Inventory and Business Management System for Agricultural supplies

Inventory and Business Management System for Agricultural Supplies

ABSTRACT

The Inventory and Business Management System for Agricultural Supplies System is a comprehensive digital solution designed to streamline and automate the core operations of agro-based retail businesses. This system caters specifically to shop owners who manage inventory, credit sales, suppliers, and customer relationships, particularly in rural and semi-urban agricultural sectors. It provides a secure user authentication system to ensure only authorized access, while enabling detailed inventory management across various product categories like fertilizers, pesticides, and seeds, with support for fixed and custom weight entries. The system offers real-time stock tracking, low stock alerts, and purchase order generation, simplifying restocking and supplier coordination. A robust billing and sales module captures daily transactions, supports bulk discounts, and generates automated sales reports. It also includes a dedicated credit sales module, allowing customers to purchase on credit, make partial payments, and view outstanding dues, with support for interest calculations and credit scoring. Customer profiles store complete transaction histories, preferred product details, and payment behaviors. All records are stored in a centralized database, allowing for seamless data retrieval and exportable reports in PDF format. By integrating modern digital tools into traditional agro-retail, the system enhances efficiency, transparency, and customer service while reducing manual workload and minimizing financial risks.

Key Words: Agricultural Shop Management, Inventory Management, Credit Sales Tracking, Partial Payment System, Admin Authentication

CONTENTS

Chapter 1

INTRODUCTION

1.1 Motivation

The agricultural sector is the backbone of many national economies, particularly in developing countries. While much focus is placed on farm-level interventions and innovations, the support ecosystem that bridges farmers with essential agricultural inputs often remains neglected. Small and medium-sized agri-input retailers are critical nodes in this system, responsible for distributing seeds, fertilizers, tools, and advisory services. Yet, they face persistent bottlenecks due to lack of digital tools and structured business processes.

This project is born from a recognition of the untapped potential within these retail networks. Currently, most of these shops operate manually, with paper-based billing, handwritten inventory logs, and limited customer management systems. This leads to overstocking, understocking, delayed service, and customer dissatisfaction. Moreover, poor recordkeeping hinders their ability to apply for loans, build creditworthiness, or qualify for government support programs.

By designing a simple, intuitive digital system to manage inventory, billing, customer records, and reporting, this project empowers agri-retailers with tools previously accessible only to large, urban businesses. The motivation goes beyond mere digitization—it is about unlocking economic resilience and long-term sustainability for these local enterprises.

Furthermore, bridging the technological gap in rural commerce is vital for holistic digital inclusion. As governments and financial institutions roll out digital-first services, equipping agri-retailers with technology ensures they are not left behind. The solution could also pave the way for a larger digital agricultural ecosystem—integrating e-commerce, remote advisory, weather forecasting, supply chain tracking, and even AI-powered demand forecasting.

The system also addresses data transparency and business intelligence. With digital data collection, these retailers gain insights into seasonal demand patterns, top-selling products, and customer preferences. This not only helps with decision-making but also enables collaboration with suppliers, agri-tech startups, and logistics providers to create a more efficient and responsive supply chain.

In a broader sense, this project contributes to rural empowerment, gender inclusion (as many small shops are run by women), job creation through digital literacy, and overall rural economic development. By making technology accessible and purpose-driven, it transforms agri-retailers into competitive, future-ready businesses that can thrive in the digital era.

1.2 Problem Definition

Agricultural input retailers serve as the frontline support system for millions of farmers by supplying them with critical resources such as seeds, fertilizers, pesticides, tools, and technical guidance. Despite their indispensable role in the agricultural value chain, the management systems of these retail stores are still largely reliant on outdated, manual methods. This presents a multifaceted set of problems that hinder operational efficiency, business growth, and ultimately, farmer satisfaction.

1.2.1 Key Challenges Faced:

1. Fragmented Inventory Management:
    - Retailers often depend on handwritten logs or memory to track inventory.
    - This results in frequent mismatches between actual stock and recorded data, leading to over-purchasing, stockouts, or expired product accumulation.
2. Lack of Customer Relationship Management:
    - There is no system to record or recall customer history, preferences, or purchase frequency.
    - This impedes loyalty-building and limits personalized service.

3. Minimal Data-Driven Insights:
    - Retailers have no access to analytics on fast-moving products, seasonal demand trends, or supplier performance.
    - Business decisions are often made on guesswork rather than evidence.
4. Difficulty in Meeting Compliance Standards:
    - Taxation (GST), subsidy claims, and regulatory documentation are harder to maintain without proper software support.
5. Barriers to Digital Adoption:
    - Existing enterprise-level solutions are either too costly, complex, or irrelevant to the specific needs of rural agri-retailers.
    - Many store owners lack digital literacy or access to robust internet infrastructure.
6. Limited Integration with Supply Chain:
    - Delays in reordering stock from suppliers due to communication gaps lead to business disruptions and missed revenue opportunities.

These issues not only affect the profitability and growth of agri-retail businesses but also have a cascading impact on farmers, who depend on timely and reliable access to inputs. Delayed or incorrect product availability can compromise crop productivity and food security.

1.3 Objectives

The primary objective of this project is to design and develop a Inventory and Business Management System for Agricultural Supplies that streamlines and automates the daily operations of agricultural input retailers. The system is aimed at addressing the current inefficiencies and limitations faced by traditional manual practices, thereby improving business management.

1.3.1 Key Objectives

1. To Digitize Stock Management:
    - Enable real-time tracking of inventory levels.
    - Automate stock addition, depletion, and low-stock alerts.

2. To Simplify Billing and Invoicing:
    - Maintain accurate records of daily sales transactions.

3. To Manage Customer Information:
    - Store and retrieve customer purchase histories.
    - Support customer identification for tailored service or discounts.
4. To Maintain Vendor and Product Records:
    - Organize supplier contact information and product catalogues.
    - Simplify purchase order generation and supplier communication.
5. To Provide Administrative Controls:
    - Allow authorized admin access for stock updates, user management, and system configuration.
6. To Support Data Backup and Security:
    - Ensure the safe storage and recovery of critical business data.
    - Prevent unauthorized access to sensitive information.
7. To Improve Decision-Making with Reports:
    - Generate basic reports on sales, inventory trends, and customer data to support business insights.
8. To Offer a User-Friendly Interface:
    - Design a system that is simple enough for users with limited digital literacy.

1.4 Limitations

Despite offering several benefits, the Inventory and Business Management System for Agricultural Supplies System has certain limitations:

1. User Roles & Authentication

- Only one role (shop owner) is mentioned, and there is no flexibility to create additional roles for employees, which could limit access control.
- If users forget their passwords, the recovery process may be cumbersome or not secure enough, potentially causing issues with account access.

2. Inventory Management

- If the system doesn't integrate with barcode scanners or a POS system, stock updates may rely on manual entry, which can lead to human error or delays in real-time inventory tracking.
- For products with shelf life (like fertilizers and pesticides), the system does not appear to track expiration dates, which may lead to inventory management issues.

3. Supplier & Order Management

- There's no mention of tracking supplier reliability, such as delivery time, quality of products, or historical performance, which could affect future order decisions.
- Without automated order placement or integration with supplier systems, all orders will need to be entered manually, which can be time-consuming.
- The system might store basic supplier information but could lack the ability to track more detailed supplier data (like contact history, contracts, etc.).

- There’s no mention of communication tools with suppliers, such as order updates or issue resolution, which could streamline order management.

4. Sales & Billing System

- The system may only support traditional payment methods (cash, card), and may not be equipped to handle emerging options like mobile payments or online transactions.
- Sales reports may be too basic and might not allow for advanced filtering or detailed insights on sales trends, customer behavior, or product popularity.

5. Credit Sales for Customer Management

- Without the ability to set credit limits for customers, shop owners risk extending excessive credit to customers who may not be able to pay.
- If the system doesn't automatically notify customers about upcoming payments or overdue balances, it could lead to missed payments and collections issues.
- Tracking partial payments over time could become confusing if not properly managed, especially if customers make multiple payments towards a single outstanding balance.

6. Customer Profiles

- If the system doesn't integrate with customer loyalty programs or discounts, it could miss out on fostering customer retention through rewards and incentives.
- Manual Notes Management: Keeping track of custom notes about customer preferences may become disorganized if the system lacks robust search, tagging, or filtering capabilities.
-

1.5 Organization of Documentation

This project documentation is organized into multiple chapters, each covering a specific aspect of the development and implementation of the Inventory and Business Management System for Agricultural Supplies , platform. The structure is designed to guide the reader through the entire lifecycle of the project, from the initial idea to the final deployment.

Chapter 1 - Introduction
This project aims to develop a management system for agricultural shops, streamlining daily operations. It focuses on inventory, credit tracking, billing, and customer handling.

Chapter 2 - Literature Survey
A review of existing retail systems showed they lack agricultural-specific features. Key gaps include flexible unit handling, credit tracking, and simple interfaces for local users.

Chapter 3 - System Survey
Other systems were evaluated and compared based on features, user interface, and cost. Practical insights were taken from real shop environments to guide development.

Chapter 4 - System Analysis
Requirements were defined through interaction with shop owners. Feasibility studies confirmed that the system can be implemented efficiently using current technology.

Chapter 5 - Implementation
The system uses a modular approach with tools like HTML, CSS, JS, NodeJS, and MySQL. Key modules include login, inventory, billing, credit management, and customer profiles.

Chapter 6 - Testing
System functionalities were tested using real-world data. Common issues such as incorrect stock updates and payment tracking were identified and resolved.

Chapter 7 - Results and Discussion
The system proved effective in reducing manual errors and improving shop efficiency. Limitations such as manual data entry and basic reporting were also observed.

Chapter 8 - Conclusion and Future Work
The solution meets its objective of digitizing and simplifying shop operations. Future work includes mobile access, employee roles, automation, and advanced analytics.

Appendices
This section includes supporting materials such as diagrams, sample database schemas, user interface screenshots, sample test cases, and user manuals for reference.

Chapter 2

LITERATURE SURVEY

1.1 Introduction

Efficient inventory and credit management play a vital role in the operations of agricultural supply shops, where timely availability of products such as fertilizers, pesticides, and seeds is critical. As small and medium-scale agricultural retailers expand their customer base and product range, the need for a digital solution to streamline stock management, supplier tracking, sales automation, and credit handling has become more pressing. Existing literature highlights various retail management systems, yet many fail to address the unique requirements of agricultural supply chains, such as product-specific measurement units, bulk pricing, credit-based sales, and regulatory documentation for inspections.

This literature survey explores previous work in retail management systems, inventory control methods, credit tracking models, and customer relationship management (CRM) tools with a focus on their applicability to agricultural shops. It aims to identify current gaps in technology, evaluate commonly used solutions, and provide a foundation for developing a customized system tailored to the needs of agricultural retailers.

Several studies have focused on inventory optimization and stock alert systems, emphasizing automation to minimize wastage and restocking delays. However, most solutions cater to general retail environments and do not incorporate agricultural product constraints such as fixed-weight packaging, seasonality, and chemical safety regulations.

Existing CRM models support customer purchase tracking and personalized service but often lack features for managing credit payments with interest calculation and payment schedules. Additionally, the importance of maintaining detailed supplier records and generating compliant purchase reports for inspections is often overlooked in conventional systems.

This survey thus investigates existing models in inventory and credit management, evaluates their strengths and limitations, and sets the groundwork for developing a holistic and scalable solution tailored for agricultural supply shops.

2.2 Existing Systems

Various inventory and billing management systems are available in the market today. While some cater to general retail, very few focus on the unique needs of agricultural supply shops, such as fixed-weight product packaging, credit sales tracking, and inspection-friendly records. Here's a review of some commonly used systems:

1. Vyapar

- **Target**: Small businesses (retailers, wholesalers)
- **Features**:
  - Billing and invoicing
  - Inventory tracking
  - Basic credit management
  - GST reporting
- **Limitations**:
  - Limited to general retail—does not support agricultural unit types (like fixed 2.5kg, 5kg packaging)
  - No integrated credit scoring or payment installment features

2. Tally ERP 9

- **Target**: Medium to large enterprises
- **Features**:
  - Comprehensive accounting
  - Inventory control
  - Sales/purchase management
  - Credit/debit management
- **Limitations**:
  - Complex interface for small rural shop owners
  - Not mobile-friendly
  - Requires training to use effectively

3. Zoho Inventory

- **Target**: Online sellers, multi-channel retail
- **Features**:
  - Cloud-based inventory tracking
  - Supplier and customer management
  - Multi-warehouse support
- **Limitations**:
  - Requires stable internet
  - Subscription-based (expensive over time)
  - Lacks specific agricultural compliance or credit book handling

2.3 Disadvantages of Existing system

Here are the disadvantages of the existing systems used by agricultural supply shops:

1. **Lack of Custom Packaging Support**:
   Most existing software doesn't support fixed or non-standard packaging units like 2.5kg, 5kg, or milliliters/liters for products like fertilizers and pesticides. Shop owners must enter these manually, which increases errors.

2. **No Credit with Interest Management**:
   These systems typically do not support managing customer credit with interest calculations. There's no way to track partial payments or apply adjustable interest rates on remaining balances.

3. **Manual Handling of Bulk Discounts**:
   There is no built-in logic to apply or adjust discounts for bulk purchases. Shopkeepers must manually calculate and update prices, which is time-consuming.

4. **No Support for Customer Preferences or Notes**:
   Existing tools do not let shop owners store custom notes, preferred products, or personal discounts for regular customers, affecting customer service quality.

5. **Requires Continuous Internet Access**:
   These systems often depend entirely on cloud services, making them unusable in areas with unreliable or no internet connectivity.

6. **Complicated User Interfaces**:
   Many retail software platforms have complex navigation and features, which are not user-friendly for small shopkeepers or those with minimal technical experience.

7. **Unnecessary Features & High Costs**:
   General retail software may come bundled with features irrelevant to agri-shops (like GST filing, e-commerce integration) and can be costly for small-scale use.

8. **Lack of Feature Integration**:
   Important modules like inventory, sales, billing, credit management, and supplier tracking are often split across different tools, causing inefficiency and confusion.

2.4 Proposed System

The proposed system is a comprehensive desktop/web-based application tailored for shop owners selling agricultural supplies like pesticides, fertilizers, and seeds. It focuses on solving the pain points of existing retail systems by offering custom features relevant to agri-shops, especially in semi-urban and rural areas.

1. User Roles & Authentication

- Shop Owner Login:
  - Secure login with email/username and password.
- Authentication Features:
  - Password and confirm password validation.
  - Encrypted password storage.

2. Inventory Management

- Product Categories:
  - Pesticides (in mL/L)
  - Fertilizers (2.5kg, 5kg, 7.5kg, 25kg, 50kg, etc.)
  - Seeds (kg-based or packet-based)
- Inventory Book:

- Tracks product name, category, quantity, supplier, purchase date, and expiry date.
- Acts as proof of purchase for agricultural inspections.
- Image upload for each product.
- Notifies owner when stock drops below threshold.

3. Supplier & Order Management

- Supplier Records:
  - Stores supplier name, contact details, and historical transactions.
- Purchase Orders:
  - Create and manage orders placed with suppliers.
  - Track delivery status, quantity received, and purchase cost.
  - Allows better tracking of frequent suppliers and pricing history.

4. Sales & Billing System

- Daily Sales:
  - Total items sold.
  - Quantity sold per product.
  - Profit margin and discount applied.
- Bulk Purchase Discounts:
  - Manual and automatic discount logic for large quantity purchases.

5. Credit Sales & Customer Credit Management

- Credit Book:
  - Records customers who buy on credit.
  - Tracks outstanding amounts and due dates.
- Partial Payments:
  - Allows customers to repay in multiple installments.
- Interest Calculation:
  - Adjustable interest rate (e.g., 1.5% per month) applied only on remaining balance.
- Exportable Reports:
  - PDF/print option with filters like “Last 3 months” etc.

6. Customer Profiles

- Profile Management:
  - Store full name, contact info, preferred products, and payment history.
- Transaction History:
  - View all past purchases (cash, card, or credit).
- Notes Section:
  - Add special instructions, preferences (e.g., prefers organic seeds).
- Credit Overview:
  - View total balance, interest rate applied, and due date.

Chapter 3

ANALYSIS

3.1 Introduction

The analysis phase is one of the most crucial stages in the software development life cycle. It lays the foundation for designing a system that meets both the functional and non-functional needs of the users. In the context of this project — a Inventory and Business Management System for Agricultural Supplies — the analysis involves a comprehensive examination of existing practices, identification of inefficiencies, user needs assessment, and the scope of automation required.

Agricultural shop owners face numerous challenges due to the unique nature of their business, including handling inventory with varying units (like kilograms and liters), offering customer credit, managing partial payments, tracking interest, and maintaining personalized customer relationships. However, existing retail management solutions are either too generic, overly complex, or lack critical features tailored for the agricultural sector.

