCHAR(50) NOT NULL UNIQUE,

password VARCHAR(255) NOT NULL,

email VARCHAR(100) NOT NULL,

phone VARCHAR(15) NOT NULL,

city VARCHAR(100) NOT NULL,

role ENUM('customer', 'admin') NOT NULL,

name varchar(50) NOT NULL

);

CREATE TABLE orders (

order_id INT AUTO_INCREMENT PRIMARY KEY,

customer_id INT,

order_date DATETIME DEFAULT CURRENT_TIMESTAMP,

total_amount DECIMAL(10, 2),

address TEXT,

FOREIGN KEY (customer_id) REFERENCES users(id)

```

);

CREATE TABLE order_items (

    item_id INT AUTO_INCREMENT PRIMARY KEY,

    order_id INT,

    product_id INT,

    quantity INT,

    price DECIMAL(10, 2),

    FOREIGN KEY (order_id) REFERENCES orders(order_id),

    product_name VARCHAR(255),

        category VARCHAR(255)

);

CREATE TABLE contacts (

    id INT AUTO_INCREMENT PRIMARY KEY,

    name VARCHAR(100) NOT NULL,

    email VARCHAR(100) NOT NULL,

    address VARCHAR(255) NOT NULL,

    phone VARCHAR(15) NOT NULL,

    message TEXT,

    submitted_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP

);

CREATE TABLE subscription (

    id INT AUTO_INCREMENT PRIMARY KEY,

    email VARCHAR(255) NOT NULL UNIQUE,

    subscribed_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP

);

CREATE TABLE items (

    id INT PRIMARY KEY AUTO_INCREMENT,

    name VARCHAR(255) NOT NULL,

    price DECIMAL(10, 2) NOT NULL,

    img VARCHAR(255) NOT NULL,

    description TEXT

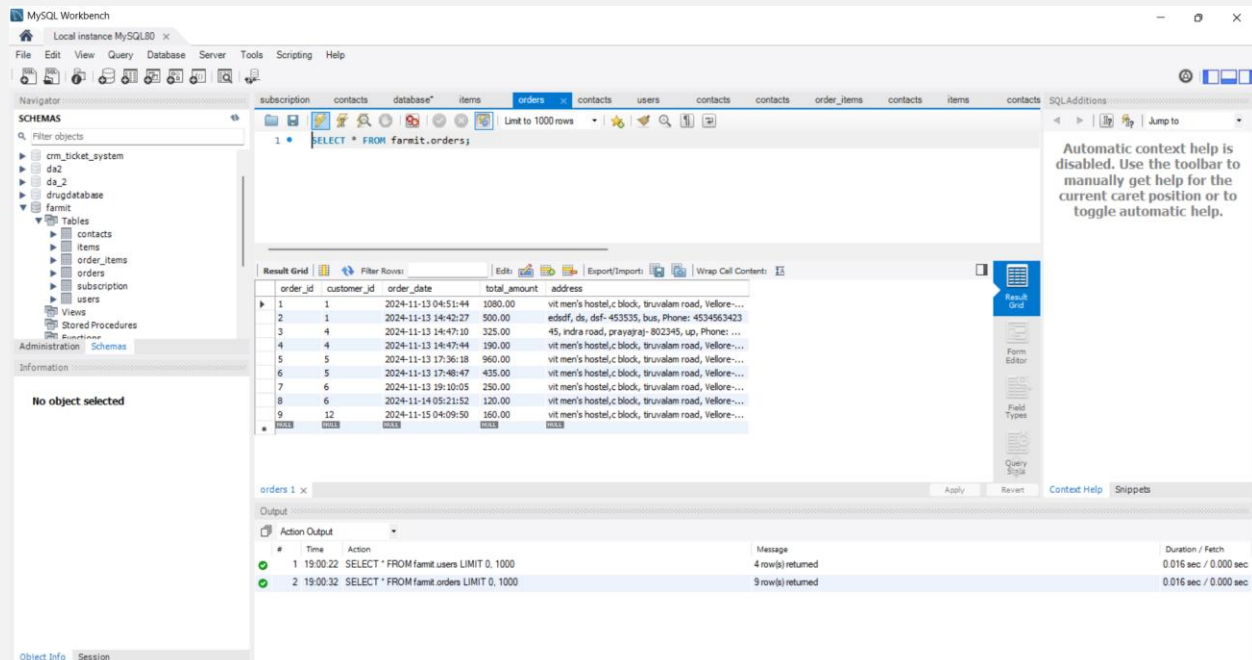
```

```
);  
  
-- Set the auto_increment starting value to 101  
  
ALTER TABLE items AUTO_INCREMENT = 101;  
  
ALTER TABLE items  
  
ADD COLUMN in_stock BOOLEAN NOT NULL DEFAULT TRUE;
```

B. CODE OF ODBC CONNECTION

```
// db/dbConnection.js  
  
const odbc = require('odbc');  
require('dotenv').config();  
  
const connectionString = process.env.ODBC_DSN;  
  
async function getConnection() {  
  try {  
    const connection = await odbc.connect(connectionString);  
    return connection;  
  } catch (error) {  
    console.error('Database connection failed:', error);  
    throw error;  
  }  
}  
  
module.exports = { getConnection };
```

C. SOFTWARE USED FOR MONITORING TABLES(MYSQL WORKBENCH)



D. REFERENCES

1. Technologies and Frameworks

- Node.js: <https://nodejs.org>
- Express.js: <https://expressjs.com>
- MySQL: <https://www.mysql.com>
- Bootstrap: <https://getbootstrap.com>
- jQuery: <https://jquery.com>

2. Libraries and Dependencies

- dotenv: <https://www.npmjs.com/package/dotenv>
- body-parser: <https://www.npmjs.com/package/body-parser>
- express-fileupload: <https://www.npmjs.com/package/express-fileupload>
- express-session: <https://www.npmjs.com/package/express-session>
- jsonwebtoken (JWT): <https://www.npmjs.com/package/jsonwebtoken>
- MySQL2: <https://www.npmjs.com/package/mysql2>

- ODBC: <https://www.npmjs.com/package/odbc>
- 3. **Design and Inspiration**
 - UI Design Reference: <https://dribbble.com>
 - Open-source Icons: <https://fontawesome.com>
- 4. **Guides and Tutorials**
 - MySQL Tutorials: <https://dev.mysql.com/doc/>
 - Node.js Documentation: <https://nodejs.org/docs/>
 - Express.js Guide: <https://expressjs.com/en/starter/guide.html>
- 5. **Images and Media**
 - Placeholder Images: <https://placeholder.com>
 - Free Icons and Illustrations: <https://undraw.co>
- 6. **Testing and Debugging Tools**
 - Postman: <https://www.postman.com>
 - Browser Developer Tools: Chrome DevTools (<https://developer.chrome.com>)
- 7. **Miscellaneous**
 - Git and GitHub: <https://github.com>
 - Stack Overflow: <https://stackoverflow.com>