The goal of this analysis is to bridge the gap between the current state and the desired outcome by identifying:

- Core pain points in existing systems.
- Specific features required by agri-shop owners.
- Operational, regulatory, and user constraints.
- Real-world use cases and scenarios.

This phase involves several sub-tasks such as:

- Requirement Gathering through observation, discussion with shop owners, and user feedback.
- Feasibility Study to determine the practicality of the proposed solution.
- System Study to understand the current workflows and data management techniques.
- Gap Analysis between existing systems and agricultural supply shop requirements.
- Pain Point Analysis presented in tabular form.
- Comparative Study to evaluate the effectiveness of available retail software.

By performing this detailed analysis, we aim to develop a system that not only automates the operations but also simplifies decision-making for shop owners, reduces manual errors, and supports long-term growth through efficient credit and customer management.

3.2 Software Requirement Specification (SRS)

The Software Requirement Specification (SRS) is a structured document that outlines all the functional and non-functional requirements of a software system. It serves as a bridge between the client's expectations and the developer's implementation, ensuring all features are accounted for and developed according to specification. For this project, which focuses on automating the operations of an

agricultural shop using web technologies, the SRS defines the system's behavior, constraints, software environment, and interface design expectations.

The system will be built using the HTML, CSS, JavaScript for the frontend, and Node.js, Express, and MySQL for the backend. It will include secure login mechanisms, data encryption (via bcrypt), and file upload capabilities for inventory image handling. The SRS is divided into three core parts:

- **User Requirements**:
  What the user expects the system to do.
- **Software Requirements**:
  Tools, technologies, and environments used.
- **Hardware Requirements**:
  Physical devices and specifications needed to run the system efficiently.

  Each of these is detailed in the following subsections:

3.2.1 User Requirements

User requirements are high-level needs and expectations of the users that the system must fulfill. For the Inventory and Business Management System for Agricultural Supplies, the primary user is the shop owner, who is responsible for managing inventory, supplier orders, sales, billing, and customer credit. These users often have limited technical knowledge, so the system must focus on simplicity, clarity, and reliability. The goal is to provide a seamless digital platform that replaces manual processes with efficient, automated operations.

Key User Requirements:

1. **Secure User Authentication**
   - Shop owners must be able to register using a username, password, and confirm password.
   - Only authorized users should gain access to the system via secure login credentials.
   - Passwords must be stored securely using encryption (bcrypt).

2. **Dashboard Overview**
   - A simple dashboard should display stock levels, recent sales, low-stock alerts, and pending customer payments.
   - Quick access to credit information, inventory stats, and sales summaries.

3. **Inventory Management**
   - Ability to add, edit, or delete products from inventory.
   - Support for different units: kg, L, mL, with flexibility for fixed and custom weights.
   - Upload product images from local files for visual identification.
   - Alerts when stock reaches a minimum threshold.

4. **Supplier & Order Management**
    - Store and view supplier contact details and past orders.
    - Generate and track purchase orders.

5. **Sales & Billing System**
    - Track discounts, profit margins, and quantity sold.
    - Support for applying bulk purchase discounts manually.

6. **Credit Sales & Payment Tracking**
    - Maintain a credit book with outstanding balances.
    - Support for partial payments and automatic interest calculation on remaining balances.
    - Adjustable interest rates per customer.

7. **Customer Profiles**
    - Store individual customer purchase and payment histories.
    - Add custom notes for preferences (e.g., preferred brands or frequent purchases).

8. **Notifications and Alerts**
    - Low-stock alerts to prompt timely restocking.
    - Notifications for overdue credit payments.

These requirements ensure that the system is both functionally rich and easy to use, catering specifically to the needs of agricultural supply shop owners who require control, visibility, and accuracy in managing their daily business activities.

3.2.2 Software Requirements

The software requirements define all the software components, tools, libraries, and platforms essential for the development, deployment, and functioning of the Agricultural Supply Shop Management System. This project is built using HTML, CSS, JavaScript (frontend) and Node.js, Express.js, MySQL (backend) along with bcrypt for security and file upload support, developed in Visual Studio Code (VS Code) and MySQL Workbench. The system is designed as a full-stack web application with a focus on simplicity, speed, security, and scalability.

A. Frontend Requirements

The frontend is responsible for presenting an interactive and user-friendly interface to the shop owners. It is built using:

- **HTML5** – Used to structure the web pages including forms, tables, and content layout.
- **CSS3** – Used for designing and styling components such as buttons, forms, and alerts to enhance user experience.
- **JavaScript** – Adds interactivity such as form validation, dynamic content updates, and alert messages (e.g., low stock notifications).

B. Backend Requirements

The backend is responsible for managing the application logic, authentication, data validation, and database communication.

- **Node.js**: Executes JavaScript code on the server side, ensuring high performance and scalability.
- **Express.js**: A lightweight web framework used for building APIs, routing requests, and managing middleware.
- **bcrypt.js**: Hashes user passwords before storage, providing secure authentication.
- **fileupload**: Handles file/image uploads (e.g., product images).
- **CORS** : Allows secure Cross-Origin Resource Sharing, which is essential when frontend and backend are hosted on different ports during development or deployment.
- **body-parser**: Parses incoming JSON and URL-encoded data.

C. Database Requirements

- **MySQL** – A robust and scalable relational database management system used to store:
  - Inventory items and quantities
  - Supplier records
  - Sales and transaction details
  - Customer profiles and credit information
  - User credentials

- **MySQL Workbench** – A GUI tool used for designing the database schema, running SQL queries, and maintaining relationships between tables.

D. Development & Testing Tools

- **Visual Studio Code (VS Code)** – The primary IDE used for writing, editing, and debugging code.
- **Postman** – Used for testing backend APIs such as login, registration, product management, and customer payment tracking.
- **Browser Developer Tools** – For testing the frontend UI and debugging JavaScript.

E. Libraries & Packages (via npm)

- **express, mysql2, body-parser** – For server routing and MySQL connectivity.
- **Bcrypt** – For password hashing.
- **Cors** – Enables secure cross-origin requests between client/server.

This technology stack offers flexibility, security, and scalability, while remaining lightweight and cost-effective. Using CORS ensures secure communication between frontend and backend, especially during local or cloud-based deployment. All tools and libraries are open-source and community-supported, which guarantees continuous improvements and extensive documentation.

3.2.3 Hardware Requirements

Hardware requirements are the physical devices and components necessary for running the Inventory and Business Management System for Agricultural Supplies. Since this is a web-based application, the hardware specifications will largely depend on the infrastructure used for hosting the system, as well as the end-user devices accessing the system.

A. Development Hardware Requirements

For the development of the system, the following hardware is recommended:

1. **Personal Computer or Laptop**:
    - At least a Dual-Core Processor (Intel i5 or equivalent), but for better performance, an Intel i7 or higher is ideal for handling code execution, server hosting, and simultaneous processes like testing and debugging.
    - Minimum 8 GB of RAM, but 16 GB is preferred for smooth multitasking, especially when running multiple tools such as Visual Studio Code, MySQL Workbench, and the server at the same time.
    - A minimum of 500 GB of storage is necessary, with a preference for SSD (Solid State Drive) for faster data access and improved overall system performance.
    - Integrated graphics are sufficient, as this is a web-based application. A dedicated GPU is not required unless advanced visual rendering is needed in the future.

2. **Operating System**:
    - Windows 10/11 or Linux (Ubuntu), as they are the most compatible with the development tools like VS Code, Node.js, MySQL, and related dependencies.
    - macOS can also be used if the developer prefers it, but Windows or Linux is more common for development with Node.js.
3. **Internet Connectivity**:
    - Stable Internet Connection for downloading and updating software packages, interacting with cloud services (if used), and pushing code to version control (e.g., GitHub).
    - A high-speed connection is recommended, especially when working with cloud-based databases or deploying to remote servers.

B. End-User Hardware Requirements

End users, i.e., the shop owners and employees who will use the system to manage their business, will need devices to access the system. Since the application is web-based, the hardware requirements for these devices are minimal and can include:

1. **Personal Computers/Laptops**:
    - A basic **Dual-Core Processor** (e.g., Intel i3 or equivalent) is sufficient for shop owners to access and manage the system.
    - **4 GB of RAM** is sufficient for smooth operation of the browser and the web application.
    - **Minimum 100 GB of storage**, though storage needs are minimal since all data will be stored on the cloud or server.

- A modern browser (such as **Google Chrome**, **Mozilla Firefox**, or **Microsoft Edge**) with the latest version to ensure compatibility with web technologies like HTML5, CSS3, and JavaScript.

3.3 Architecture Diagram and Description of Project

1. Frontend (Client-Side)
   - Built using HTML, CSS, and JavaScript
   - Users interact with the system via browsers (Chrome, Firefox, Edge) or mobile devices.
   - Features include:
     - Login/Register pages
     - Inventory dashboard
     - Sales, credit book, customer details interface
2. Backend (Server-Side – Node.js + Express.js)
   - Acts as the middleware between the frontend and the database.
   - Handles:
     - User authentication (with bcrypt for hashing passwords)
     - API routing (GET, POST, PUT, DELETE)
     - File uploads (product images)
     - CORS management (enables secure cross-origin requests)
3. Database (MySQL)
   - Created and managed using MySQL Workbench
   - Stores:
     - Shop owner credentials
     - Inventory/product records
     - Supplier and order details
     - Sales and billing history
     - Credit sales and customer data

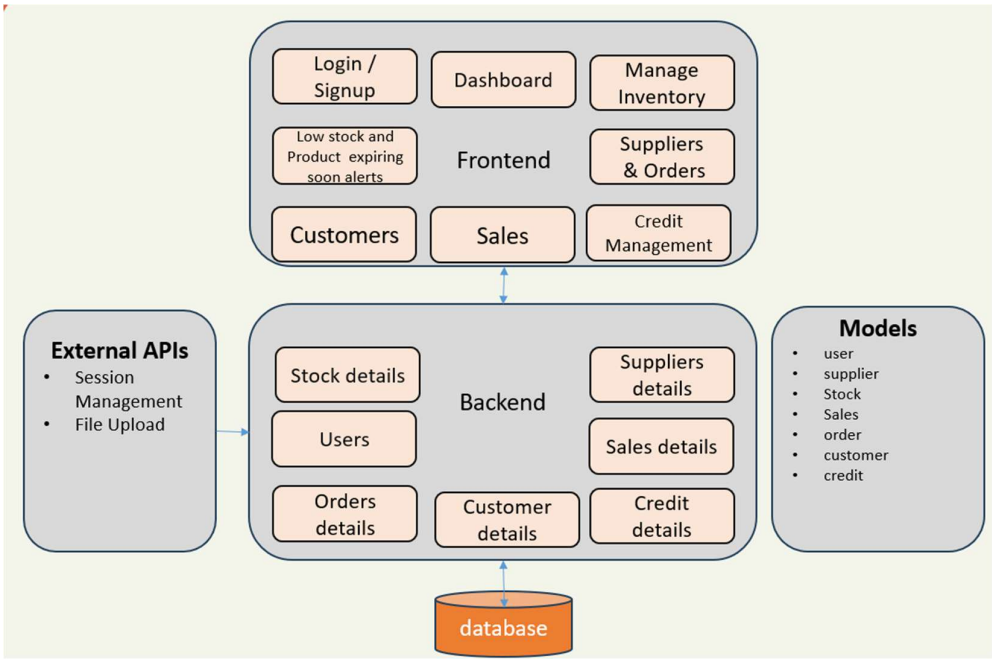


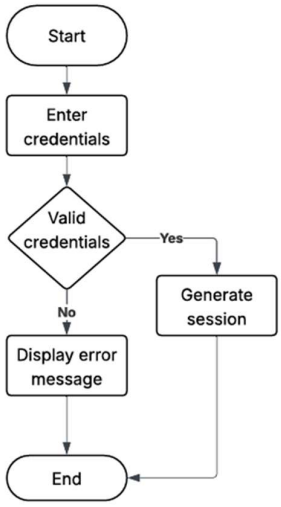
Figure 1: Architecture diagram

Workflow description:

The workflow of the Inventory and Business Management System for agricultural supplies is structured to ensure efficient, secure, and streamlined operations for managing inventory, sales, credit, and customer data. Below is a detailed breakdown of each phase in the workflow:

User Login Flow

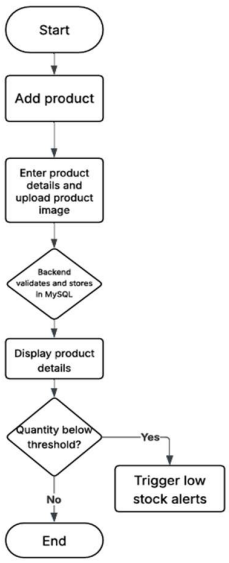
Figure 2 : User authentication workflow



- Shop owners register with their details and log in using secure credentials.
- Backend verifies login using bcrypt hashed passwords and CORS policy ensures access from trusted sources only.

Inventory Management Workflow

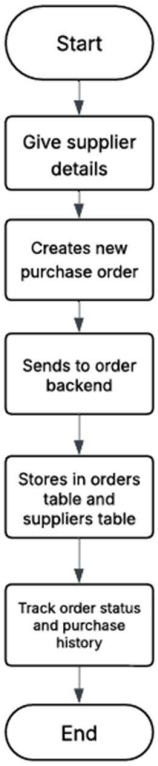
Figure 3: Inventory Management Workflow



- Upon login, the user accesses the dashboard.
- User fills in product details (name, category, price, quantity, expiry) and uploads an image.
- Backend validates data before inserting it into MySQL.
- The system automatically checks for low stock levels and sends alerts.
- Any change in quantity due to purchase or sale is reflected in real-time.

Supplier & Order Management Workflow

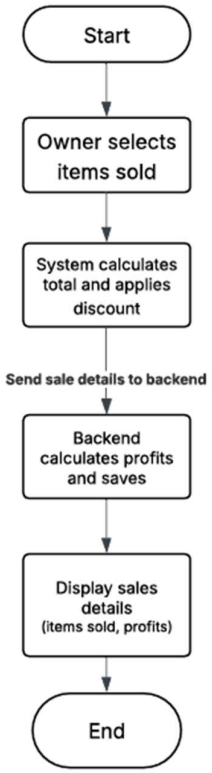
Figure 4 : Supplier & Order Management Workflow



- Shop owners add and view supplier details.
- Orders placed to suppliers are recorded along with dates and expected delivery.
- Maintains historical data for future reference and audits.

Sales & Billing Workflow

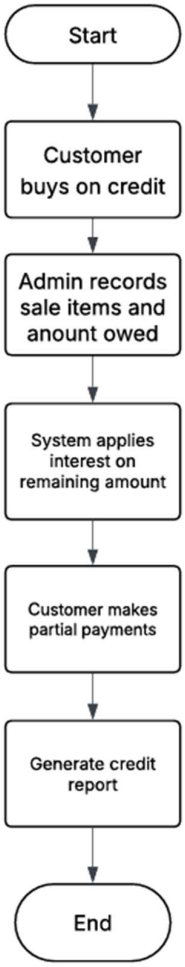
Figure 5: Sales and Billing Workflow



- When a sale is made, products are selected, and quantities/discounts applied.
- A final bill is calculated, displaying unit price, total, discounts, and payment method.
- Inventory is automatically adjusted after each sale.

Credit Sales Workflow

Figure 6: Credit Sales Workflow



- For credit purchases, the owed amount is recorded with date.
- Interest is applied to the unpaid balance based on defined algorithms.
- Allows recording of partial payments, updating balance and logs accordingly.
- System generates outstanding credit report and evaluates credit score based on payment behavior.

Customer Profile Management Workflow

Figure 6: Customer Profile Management Workflow

- Stores customer details, purchase history, and credit behavior.
- Owners can add custom remarks about preferences and deals.
- Tracks total credit, partial payments, and due amounts.
- PDF/Excel reports exportable for accounting or auditing purposes.

3.4 Algorithms, Flowcharts and Mathematical Formulas

The system incorporates a variety of algorithms to enhance efficiency and accuracy. These algorithms optimize processes, improve decision-making, and ensure seamless functionality.

1. **Interest Calculation Algorithms**
   Simple Interest:
   Interest = (Principal × Rate × Years) / 100
   Prorated Interest:
   Interest = (Balance × Rate × Days) / (100 × 30)
   Date Range Interest:
   Calculates interest between specific start and end dates.

2. **Stock Management Algorithms**

   Quantity Calculation:
   Total Quantity = Number of Items × Package Size
   Stock Value:
   Stock Value = Total Quantity × Actual Price
   Stock Update:
   New Quantity = (Current Items ± Change) × Package Size
   Stock Validation:
   Prevents negative stock levels and checks availability.

3. **Sales Processing Algorithms**
   Total Amount:
   Total = Quantity × Unit Price
   Discount Application:
   Final Amount = Total Amount − Discount
   Payment Status:
   Tracks sales as Pending, Partially Paid, or Paid.
   Stock Reduction:
   Decreases stock quantities after successful sale.

4. **Credit Management Algorithms**
   Credit Status:
   Manages states like Pending, Partially Paid, Paid, Archived.
   Payment Processing:
   Remaining Amount = Credit Amount − Payment Amount
   Interest Tracking:
   Tracks interest accrued on pending balances.
   Balance Updates:
   Updates remaining credit after each payment or interest application.

5. **Date and Time Algorithms**
   Duration Calculation:
   Days = (EndDate − StartDate) / (1000 × 60 × 60 × 24)
   Month Difference:
   Months = (Year2 − Year1) × 12 + (Month2 − Month1)
   Interest Period:
   Determines applicable interest period between two dates.
   Payment Scheduling:
   Manages scheduled due dates for customer payments.

6. **Data Validation Algorithms**
   Input Validation:
   Ensures required fields are filled and data types are valid.
   Business Rule Validation:
   Verifies input against logical business rules.
   Stock Availability:
   Ensures sufficient stock is available before processing sales.
   Credit Limit Validation:
   Prevents assigning credit beyond customer's allowed limit.

7. **Search and Filter Algorithms**
   Date Range Filtering:
      Filters records within selected start and end dates.
   Status Filtering:
      Filters by credit or sales status.
   Customer Filtering:
      Filters by customer name or contact info.
   Product Filtering:
      Filters inventory by category, product name, or availability.

8. **Report Generation Algorithms**
   Sales Summary:
      Aggregates and displays daily/monthly total sales and revenue.
   Credit Summary:
      Tracks total credits given and amounts repaid.
   Interest Summary:
      Summarizes interest accrued across all customers.
   Stock Summary:
      Provides report of current stock levels and low stock alerts.

Chapter 4

DESIGN

4.1 Modules Introduction

The Inventory and Business Management System for Agricultural Supplies is a comprehensive software solution designed to streamline and digitize the operations of agricultural input retailers. Traditional methods of managing inventory, handling credit sales, and maintaining customer and supplier records often lead to inefficiencies, data loss, and poor customer service. To address these challenges, this system offers a centralized platform built using modern web technologies including HTML, CSS, JavaScript, Node.js, Express, MySQL, bcrypt, and file upload mechanisms. Development and deployment tools include VS Code, MySQL Workbench, and cloud platforms like Render.

This system is modular and scalable, covering all core aspects of shop operations—from secure login for shop owners to advanced credit tracking and sales reporting. Each module is tailored to the specific needs of agricultural retailers, ensuring usability, real-time performance, and data accuracy.

This document outlines each module in detail, explaining its functionality, purpose, and key features to offer a complete understanding of the system architecture and workflow.

1. User Authentication Module

- Secure access to the system ensuring only authorized shop owners can manage data.
- User registration with email, password (hashed using bcrypt)
- Secure login/logout system
- Password confirmation validation
- Access control using session/token-based authentication
- CORS-enabled backend to restrict access from unauthorized sources

2. Inventory Management Module

- Manage stock details, track inventory changes, and trigger low stock alerts.
- Add/edit/delete product details (name, category, price, quantity, expiry)
- File upload for product images
- Automatic low stock alert system
- Real-time stock quantity update after sales/purchases
- Categorized inventory: Fertilizers, Seeds, Pesticides (with unit-based tracking)

3. Supplier & Order Management Module

- Maintain supplier information and track purchase orders efficiently.
- Record supplier name, contact, and purchase history
- Generate and log purchase orders
- View historical data of orders for reference or audits
- Helps in planning procurement based on stock levels

4. Sales & Billing Module

- Manage customer purchases, generate bills, and calculate revenue.
- Add sales entries with quantity, unit price, and discounts
- Generate itemized bills with total amount and final payable
- Automatically update stock on successful sale
- Display profit and sales report per transaction

5. Credit Management Module

- Track credit purchases and automate interest calculations.
- Record credit-based transactions with due amount
- Apply interest only on remaining balance
- Handle partial payments and update credit logs
- Generate outstanding reports and customer credit scores
- Customizable interest rates per customer

6. Customer Profile Management Module

- Maintain detailed records of each customer for personalized service.
- Store personal details and transaction history
- View credit details, payment history, and due dates
- Add custom notes (preferred products, pricing deals, etc.)
- Search and filter customers by name, payment status

7. Reporting & Analytics Module

- Provide meaningful insights through summarized reports.
- Credit tracking reports
- Stock level and expiry date alerts
- Exportable reports in PDF or Excel formats

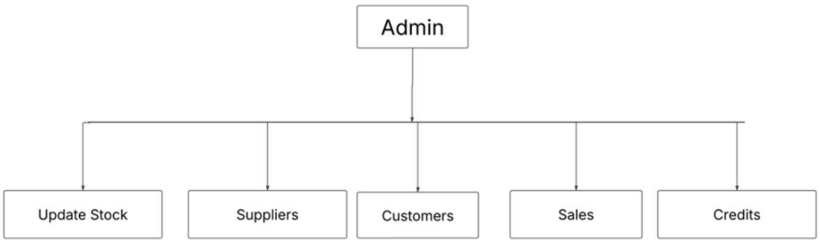
4.2 DFD / ER / UML Diagrams

4.2.1 Data Flow Diagram (DFD)

Level 0 DFD :

- Represents the entire system as a single process
- Shows interactions between the system and external entities
- Manages overall operations related to inventory, sales, and supplier management
- Handles billing, credit transactions, and customer interactions

Figure 7: Level 0 DFD



External Entities:

Admin:

- Manages inventory and stock updates
- Handles billing processes, including sales and credit transactions
- Generates reports on sales, inventory levels, and financials

Customer

- Purchases products using cash or credit
- Receives invoices and payment confirmations
- Interacts with the shop for product selection and inquiries

Supplier

- Supplies goods required for stock replenishment
- Receives purchase orders from the shop
- Updates records based on deliveries and payments

Sales

- Manage customer purchases, generate bills, and calculate revenue.
- Add sales entries with quantity, unit price, and discounts
- Automatically update stock on successful sale
- Display profit and sales report per transaction

Credits

- Record credit-based transactions with due amount
- Handle partial payments and update credit logs
- Generate outstanding reports
- Customizable interest rates per customer

Data Flows:

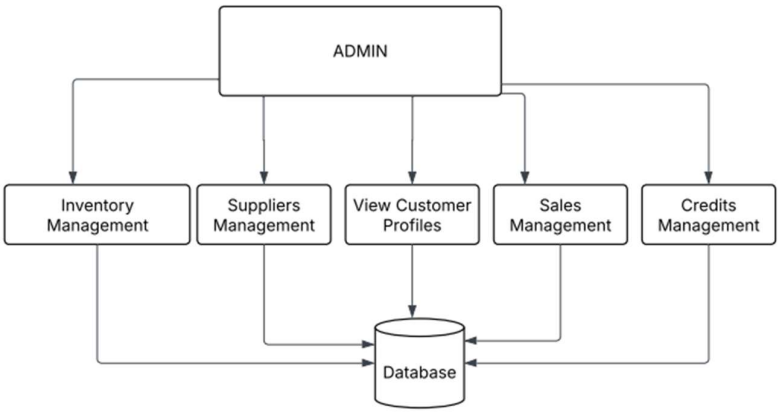
- **Shop Owner** → Sends/receives product data, transaction records, and inventory updates
- **Customer** → Sends purchase details, receives invoices and credit balance updates

- **Supplier** → Sends stock, receives purchase orders, and confirms deliveries

Level 1 DFD

- Expands the context-level diagram into specific modules
- Displays detailed data stores and how processes interact with them

Figure 7: Level 1 DFD



Processes:
User Authentication

- Input: Username, Password
- Output: Access granted/denied
- Data Store: Users
- Purpose: Ensures secure access control for shop management

Inventory Management

- Input: Product info including name, category, price, expiry, and quantity
- Output: Inventory updates reflecting stock additions and reductions
- Data Store: Stock
- Additional Functionality: Generates low stock alerts for timely restocking

Sales & Billing

- Input: Selected products, purchase quantity, applied discounts, and payment mode
- Output: Generates final bill, updates payment status, modifies stock levels accordingly
- Data Store: Sales,
- Additional Features: Supports credit purchases and cash transactions

Supplier & Order Management

- Input: New supplier details, purchase order specifics
- Output: Generates purchase order documents, updates supplier records
- Data Store: Suppliers and Orders
- Extra Functions: Tracks supplier deliveries and payment history

Credit Sales Management

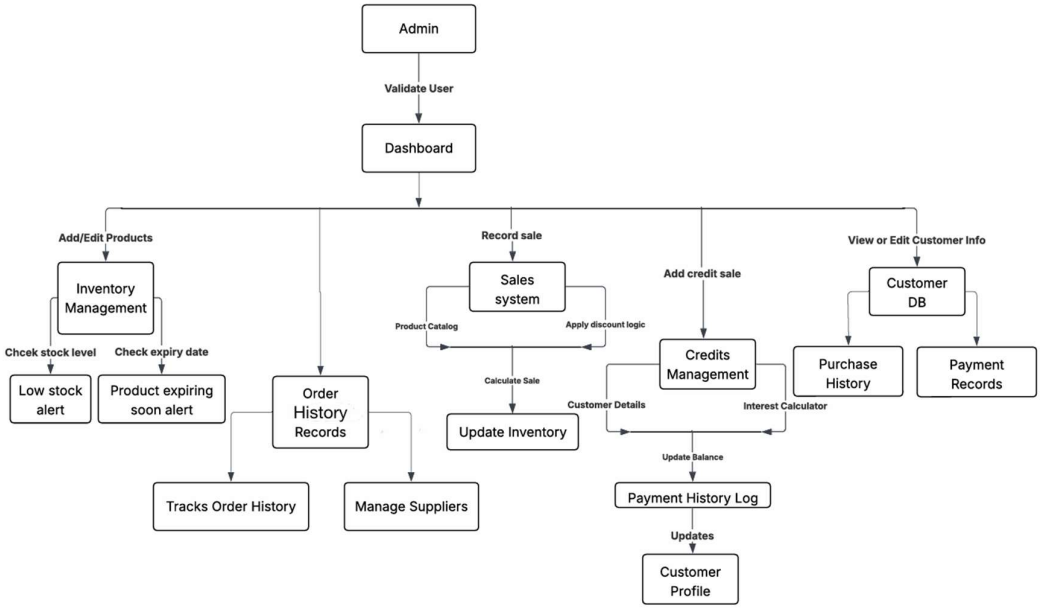
- Input: Customer ID, product selection, and credit amount details
- Output: Updates credit balance, applies interest rates if applicable, logs payment records
- Data Store: Credits
- Further Actions: Monitors overdue payments and sends reminders

Customer Management

- Input: Customer profile information, purchase history, and credit status
- Output: Maintains purchase logs, payment tracking, and credit scoring
- Data Store: Customers
- Extended Features: Helps analyze purchasing behavior and manage loyalty programs

Level 2 DFD

Figure 8: Level 2 DFD



1. User Authentication & Access Management

- Register Shop Owner:
    - Captures shop owner details (name, email, phone, password).

    - Validates uniqueness (e.g., no duplicate emails/phones).
- Login Process:
    - Authenticates credentials against stored password hashes.
    - Generates session tokens for authorized access.
- Access Control:
    - Restricts unauthorized actions (e.g., only owners can modify inventory).

Data Stores:

- **users** Table:
    - Stores: shop_name, email, phone, password_hash, role.
    - Used for: Login validation, session management, and multi-user segregation.

2. Inventory Management

- Add/Update/Delete Product:
    - Manages product details (name, category, cost/selling price, expiry).
- Track Low Stock:
    - Triggers alerts when stock falls below a threshold.
- Calculate Stock Value:
    - Sums total inventory value (quantity × cost price).

Data Stores:

- **stock** Table:
    - Stores: product_id, name, category, quantity, cost_price, selling_price, expiry_date, supplier_id.
    - Linked to users for shop-specific inventory.
- **orders** Table:
    - Updates stock when orders are marked "Delivered."

3. Supplier & Order Management

- Add Supplier:
    - Records supplier details (name, contact, items supplied).
- Create Purchase Order:
    - Generates orders for suppliers (product, quantity, expected delivery date).
- View Order History:
    - Filters orders by status (Pending/Delivered/Cancelled).

Data Stores:

- **orders** Table:
  - Stores: order_id, product_id, quantity, supplier_id, status, date_ordered.
- **stock** Table:
  - Updated upon order delivery (increases product quantity).

4. Sales & Billing System

- Sell Product (Cash/Card):
  - Records sales, updates stock, and generates receipts.
- Generate Daily Sales:
  - Aggregates sales by date, payment mode, and discounts.
- Apply Bulk Discounts:
  - Applies discounts for bulk purchases (configurable rules).

Data Stores:

- **sales** Table:
  - Stores: sale_id, customer_id, total_amount, discount, payment_mode, date.
- **sale_items** Table:
  - Stores: sale_id, product_id, quantity, unit_price.
- **stock** Table:
  - Deducts sold quantities in real-time.

5. Credit Sales Management

- Record Credit Sale:
  - Tracks products sold on credit (customer, due date, interest rate).
- Calculate Interest:
  - Applies interest on overdue balances (configurable rates).
- Accept Partial Payment:
  - Logs partial payments against credit dues.
- Generate Credit Report:
  - Shows pending credits, interest accrued, and payment history.

Data Stores:

- **credit_sales** Table:
  - Stores: credit_sale_id, customer_id, total_amount, due_date, balance.

- **credit_payments** Table:
    - Logs: payment_id, credit_sale_id, amount_paid, payment_date.
- **credit_interest_calculations** Table:
    - Tracks: calculation_id, credit_sale_id, interest_rate, interest_amount.

6. Customer Profile Management

- Add Customer:
    - Captures customer details (name, phone, address).
- View Purchase History:
    - Displays past sales/credit transactions.
- Manage Credit Info:
    - Tracks credit limits, repayment behavior, and blacklisting.

Data Stores:

- **customers** Table:
    - Stores: customer_id, name, phone, address, total_purchases, credit_limit.
- Linked to sales, credit_sales, and credit_payments for comprehensive history.

4.3 Module Design and Organisation

The project is divided into logically structured modules to improve maintainability, scalability, and clarity. Each module handles a core business function and interacts with others through well-defined interfaces (APIs or function calls).

User Authentication & Access Control Module

Handles user registration, login, session, and access control.

Sub-components:

- RegistrationController: Collects shop owner details and stores in users table.
- LoginService: Authenticates using hashed passwords (bcrypt).
- SessionManager: Handles login state and session expiry.
- Middleware: Ensures authorized access to routes.

Data Stores:

- users

Inventory Management Module

Manages product stock, updates, and low stock alerts.

Sub-components:

- ProductManager: Adds, updates, and deletes products.
- StockTracker: Monitors product quantities.
- AlertGenerator: Generates low stock notifications.
- ImageHandler: Uploads & manages product images.

Data Stores:

- stock

Supplier & Order Management Module

Maintains supplier records and tracks product orders.

Sub-components:

- SupplierController: Adds/updates supplier info.
- OrderService: Creates/manages product orders.
- OrderStatusTracker: Tracks pending, approved, and delivered orders.

Data Stores:

- orders, suppliers , stock

Sales & Billing Module

Manages product sales, billing, and discounts.

Sub-components:

- SalesController: Handles normal and credit-based sales.
- BillingEngine: Calculates totals, applies discounts.
- ReportGenerator: Generates daily billing summaries.

Data Stores:

- sales, sale_items, customers

Credit Sales & Interest Module

Manages credit transactions, interest, and repayments.

Sub-components:

- CreditSaleManager: Records new credit sales.

- InterestCalculator: Applies interest on outstanding amounts.
- PaymentTracker: Accepts and updates partial/full payments.
- CreditReportGenerator: Produces outstanding and interest reports.

Data Stores:

- credit_sales, credit_products, credit_interest_calculations, credit_payments

Customer Management Module

Maintains detailed customer profiles and credit info.

Sub-components:

- CustomerService: Adds/updates customer details.
- HistoryViewer: Shows purchase and credit history.
- CreditStatusManager: Calculates credit score and credit limits.

Data Stores:

- customers, credit_sales, sales

Chapter 5

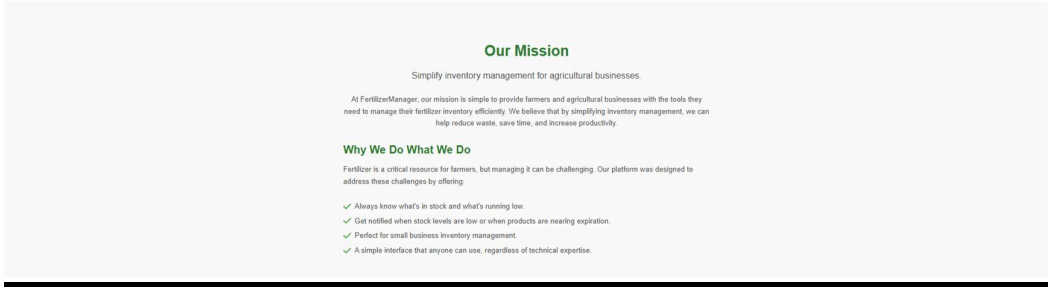
Output Screenshots and Description

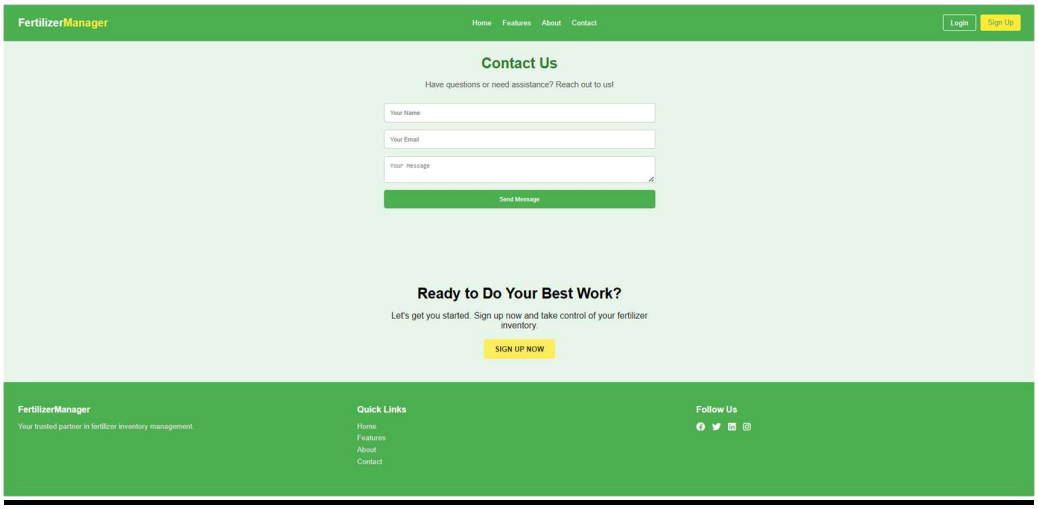
This chapter presents the key user interface screens of the Inventory and Business Management System for agricultural supplies. Each screenshot is accompanied by a description of its features and functionality, providing a clear understanding of the application flow.

5.1 Home page

Description:

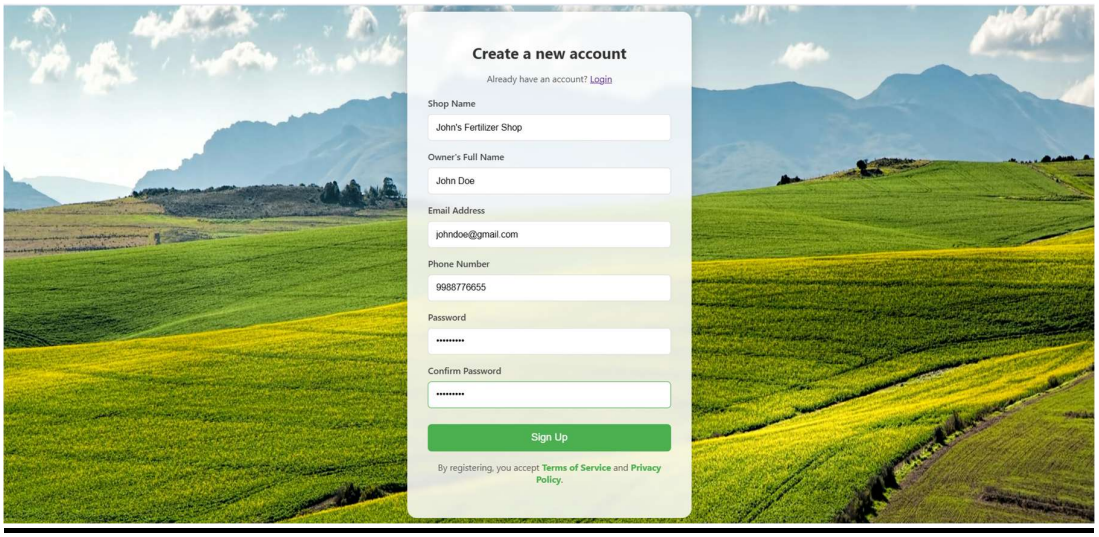
The Home Page of the Inventory and Business Management System for agricultural supplies acts as a central dashboard, offering users an overview of key business metrics and easy access to core functionalities. It includes a personalized welcome message, real-time summaries of customer data, stock levels, outstanding credits, and recent sales. Users can login/sign up to access this system.



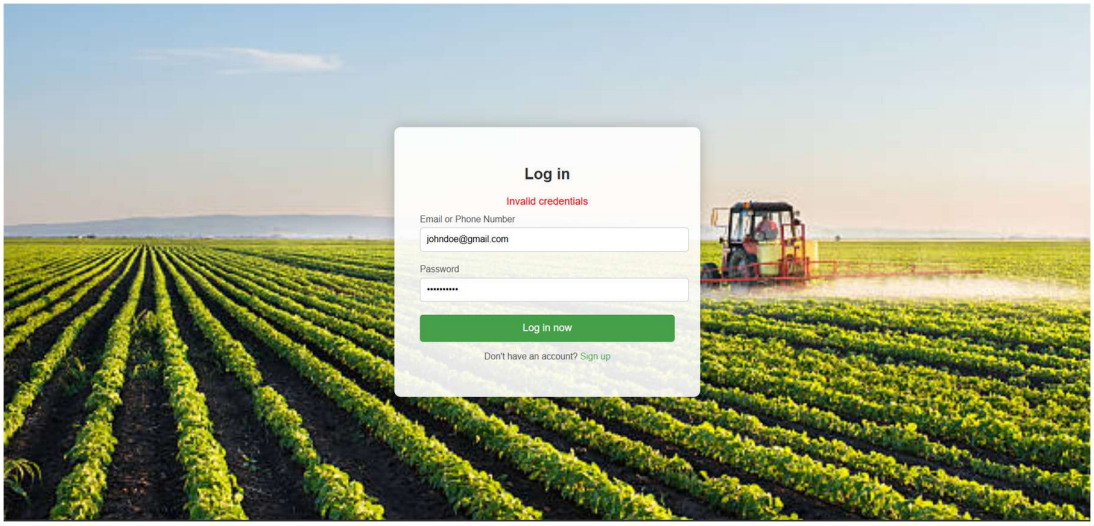


5.3 Login and Sign-Up Page

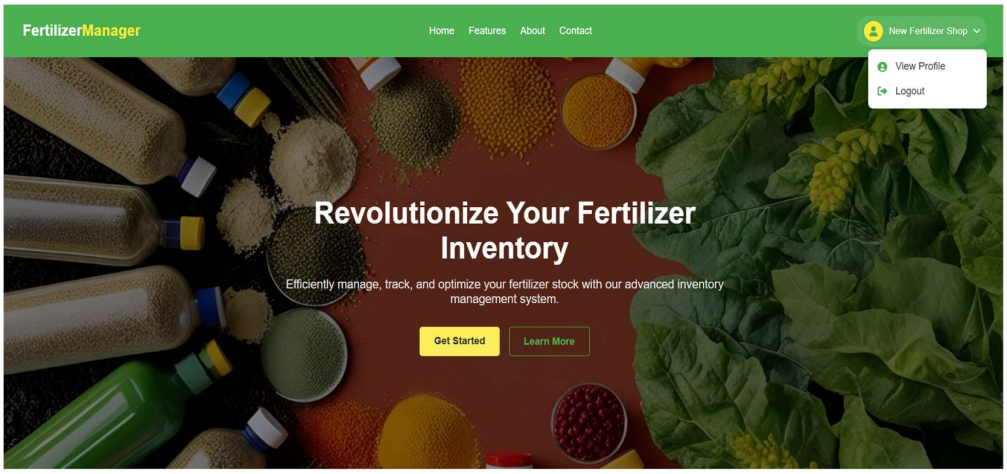
Sign-Up Section: New users can register by providing their name, email, password, and choosing their role (mentor or learner).It uses SQL Authentication for secure access.

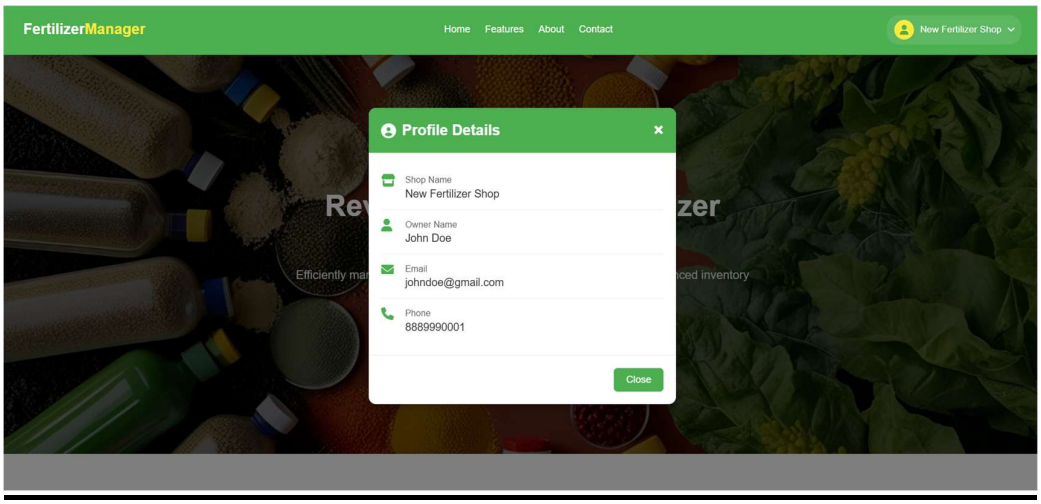


Login Section: Allows existing users to log in using their email and password securely.



The homepage shown to logged-in users includes and option to view profile. It also has the log out option to close the session of that user.

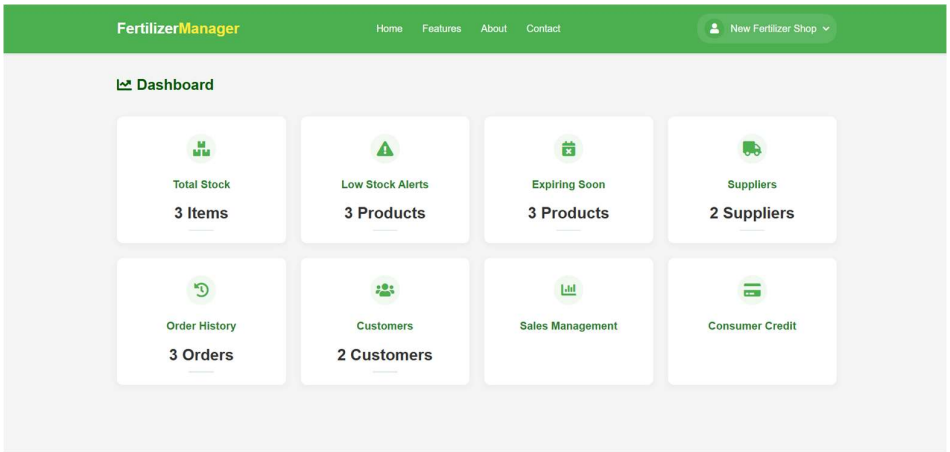




5.3 Admin page

Description:

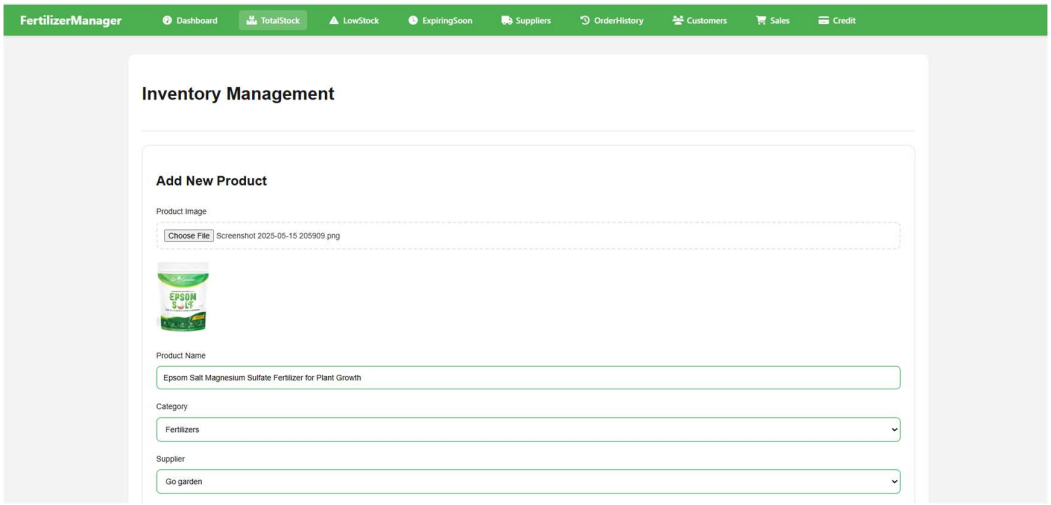
The dashboard page serves as the central hub after login, providing an overview of key business metrics and quick access to various modules. It displays summaries such as total order history, stock levels, and alerts for low stock or expiring products. The dashboard offers intuitive navigation to inventory, sales, customers, and reports, enabling shop owners to efficiently monitor and manage their operations from a single interface.

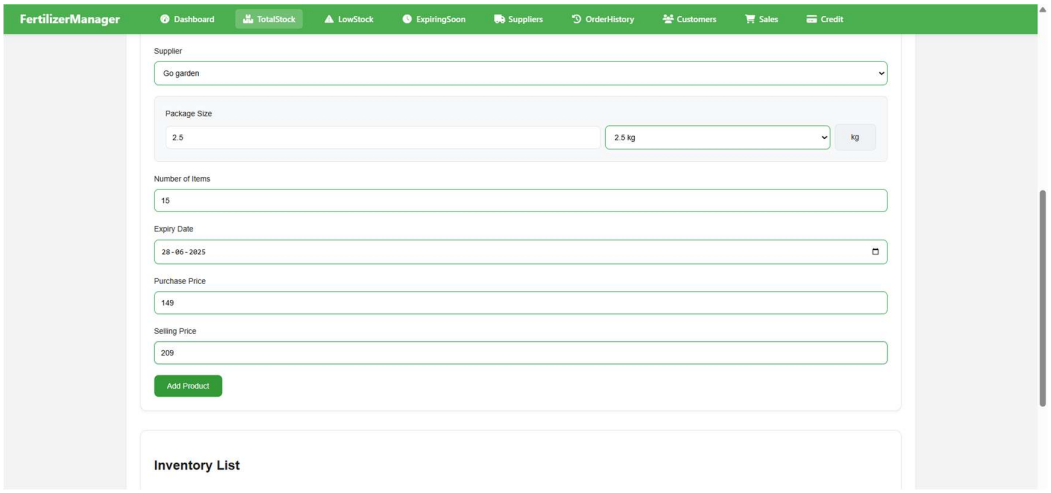


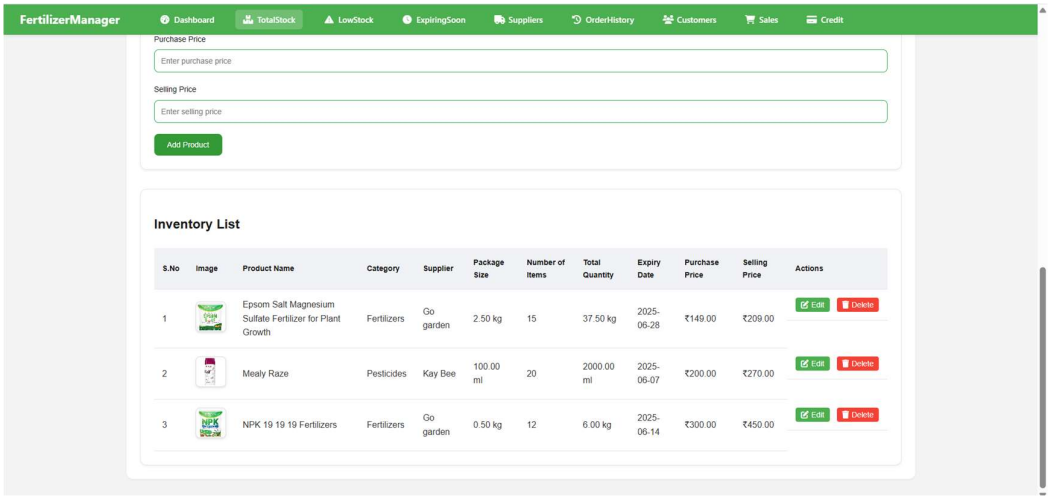
5.4 Inventory Page

Description:

The inventory page allows shop owners to manage their stock efficiently by adding, editing, or deleting product entries. When adding a product, the user enters details such as name, category, supplier, quantity, unit, prices, expiry date, and an image. Editing allows modifications to existing stock records, including quantity updates and image replacement. Deletion removes a product from the inventory after checking for any associated sales or credit transactions. The page ensures data validation to maintain accurate and up-to-date inventory records.



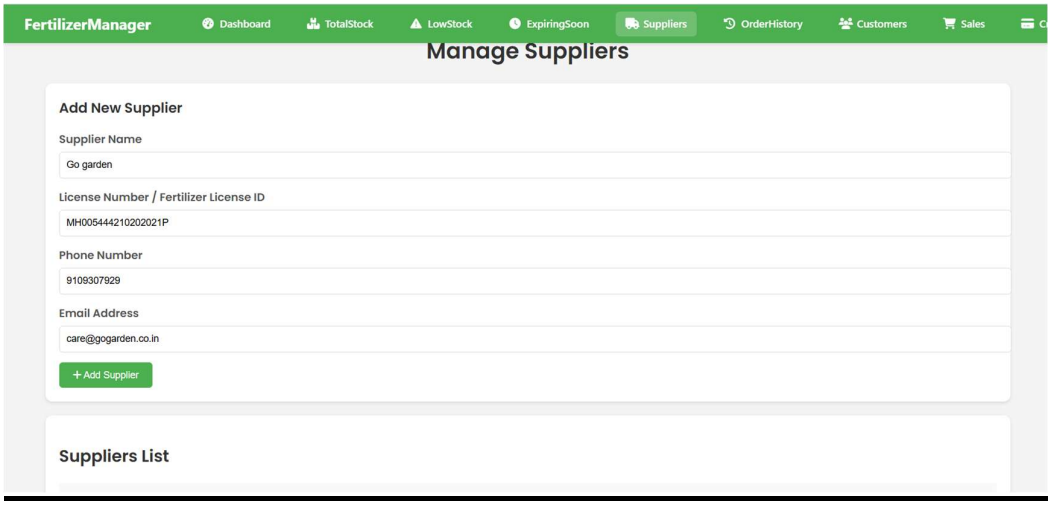


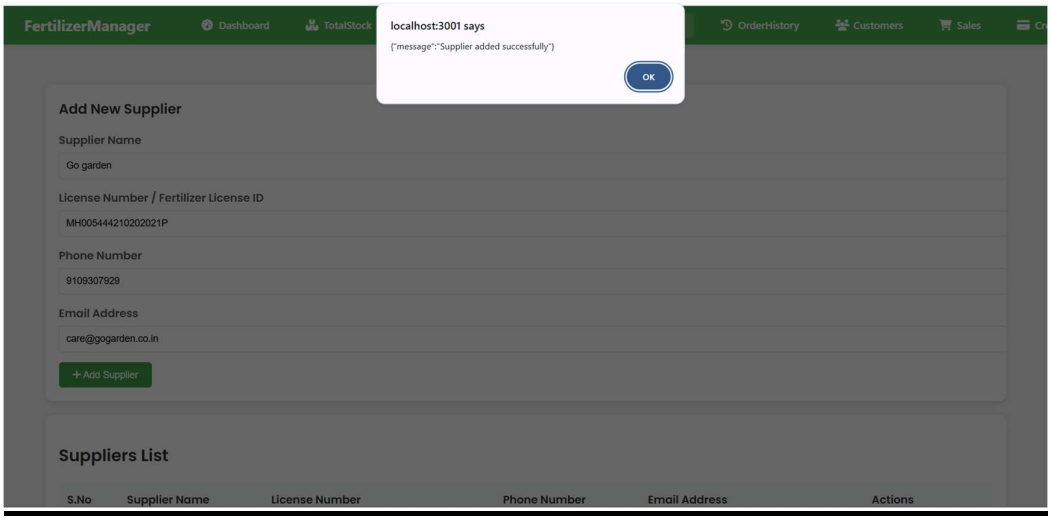


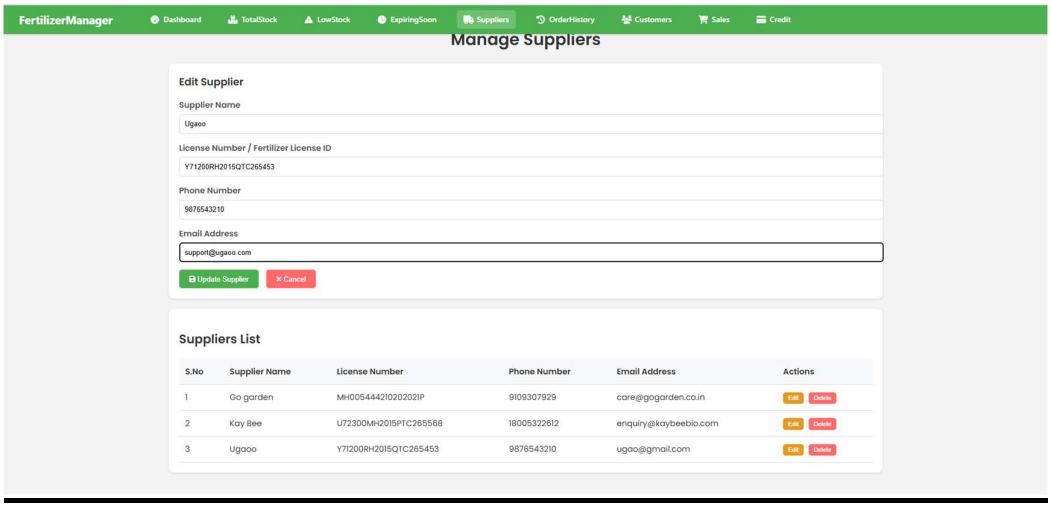
5.5 Suppliers Page

Description:

The suppliers page helps shop owners manage supplier information and track their order history. Users can add new suppliers by entering details such as name, contact information, and linked products. Existing supplier records can be edited to update contact details or association with products. Deletion is restricted if there are active purchase orders linked. This page ensures smooth coordination with suppliers and maintains an organized record for efficient stock procurement.



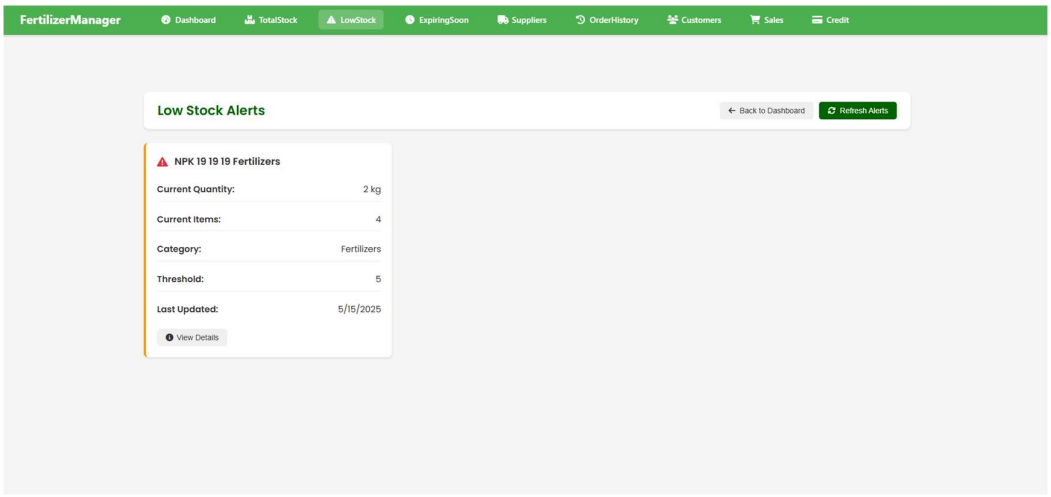




5.6 Low stock alerts

Description:

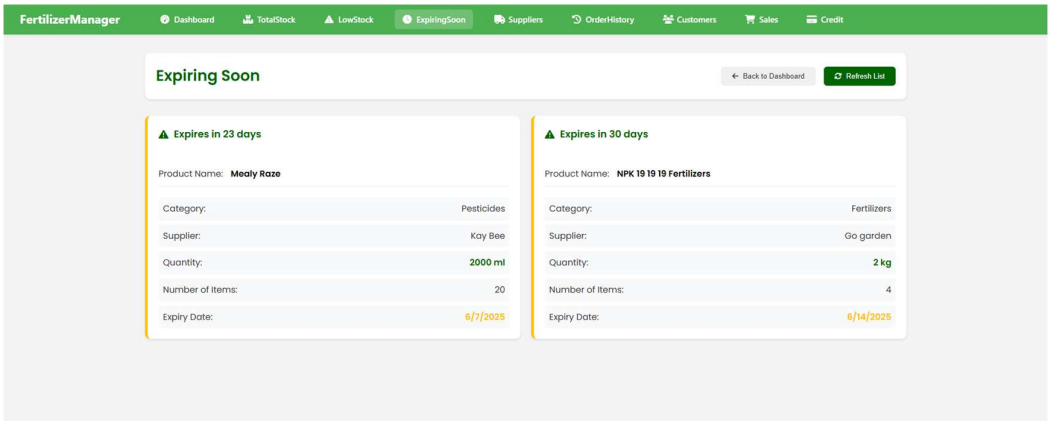
The low stock alerts page notifies the shop owner whenever product quantities fall below a predefined threshold. It displays a list of such items with details like product name, current quantity, minimum required level, and category. This page helps in proactive inventory management by allowing quick reordering to avoid stockouts. It ensures the shop remains well-stocked and ready to meet customer demand.



5.7 Expiring soon product alerts

Description:

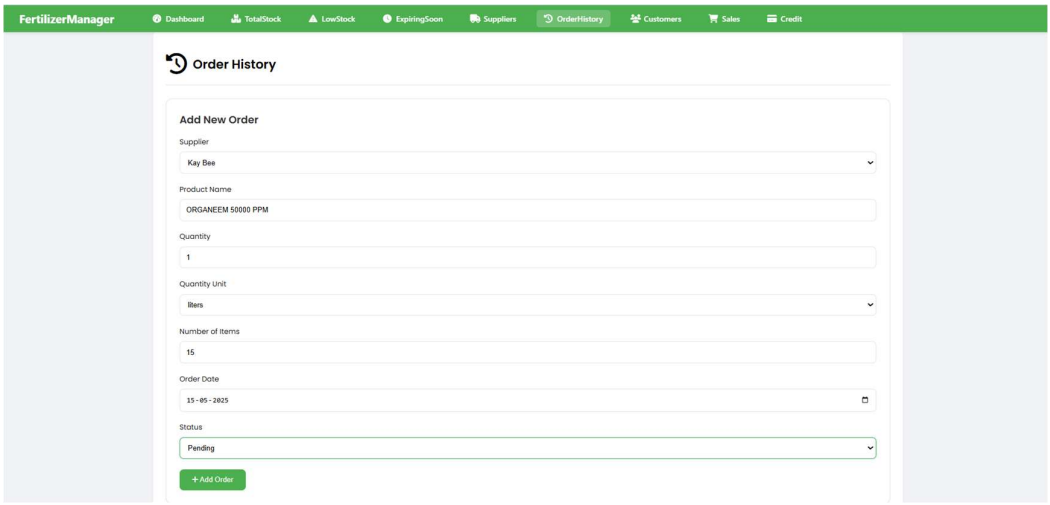
The expiring soon product alerts page highlights products that are nearing their expiry dates. It presents a list with details such as product name, expiry date, remaining shelf life, and quantity. This feature helps the shop owner manage inventory efficiently by prioritizing the sale or clearance of these items, reducing waste and ensuring that expired products are not sold to customers.

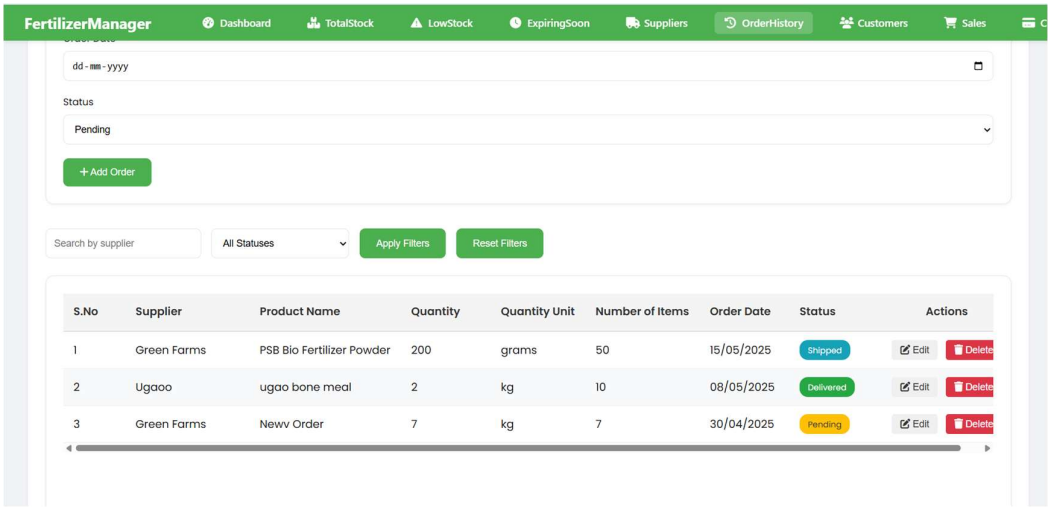


5.8 Order History Record

Description:

The order history records page provides a comprehensive view of all past purchase orders made to suppliers. It displays essential details such as supplier name, ordered products, quantities, order date, and current status (e.g., pending, delivered). Users can filter records based on supplier, date range, or status for quick access. This page helps track delivery timelines, verify stock updates, and maintain accountability in supplier transactions.

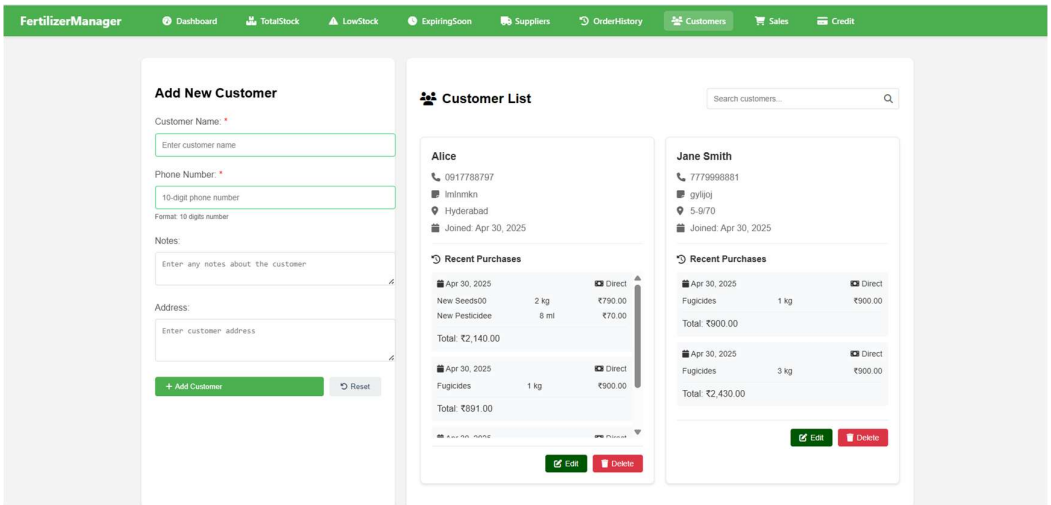


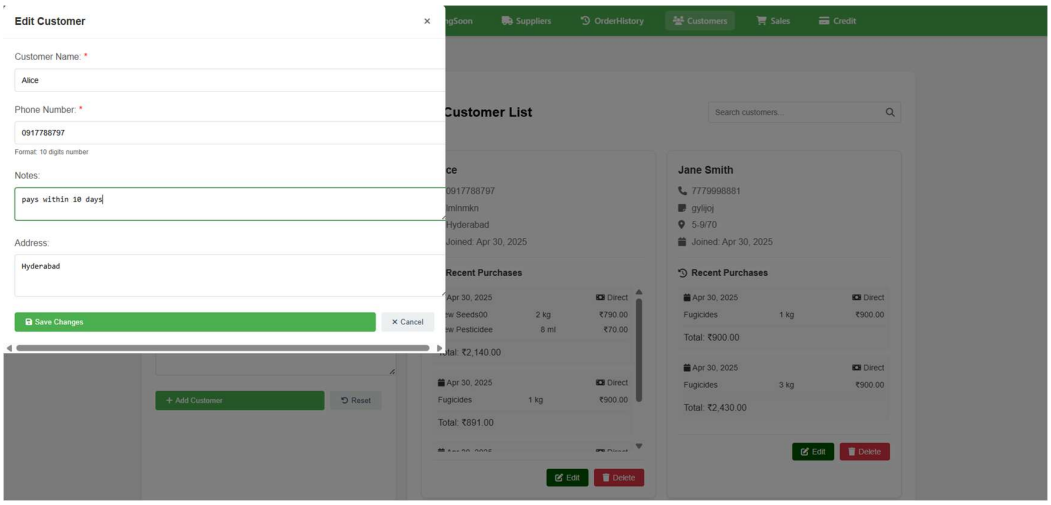


5.9 Customers

Description:

The customer page allows the shop owner to manage customer information efficiently. It supports adding, editing, and deleting customer details such as name, contact information, and address. Additionally, the page provides an option to add personalized notes for each customer, helping track preferences, credit history, or important remarks. This enhances customer relationship management by maintaining organized and up-to-date customer profiles.

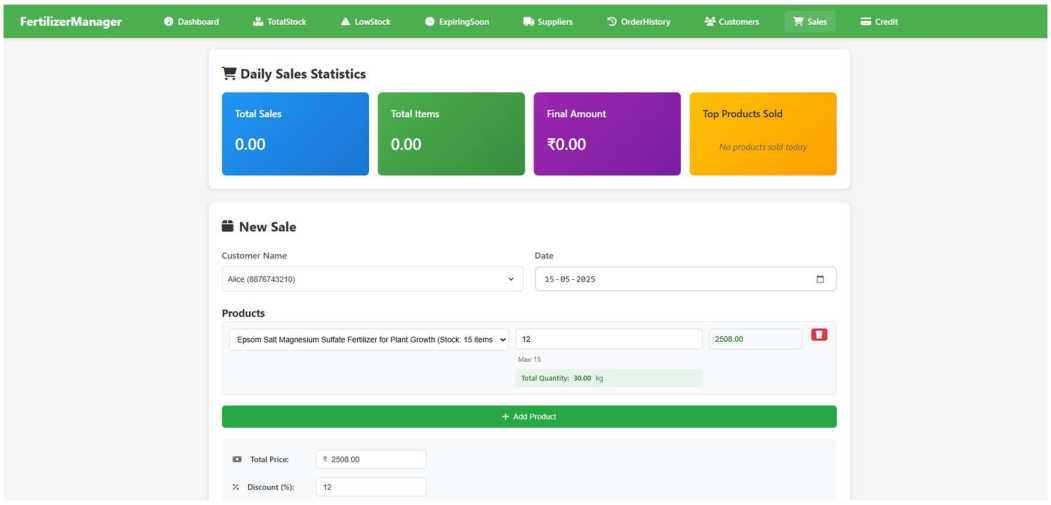


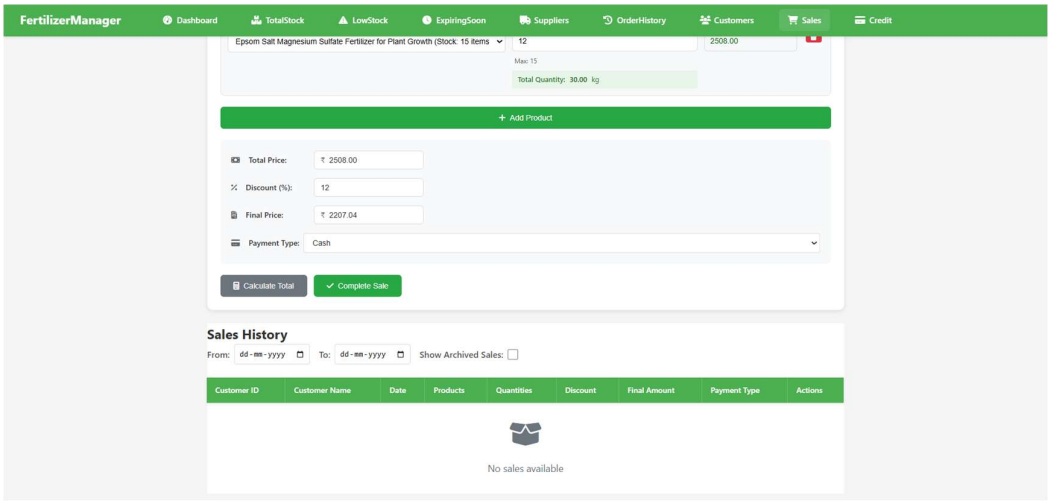


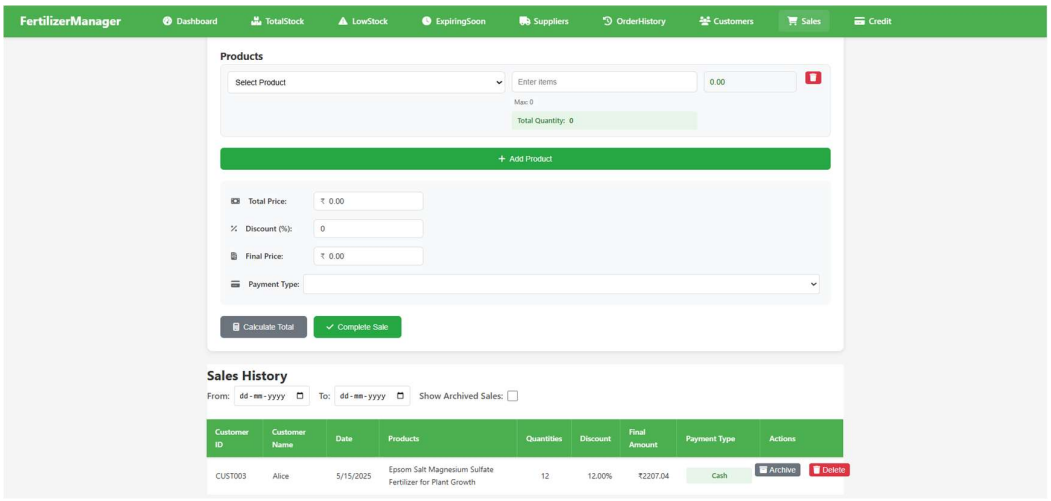
5.10 Sales

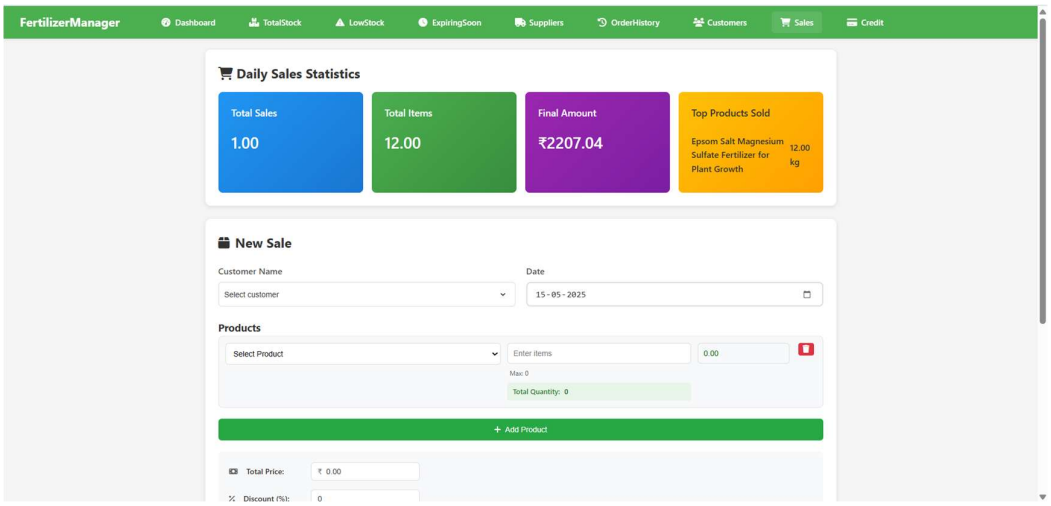
Description:

The sales page is designed to manage and record product sales, whether done through cash, card, or credit. It allows the user to select products, apply discounts, and calculate totals in real-time. Once a sale is completed, it updates the stock levels accordingly. The page also links sales to specific customers and supports partial or full payments for credit sales, ensuring accurate transaction tracking and smooth billing operations.




FertilizerManager
Dashboard
TotalStock
LowStock
ExpiringSoon
Suppliers
OrderHistory
Customers
Sales
Credit
Epsom Salt Magnesium Sulfate Fertilizer for Plant Growth (Stock: 15 items
12
2508.00
Max: 15
Total Quantity: 30.00 kg
+ Add Product
Total Price:
₹ 2508.00
Discount (%):
12
Final Price:
₹ 2207.04
Payment Type:
Cash
Calculate Total
Complete Sale
Sales History
From:
dd-mm-yyyy
To:
dd-mm-yyyy
Show Archived Sales:
Customer ID
Customer Name
Date
Products
Quantities
Discount
Final Amount
Payment Type
Actions
No sales available

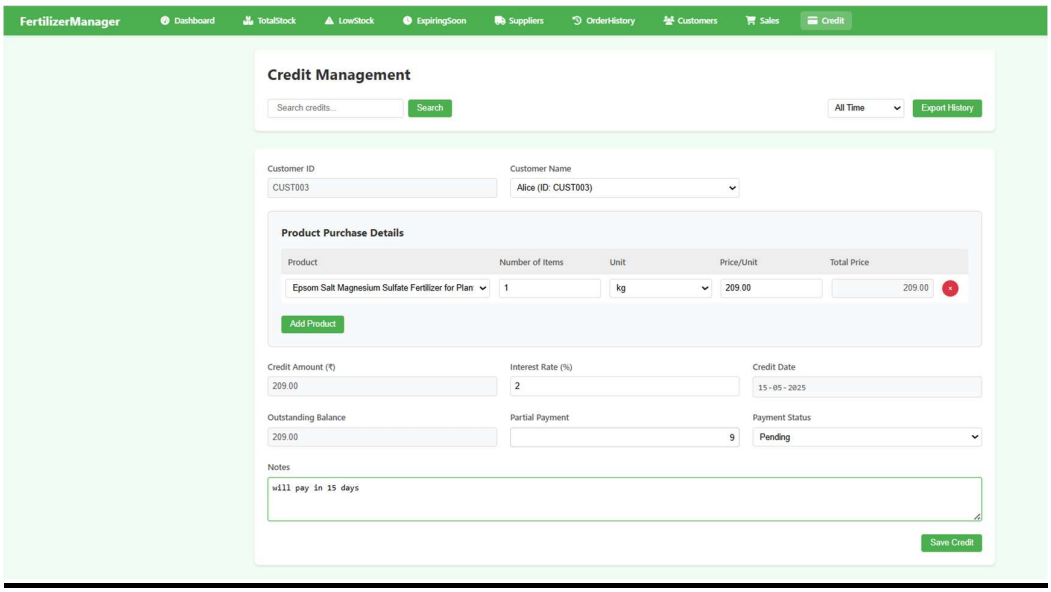

FertilizerManager
Dashboard
TotalStock
LowStock
ExpiringSoon
Suppliers
OrderHistory
Customers
Sales
Credit
Products
Select Product
Enter items
0.00
Max: 0
Total Quantity: 0
+ Add Product
Total Price:
₹ 0.00
Discount (%):
0
Final Price:
₹ 0.00
Payment Type:
Calculate Total
Complete Sale
Sales History
From:
dd-mm-yyyy
To:
dd-mm-yyyy
Show Archived Sales:
Customer ID
Customer Name
Date
Products
Quantities
Discount
Final Amount
Payment Type
Actions
CUST003
Alice
5/15/2025
Epsom Salt Magnesium Sulfate Fertilizer for Plant Growth
12
12.00%
₹2207.04
Cash
Archive
Delete

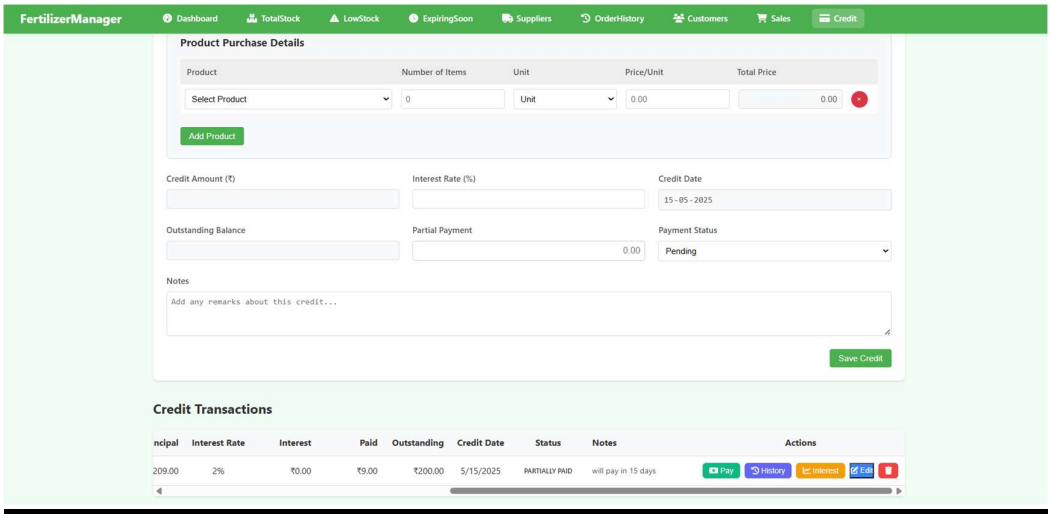

FertilizerManager
Dashboard
TotalStock
LowStock
ExpiringSoon
Suppliers
OrderHistory
Customers
Sales
Credit
Daily Sales Statistics
Total Sales
1.00
Total Items
12.00
Final Amount
₹2207.04
Top Products Sold
Epsom Salt Magnesium Sulfate Fertilizer for Plant Growth
12.00 kg
New Sale
Customer Name
Select customer
Date
15-05-2025
Products
Select Product
Enter items
0.00
Max: 0
Total Quantity: 0
+ Add Product
Total Price:
₹ 0.00
Discount (%):
0

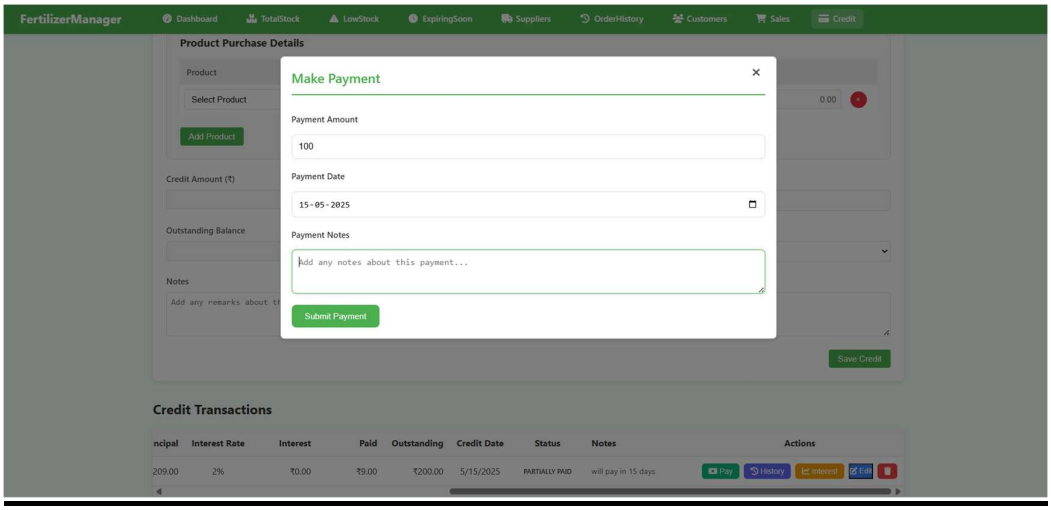
5.11 Credits

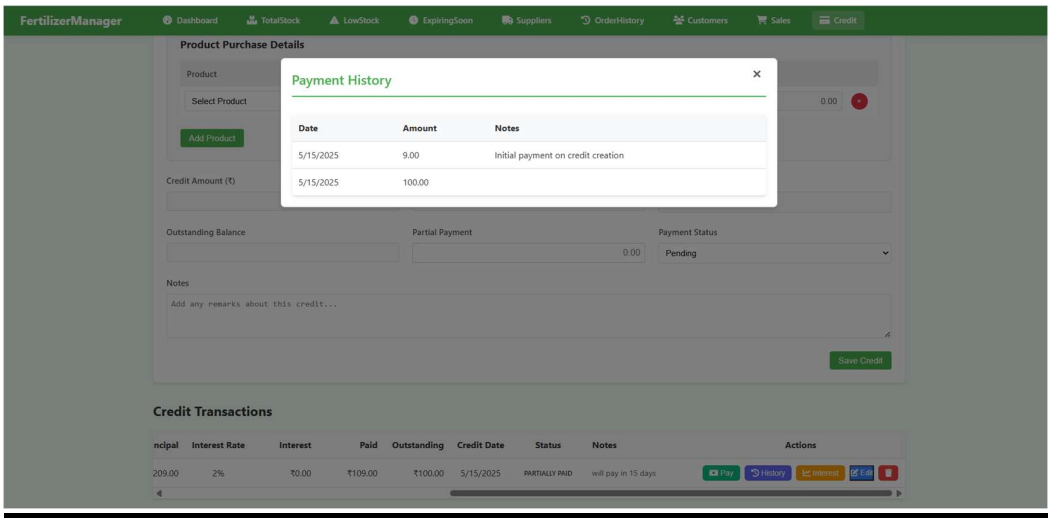
Description:

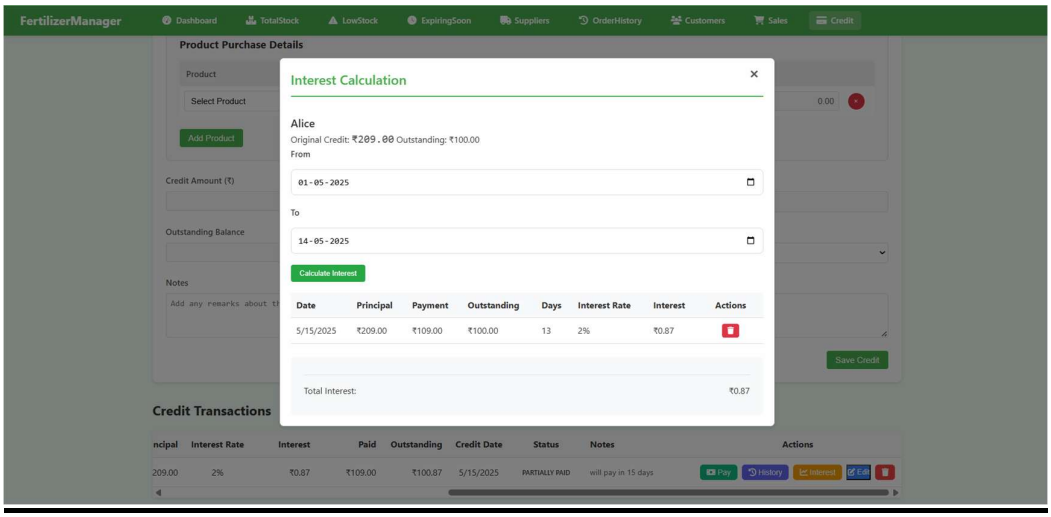
The credits page manages all credit-based transactions, allowing users to record new credit sales, track outstanding balances, and apply interest calculations. It provides features to update payment statuses, record partial payments, and maintain detailed payment histories. This page helps monitor customer credit limits, generate credit reports, and ensures accurate management of credit accounts for smooth financial operations.






FertilizerManager
Dashboard
TotalStock
LowStock
ExpiringSoon
Suppliers
OrderHistory
Customers
Sales
Credit
Product Purchase Details
Product
Select Product
Add Product
Credit Amount (₹)
Outstanding Balance
Notes
Add any remarks about th
Save Credit
Make Payment
Payment Amount
100
Payment Date
15-05-2025
Payment Notes
Add any notes about this payment...
Submit Payment
Credit Transactions
ncipal
Interest Rate
Interest
Paid
Outstanding
Credit Date
Status
Notes
Actions
209.00
2%
₹0.00
₹9.00
₹200.00
5/15/2025
PARTIALLY PAID
will pay in 15 days
Pay
History
Interest
Edit


FertilizerManager
Dashboard
TotalStock
LowStock
ExpiringSoon
Suppliers
OrderHistory
Customers
Sales
Credit
Product Purchase Details
Product
Select Product
Add Product
Credit Amount (₹)
Outstanding Balance
Partial Payment
Payment Status
0.00
Pending
Notes
Add any remarks about this credit...
Save Credit
Payment History
Date
Amount
Notes
5/15/2025
9.00
Initial payment on credit creation
5/15/2025
100.00
Credit Transactions
ncipal
Interest Rate
Interest
Paid
Outstanding
Credit Date
Status
Notes
Actions
209.00
2%
₹0.00
₹109.00
₹100.00
5/15/2025
PARTIALLY PAID
will pay in 15 days
Pay
History
Interest
Edit


FertilizerManager
Dashboard
TotalStock
LowStock
ExpiringSoon
Suppliers
OrderHistory
Customers
Sales
Credit
Product Purchase Details
Product
Select Product
Add Product
Credit Amount (₹)
Outstanding Balance
Notes
Add any remarks about th
Save Credit
Interest Calculation
Alice
Original Credit: ₹209.00 Outstanding: ₹100.00
From
01-05-2025
To
14-05-2025
Calculate Interest
Date
Principal
Payment
Outstanding
Days
Interest Rate
Interest
Actions
5/15/2025
₹209.00
₹109.00
₹100.00
13
2%
₹0.87
Total Interest:
₹0.87
Credit Transactions
ncipal
Interest Rate
Interest
Paid
Outstanding
Credit Date
Status
Notes
Actions
209.00
2%
₹0.87
₹109.00
₹100.87
5/15/2025
PARTIALLY PAID
will pay in 15 days
Pay
History
Interest
Edit

Chapter 6

Sample Code

This chapter highlights important sections of the source code used in the development of the Inventory and Business Management System for Agriculture Supplies platform. The following code snippets demonstrate how core functionalities such as home page, user authentication, data storage, and UI rendering are implemented.

The homepage code serves as the welcoming interface for users visiting the application. It typically includes a navigation bar, a banner or hero section with introductory text about the shop or system, and buttons for login or registration. The layout is structured using HTML CSS with responsiveness and custom styling. The page may also include a brief description of the platform's features and a footer with contact or support information. JavaScript is used to enhance interactivity, such as smooth scrolling or animations.

homepage.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Fertilizer Inventory Management</title>
  <style>
body {
    font-family: 'Arial', sans-serif;
    margin: 0;
    padding: 0;
    background-color: #f4f4f9;
    color: #333;
  }
  header {
    background-color: #4CAF50;
    color: white;
    padding: 1rem 2rem;
    position: sticky;
    top: 0;
    z-index: 1000;
  }

  nav {
    display: flex;
    justify-content: space-between;
    align-items: center;
  }
```

```
.logo {
  font-size: 1.5rem;
  font-weight: bold;
}

.logo span {
  color: #ffeb3b;
}

.nav-links {
  list-style: none;
  display: flex;
  gap: 1.5rem;
}

.nav-links a {
  color: white;
  text-decoration: none;
  transition: color 0.3s ease;
}

.nav-links a:hover {
  color: #ffeb3b;
}

.auth-buttons {
  display: flex;
  gap: 10px;
}

.login-btn, .signup-btn {
  padding: 8px 16px;
  border-radius: 4px;
  text-decoration: none;
  font-weight: 500;
  transition: all 0.3s ease;
}

.login-btn {
  background-color: transparent;
  border: 1px solid white;
  color: white;
}

.login-btn:hover {
  background-color: rgba(255, 255, 255, 0.1);
}

.signup-btn {
  background-color: #ffeb3b;
```

```
    color: #4CAF50;
    border: none;
  }

  .signup-btn:hover {
    background-color: #fdd835;
  }

/* Hero Section */
.hero {
    position: relative;
    height: 80vh; /* Adjust height as needed */
    display: flex;
    align-items: center;
    justify-content: center;
    text-align: center;
    color: white;
    background-image: url('../image/bg1.png'); /* Replace with your image */
    background-size: cover;
    background-position: center;
    background-repeat: no-repeat;
    animation: fadeIn 2s ease-in-out;
  }

  .hero::before {
    content: '';
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    background-color: rgba(0, 0, 0, 0.5); /* Dark overlay for better text visibility */
  }

  .hero-content {
    position: relative;
    z-index: 1;
    max-width: 800px;
    padding: 0 1rem;
  }

  .hero-content h1 {
    font-size: 3rem;
    margin-bottom: 1rem;
    color: white;
  }

  .hero-content p {
    font-size: 1.2rem;
    margin-bottom: 2rem;
```

```
  color: #e0e0e0;
}

.cta-buttons {
  display: flex;
  justify-content: center;
  gap: 1rem;
}

.cta-button {
  background-color: #4CAF50;
  color: white;
  padding: 0.8rem 1.5rem;
  border: none;
  border-radius: 5px;
  text-decoration: none;
  transition: background-color 0.3s ease, transform 0.3s ease;
}

.cta-button.secondary {
  background-color: transparent;
  border: 1px solid #4CAF50;
  color: #4CAF50;
}

.cta-button:hover {
  background-color: #45a049;
  transform: translateY(-5px);
}

.cta-button.secondary:hover {
  background-color: #4CAF50;
  color: white;
}

/* Responsive Design for Hero Section */
@media (max-width: 768px) {
  .hero {
    height: 60vh; /* Adjust height for smaller screens */
  }

  .hero-content h1 {
    font-size: 2rem;
  }

  .hero-content p {
    font-size: 1rem;
  }

  .cta-buttons {
```

```
    flex-direction: column;
    align-items: center;
  }
}
/* Responsive Design for Hero Section */
@media (max-width: 768px) {
  .hero {
    flex-direction: column;
    text-align: center;
    padding: 2rem 1rem;
  }

  .hero-content {
    max-width: 100%;
    margin-bottom: 2rem;
  }

  .hero-image {
    justify-content: center;
  }

  .cta-buttons {
    justify-content: center;
  }
}

/* Features Section */
.features {
  padding: 4rem 2rem;
  background-color: white;
  text-align: center;
}

.features h2 {
  font-size: 2rem;
  margin-bottom: 2rem;
  color: #2e7d32;
}

.feature-grid {
  display: grid;
  grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));
  gap: 2rem;
}

.feature-card {
  padding: 2rem;
  border-radius: 10px;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
  transition: transform 0.3s ease, box-shadow 0.3s ease;
```

```
  }

  .feature-card:hover {
    transform: translateY(-10px);
    box-shadow: 0 8px 16px rgba(0, 0, 0, 0.2);
  }

  .feature-card i {
    font-size: 2rem;
    color: #4CAF50;
    margin-bottom: 1rem;
  }

  .feature-card h3 {
    font-size: 1.5rem;
    color: #2e7d32;
    margin-bottom: 1rem;
  }

  .feature-card p {
    color: #777;
  }

/* About Section */
 .about {
    padding: 4rem 2rem;
    background-color: #f9f9f9;
    text-align: center;
  }

  .about-content {
    max-width: 800px;
    margin: 0 auto;
  }

  .about-content h2 {
    font-size: 2rem;
    margin-bottom: 0.5rem;
    color: #2e7d32;
  }

  .about-content .subtitle {
    font-size: 1.2rem;
    color: #555;
    margin-bottom: 1.5rem;
  }

  .about-content p {
    font-size: 1rem;
    color: #555;
```

```
  margin-bottom: 2rem;
  line-height: 1.6;
}
.mission-statement {
  text-align: left;
  margin-bottom: 2rem;
}

.mission-statement h3 {
  font-size: 1.5rem;
  color: #2e7d32;
  margin-bottom: 1rem;
}

.mission-statement ul {
  list-style: none;
  padding: 0;
}

.mission-statement ul li {
  font-size: 1rem;
  color: #555;
  margin-bottom: 0.8rem;
  display: flex;
  align-items: center;
  gap: 0.5rem;
}

.mission-statement ul li i {
  color: #4CAF50;
  font-size: 1.2rem;
}

/* Commitment Section */
.commitment {
  text-align: left;
}

.commitment h3 {
  font-size: 1.5rem;
  color: #2e7d32;
  margin-bottom: 1rem;
}

.commitment p {
  font-size: 1rem;
  color: #555;
  line-height: 1.6;
}
```

```
@media (max-width: 768px) {
  .about {
    padding: 2rem 1rem;
  }

  .mission-statement ul li {
    font-size: 0.9rem;
  }
}

/* Contact Section */
.contact {
  padding: 4rem 2rem;
  background-color: #e8f5e9;
  text-align: center;
}

.contact h2 {
  font-size: 2rem;
  margin-bottom: 1rem;
  color: #2e7d32;
}

.contact p {
  font-size: 1.1rem;
  margin-bottom: 2rem;
  color: #555;
}

.contact-form {
  max-width: 600px;
  margin: 0 auto;
  display: flex;
  flex-direction: column;
  gap: 1rem;
}

.contact-form input, .contact-form textarea {
  padding: 0.8rem;
  border: 1px solid #ccc;
  border-radius: 5px;
  transition: border-color 0.3s ease;
}

.contact-form input:focus, .contact-form textarea:focus {
  border-color: #4CAF50;
  outline: none;
}

.contact-form button {
```

```
  padding: 0.8rem;
  background-color: #4CAF50;
  color: white;
  border: none;
  border-radius: 5px;
  cursor: pointer;
  transition: background-color 0.3s ease, transform 0.3s ease;
}

.contact-form button:hover {
  background-color: #45a049;
  transform: translateY(-3px);
}

.form-message {
  margin-top: 1rem;
  color: #4CAF50;
  font-weight: bold;
}
.cta-section {
  padding: 4rem 2rem;
  background-color: #e8f5e9;
  color: rgb(0, 0, 0);
  text-align: center;
}

.cta-content {
  max-width: 600px;
  margin: 0 auto;
}

.cta-content h2 {
  font-size: 2rem;
  margin-bottom: 1rem;
}

.cta-content p {
  font-size: 1.2rem;
  margin-bottom: 2rem;
  color: #212121;
}

.cta-button {
  background-color: hsl(54, 98%, 68%);
  color: #333;
  padding: 0.8rem 1.5rem;
  border: none;
  border-radius: 5px;
  text-decoration: none;
  font-size: 1rem;
```

```
  font-weight: bold;
  transition: background-color 0.3s ease, transform 0.3s ease;
}

.cta-button:hover {
  background-color: #fdd835;
  transform: translateY(-3px);
}
@media (max-width: 768px) {
  .cta-section {
    padding: 2rem 1rem;
  }

  .cta-content h2 {
    font-size: 1.5rem;
  }

  .cta-content p {
    font-size: 1rem;
  }
}
footer {
  background-color: #4CAF50;
  color: white;
  padding: 2rem;
}

.footer-content {
  display: grid;
  grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));
  gap: 2rem;
  margin-bottom: 2rem;
}

.footer-section h3 {
  font-size: 1.2rem;
  margin-bottom: 1rem;
}
.footer-section p, .footer-section ul {
  color: #e0e0e0;
}
.footer-section ul {
  list-style: none;
  padding: 0;
}
.footer-section ul li {
  margin-bottom: 0.5rem;
}
.footer-section ul li a {
  color: #e0e0e0;
```

```
    text-decoration: none;
    transition: color 0.3s ease;
  }
  .footer-section ul li a:hover {
    color: #ffeb3b;}
  .social-icons {
    display: flex;
    gap: 1rem;}
  .social-icons a {
    color: white;
    font-size: 1.2rem;
    transition: color 0.3s ease, transform 0.3s ease;
  }
  .social-icons a:hover {
    color: #ffeb3b;
    transform: translateY(-3px);
  }
  .footer-bottom {
    text-align: center;
    padding-top: 1rem;
    border-top: 1px solid rgba(255, 255, 255, 0.1);
  }
  @keyframes fadeIn {
    from {
      opacity: 0;
    }
    to {
      opacity: 1;
    }
  }
  @keyframes float {
    0%, 100% {
      transform: translateY(0);
    }
    50% {
      transform: translateY(-10px);
    }
  }
.profile-section {
    display: flex;
    align-items: center;
    position: relative;
}
.profile-info {
    display: flex;
    align-items: center;
    gap: 10px;
    cursor: pointer;
    padding: 8px 12px;
    border-radius: 20px;
```

```
    background-color: rgba(255, 255, 255, 0.1);
    transition: all 0.3s ease;
}
.profile-info:hover {
    background-color: rgba(255, 255, 255, 0.2);
}
.profile-icon {
    width: 32px;
    height: 32px;
    background-color: #ffeb3b;
    border-radius: 50%;
    display: flex;
    align-items: center;
    justify-content: center;
    color: #4CAF50;
}

.shop-name {
    color: white;
    font-weight: 500;
    font-size: 0.95rem;
}
.profile-dropdown.show {
    display: block;
    animation: fadeIn 0.2s ease;
}
.profile-dropdown a {
    display: flex;
    align-items: center;
    gap: 10px;
    padding: 10px 16px;
    color: #333;
    text-decoration: none;
    transition: background-color 0.3s ease;
}
.profile-dropdown a:hover {
    background-color: #f5f5f5;
}
.profile-dropdown i {
    color: #4CAF50;
    width: 20px;
}
.modal {
    display: none;
    position: fixed;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    background-color: rgba(0, 0, 0, 0.5);
```

```
    z-index: 1000;
    animation: fadeIn 0.3s ease;
}
.modal-content {
    position: relative;
    background-color: #fff;
    margin: 10% auto;
    width: 90%;
    max-width: 500px;
    border-radius: 12px;
    box-shadow: 0 5px 15px rgba(0, 0, 0, 0.2);
    animation: slideIn 0.3s ease;
}
modal-header {
    padding: 20px;
    border-bottom: 1px solid #eee;
    display: flex;
    justify-content: space-between;
    align-items: center;
    background-color: #4CAF50;
    color: white;
    border-radius: 12px 12px 0 0;
}
.modal-header h2 {
    margin: 0;
    font-size: 1.5rem;
    display: flex;
    align-items: center;
    gap: 10px;
}
.modal-body {
    padding: 20px;
}
.profile-detail {
    display: flex;
    align-items: flex-start;
    gap: 15px;
    padding: 15px 0;
    border-bottom: 1px solid #eee;
}
.profile-detail:last-child {
    border-bottom: none;
}
.profile-detail i {
    color: #4CAF50;
    font-size: 20px;
    width: 24px;
}
.detail-info {
    flex: 1;
```

```
}
.detail-info label {
    display: block;
    color: #666;
    font-size: 0.9rem;
    margin-bottom: 4px;
}
modal-footer {
    padding: 20px;
    border-top: 1px solid #eee;
    text-align: right;
}
.btn-close {
    background-color: #4CAF50;
    color: white;
    border: none;
    padding: 10px 20px;
    border-radius: 6px;
    cursor: pointer;
    font-size: 1rem;
    transition: background-color 0.3s ease;
}
.btn-close:hover {
    background-color: #3d8b40;
}
@keyframes fadeIn {
    from { opacity: 0; }
    to { opacity: 1; }
}
@keyframes slideIn {
    from {
        transform: translateY(-20px);
        opacity: 0;
    }
    to {
        transform: translateY(0);
        opacity: 1;
    }
}
  </style>
        <link        rel="stylesheet"        href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css">
</head> <body>
  <div id="profileModal" class="modal">
    <div class="modal-content">
      <div class="modal-header">
        <h2><i class="fas fa-user-circle"></i> Profile Details</h2>
        <span class="close-modal">&times;</span>
      </div>
      <div class="modal-body">
```

```
      <div class="profile-detail">
        <i class="fas fa-store"></i>
        <div class="detail-info">
          <label>Shop Name</label>
          <p id="modalShopName"></p>
        </div> </div>
      <div class="profile-detail">
        <i class="fas fa-user"></i>
        <div class="detail-info">
          <label>Owner Name</label>
          <p id="modalOwnerName"></p>
        </div></div>
      <div class="profile-detail">
        <i class="fas fa-envelope"></i>
        <div class="detail-info">
          <label>Email</label>
          <p id="modalEmail"></p>
        </div></div>
      <div class="profile-detail">
        <i class="fas fa-phone"></i>
        <div class="detail-info">
          <label>Phone</label>
          <p id="modalPhone"></p>
        </div>
      </div>
    </div>
    <div class="modal-footer">
      <button class="btn-close">Close</button>
    </div>
  </div>
</div>
<header>
  <nav>
    <div class="logo">Fertilizer<span>Manager</span></div>
    <ul class="nav-links">
      <li><a href="#home">Home</a></li>
      <li><a href="afterlogindashboard.html">Features</a></li>
      <li><a href="#about">About</a></li>
      <li><a href="#contact">Contact</a></li>
    </ul>
    <div class="auth-buttons" id="authButtons">
      <a href="loginpage.html" class="login-btn">Login</a>
      <a href="signup.html" class="signup-btn">Sign Up</a>
    </div>
    <div class="profile-section" id="profileSection" style="display: none;">
      <div class="profile-info">
        <div class="profile-icon">
          <i class="fas fa-user"></i>
        </div>
        <span class="shop-name" id="shopName"></span>
        <i class="fas fa-chevron-down dropdown-icon"></i>
```

```
      </div>
      <div class="profile-dropdown">
            <a href="#" id="viewProfile"><i class="fas fa-user-circle"></i> View Profile</a>
        <a href="#" id="logoutBtn"><i class="fas fa-sign-out-alt"></i> Logout</a>
      </div>
    </div>
  </nav>
 </header>
 <main>
<section id="home" class="hero">
  <div class="hero-content">
   <h1>Revolutionize Your Fertilizer Inventory</h1>
     <p>Efficiently manage, track, and optimize your fertilizer stock with our advanced inventory management system.</p>
    <div class="cta-buttons">
     <a href="signup.html" class="cta-button">Get Started</a>
     <a href="#features" class="cta-button secondary">Learn More</a>
    </div>
  </div>
 </section>
<section id="features" class="features">
  <h2>Why Choose FertilizerManager?</h2>
  <div class="feature-grid">
   <div class="feature-card">
    <i class="fas fa-boxes"></i>
    <h3>Real-Time Tracking</h3>
    <p>Monitor your fertilizer stock in real-time with our intuitive dashboard.</p>
   </div>
   <div class="feature-card">
    <i class="fas fa-chart-line"></i>
    <h3>Advanced Analytics</h3>
    <p>Generate detailed reports and insights for better decision-making.</p>
   </div>
   <div class="feature-card">
    <i class="fas fa-truck"></i>
    <h3>Order Management</h3>
          <p>Easily manage incoming and outgoing orders with automated notifications.</p>
   </div>
   <div class="feature-card">
    <i class="fas fa-bell"></i>
    <h3>Alerts & Notifications</h3>
    <p>Get notified about low stock levels, expiring products, and more.</p>
   </div>
   <div class="feature-card">
    <i class="fas fa-credit-card"></i>
    <h3>Consumer Credit Management</h3>
    <p>Efficiently manage customer credit accounts and payment schedules.</p>
   </div>
  </div>
```

```
 </section>
<section id="about" class="about">
   <div class="about-content">
    <h2>Our Mission</h2>
          <p class="subtitle">Simplify inventory management for agricultural businesses.</p>
    <p>At FertilizerManager, our mission is simple to provide farmers and agricultural businesses with the tools they need to manage their fertilizer inventory efficiently. We believe that by simplifying inventory management, we can help reduce waste, save time, and increase productivity.</p>

    <div class="mission-statement">
     <h3>Why We Do What We Do</h3>
     <p>Fertilizer is a critical resource for farmers, but managing it can be challenging. Our platform was designed to address these challenges by offering:</p>
      <ul>
         <li><i class="fas fa-check"></i>  Always know what's in stock and what's running low.</li>
       <li><i class="fas fa-check"></i> Get notified when stock levels are low or when products are nearing expiration.</li>
            <li><i class="fas fa-check"></i>Perfect for small business inventory management.</li>
          <li><i class="fas fa-check"></i> A simple interface that anyone can use, regardless of technical expertise.</li>
      </ul>
    </div>
   </div>
 </section>
  <section id="contact" class="contact">
    <h2>Contact Us</h2>
    <p>Have questions or need assistance? Reach out to us!</p>
    <form class="contact-form">
     <input type="text" placeholder="Your Name" required>
     <input type="email" placeholder="Your Email" required>
     <textarea placeholder="Your Message" required></textarea>
     <button type="submit">Send Message</button>
    </form>
    <div class="form-message"></div>
  </section>
 </main>
  <section class="cta-section">
    <div class="cta-content">
     <h2>Ready to Do Your Best Work?</h2>
     <p>Let's get you started. Sign up now and take control of your fertilizer inventory.</p>
     <a href="signup.html" class="cta-button">SIGN UP NOW</a>
     </div>
    </section>
 <footer>
  <div class="footer-content">
   <div class="footer-section">
```

```
      <h3>FertilizerManager</h3>
      <p>Your trusted partner in fertilizer inventory management.</p>
    </div>
    <div class="footer-section">
      <h3>Quick Links</h3>
      <ul>
        <li><a href="#home">Home</a></li>
        <li><a href="#features">Features</a></li>
        <li><a href="#about">About</a></li>
        <li><a href="#contact">Contact</a></li>
      </ul>
    </div>
    <div class="footer-section">
      <h3>Follow Us</h3>
      <div class="social-icons">
        <a href="#"><i class="fab fa-facebook"></i></a>
        <a href="#"><i class="fab fa-twitter"></i></a>
        <a href="#"><i class="fab fa-linkedin"></i></a>
        <a href="#"><i class="fab fa-instagram"></i></a>
      </div>
    </div>
  </div>
  <div class="footer-bottom">
    <p>&copy; 2023 FertilizerManager. All rights reserved.</p>
  </div>
 </footer>
 <script src="./js/dashboard.js"></script>
</body>
</html>
```

homepage.js:

```
document.addEventListener('DOMContentLoaded', () => {
    // Check authentication first
    if (!checkAuth()) {
        // If not authenticated, ensure profile section is hidden
        const authButtons = document.getElementById('authButtons');
        const profileSection = document.getElementById('profileSection');
        if (authButtons) authButtons.style.display = 'flex';
        if (profileSection) profileSection.style.display = 'none';
        return;
    }
    const authButtons = document.getElementById('authButtons');
    const profileSection = document.getElementById('profileSection');
    const shopName = document.getElementById('shopName');
    const viewProfile = document.getElementById('viewProfile');
    const logoutBtn = document.getElementById('logoutBtn');
    const userId = localStorage.getItem('userId');
    const userData = localStorage.getItem('user');

    if (userId && userData) {
```

```
        try {
            const user = JSON.parse(userData);
            // Show profile section
            if (authButtons) authButtons.style.display = 'none';
            if (profileSection) {
                profileSection.style.display = 'flex';
                shopName.textContent = user.shop_name || 'My Shop';
            }
        } catch (error) {
            console.error('Error parsing user data:', error);
            // Show login/signup buttons on error
            if (authButtons) authButtons.style.display = 'flex';
            if (profileSection) profileSection.style.display = 'none';
            // Redirect to login page on error
            window.location.href = 'loginpage.html';
            return;
        }
    } else {
        // Not logged in, show login/signup buttons and redirect
        if (authButtons) authButtons.style.display = 'flex';
        if (profileSection) profileSection.style.display = 'none';
        window.location.href = 'loginpage.html';
        return;
    }
    const profileInfo = document.querySelector('.profile-info');
    const dropdown = document.querySelector('.profile-dropdown');

    if (profileInfo && dropdown) {
        profileInfo.addEventListener('click', (e) => {
            e.stopPropagation();
            dropdown.classList.toggle('show');
        });

        document.addEventListener('click', (e) => {
            if (!profileSection.contains(e.target)) {
                dropdown.classList.remove('show');}
        });}
    if (viewProfile) {
        const modal = document.getElementById('profileModal');
        const closeModal = document.querySelector('.close-modal');
        const closeButton = document.querySelector('.btn-close');
        viewProfile.addEventListener('click', (e) => {
            e.preventDefault();
            const user = JSON.parse(localStorage.getItem('user'));
           document.getElementById('modalShopName').textContent = user.shop_name ||
'N/A';
 document.getElementById('modalOwnerName').textContent = user.owner_name ||
'N/A';
            document.getElementById('modalEmail').textContent = user.email || 'N/A';
            document.getElementById('modalPhone').textContent = user.phone || 'N/A';
```

```
            modal.style.display = 'block';
            dropdown.classList.remove('show');
        });
        if (closeModal) {
            closeModal.addEventListener('click', () => {
                modal.style.display = 'none';
            });}
        if (closeButton) {
            closeButton.addEventListener('click', () => {
                modal.style.display = 'none';
            });}
        window.addEventListener('click', (e) => {
            if (e.target === modal) {
                modal.style.display = 'none';
            }});
    if (logoutBtn) {
        logoutBtn.addEventListener('click', (e) => {
            e.preventDefault();
            localStorage.removeItem('userId');
            localStorage.removeItem('lastAuthTime');
            localStorage.removeItem('user');
            window.location.href = 'loginpage.html';
        });}
    document.querySelectorAll('a[href^="#"]').forEach(anchor => {
        anchor.addEventListener('click', function (e) {
            e.preventDefault();
            const href = this.getAttribute('href');
            const target = document.querySelector(href);
            if (target) {
                target.scrollIntoView({
                    behavior: 'smooth'
                });}
        });});
    const contactForm = document.querySelector('.contact-form');
    if (contactForm) {
        contactForm.addEventListener('submit', function(e) {
            e.preventDefault();
            const formMessage = document.querySelector('.form-message');
            formMessage.textContent = 'Thank you for your message. We will get back to
you soon!';
            formMessage.style.color = '#4CAF50';
            this.reset();
        });}
    const navLinks = document.querySelector('.nav-links');
    if (window.innerWidth <= 768) {
        navLinks.style.display = 'none';

        if (!document.querySelector('.mobile-menu-toggle')) {
            const toggleButton = document.createElement('button');
            toggleButton.className = 'mobile-menu-toggle';
```

```
            toggleButton.innerHTML = '<i class="fas fa-bars"></i>';
            document.querySelector('nav').prepend(toggleButton);
            toggleButton.addEventListener('click', () => {
                const currentDisplay = navLinks.style.display;
                navLinks.style.display = currentDisplay === 'none' ? 'flex' : 'none';
            });
        }}});
function checkAuth() {
    const userId = localStorage.getItem('userId');
    const lastAuthTime = localStorage.getItem('lastAuthTime');
    const sessionTimeout = 30 * 60 * 1000; // 30 minutes
    if (!userId || !lastAuthTime) {
        return false;}
    const currentTime = new Date().getTime();
    if (currentTime - parseInt(lastAuthTime) >= sessionTimeout) {
        localStorage.removeItem('userId');
        localStorage.removeItem('lastAuthTime');
        localStorage.removeItem('user');
        return false;
    }return true;
}
```

Chapter 7

Conclusion

The Inventory and Business Management System for agriculture supplies is a comprehensive web-based solution designed to streamline the operations of small-to-medium retail businesses, especially those dealing with agricultural supplies. It simplifies the management of stock, sales, suppliers, customer credit, and payments through a user-friendly interface and intelligent backend logic.

Built using HTML, CSS, and JavaScript for the frontend, with enhanced visuals through font awesome icons, the system ensures a clean and responsive user experience. The backend is powered by Node.js and Express.js, providing a fast and scalable server-side environment. MySQL serves as the relational database, effectively organizing and managing large volumes of structured data. For secure user authentication and password protection, bcrypt is implemented. Additionally, file management is handled through fileupload, enabling seamless handling of images and documents like product photos or receipts.

Each module—Inventory, Sales, Credit Management, Customer Profiles, Supplier Handling, and Reporting—is integrated in a modular fashion, allowing for clean data flow, ease of maintenance, and future scalability. The system supports secure user authentication, low stock and expiry alerts, credit interest tracking, and partial payments, making it ideal for businesses that offer flexible customer billing. ER diagrams and DFD levels (0, 1, and 2) were used during the design phase to outline the structural and process-oriented flow of the system. The UML Use Case Diagram maps the interaction between the users and the core functionalities, ensuring all user needs are addressed systematically.

In conclusion, this system is a robust digital framework that enhances operational efficiency, improves customer credit tracking, and supports decision-making through detailed reporting. With strong backend technologies and a responsive frontend, it stands as a reliable tool that bridges traditional inventory management with modern digital convenience, ready for real-world deployment and future feature expansion.

Chapter 8

Future Work

While the existing Inventory and Business Management System for agricultural supplies effectively achieves its core objectives, strategic enhancements can significantly improve its efficiency, scalability, and user engagement across campuses. Additionally, strengthening security measures, enabling multi-platform accessibility, and incorporating automated reporting tools will further streamline operations. By refining these aspects, the system can offer a more comprehensive, adaptive, and user-friendly solution for agricultural supply management.

- Mobile Application Integration: Develop a mobile app version for Android/iOS to allow shop owners to manage inventory and credit on the go.
- SMS/Email Notifications: Send automatic alerts to customers for due payments, low stock, or order confirmations.
- GST/Billing Integration: Add support for tax calculation and generating GST-compliant invoices for official transactions.
- Multi-language Support: Include regional language options to make the system more user-friendly for local shopkeepers.
- Advanced Analytics Dashboard: Implement visual charts and graphs for profit/loss, sales trends, and credit recovery status.
- Barcode Scanner Integration: Allow quick product entry and billing using barcode scanners for efficiency.
- Role-Based Access Control (RBAC): Add support for multiple user roles (e.g., manager, staff) with different permissions.
- Data Backup and Restore: Provide options for automated backups and secure data restoration in case of failure.
- Customer Credit Score Prediction: Use simple ML models to suggest credit limits and predict repayment reliability.
- Supplier Payment Tracking: Add modules to manage pending payments to suppliers along with due reminders.
- Dynamic Pricing Suggestions: Provide automated suggestions for price adjustments based on demand, expiry, or supplier rates.
- AI Chatbot for Support: Provide a chatbot to help shop owners with FAQs, usage help, and troubleshooting.

  These improvements, once implemented, will help transition the platform from a pilot project to a robust solution ready for real-world deployment and long-term use.

Chapter 9

References

[1] Gaddam, Rashmika Ms, “Sales and Stock Management System”, 2023.

[2] Ina Freeman “Inventory Planning In Small Business”, 2008.

[3] Athieno lilian hope: “Stock Management System”, 2023.

[4] Johni S Pasaribu:“Development of a Web Based Inventory Information System”,2023