**FARM MANAGEMENT SYSTEM**

**A MINI PROJECT REPORT**

**Submitted by**

**ABISHEK P.S. 220701010**

**AMALESH .B 220701024**

**ABILASH A 220701004**

**In partial fulfillment for the award of the degree of**

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE**

**RAJALAKSHMI ENGINEERING COLLEGE (AUTONOMOUS)**

**THANDALAM**

**CHENNAI-602105**

**2023 – 24**

**BONAFIDE CERTIFICATE**

Certified that this project report **“FARM MANAGEMENT SYSTEM”** is the bonafide work of

**“ABISHEK P.S. ( 220701010), AMALESH B (220701024), ABILASH A(220701004)”**

who carried out the project work under my supervision**.**

**Submitted for the Practical Examination held on\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**SIGNATURE SIGNATURE**

**Dr.R.SABITHA Mrs.S. Vinothini**

**Professor and II Year Academic Head Assistant Professor (SG),**

**Computer Science and Engineering, Computer Science and Engineering,**

**Rajalakshmi Engineering College Rajalakshmi Engineering College,**

**(Autonomous), (Autonomous),**

**Thandalam, Chennai - 602 105 Thandalam, Chennai - 602 105**

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**ABSTRACT**

The Farm Management System (FMS) serves as a pivotal tool in the agricultural landscape, facilitating the seamless coordination of operations and the efficient management of resources. With its user-centric design, FMS caters to the diverse needs of Customers, Employees, and Admins, each with distinct roles and responsibilities within the system.

Customers are granted access to a comprehensive database housing detailed information on plants and medicines, empowering them to make informed decisions regarding their agricultural pursuits. Through an intuitive interface, customers can browse, search, and access essential data such as plant descriptions, medicinal properties, and availability, enhancing their ability to cultivate and nurture their crops effectively.

Meanwhile, Employees wield the administrative capabilities of FMS to maintain the integrity of the system's data. Empowered with the authority to add, delete, or modify plant and medicine entries, employees ensure that the database remains up-to-date and reflective of the agricultural inventory. Their meticulous oversight guarantees accurate records, enabling smooth operations and informed decision-making at every stage of the agricultural process.

At the helm of the FMS infrastructure are Admins, entrusted with overarching control and oversight. Armed with elevated privileges, admins oversee user management, ensuring that employees and customers are appropriately registered and managed within the system. Moreover, admins exercise authority over core data entities, including plants and medicines, to maintain data integrity and system functionality.

Through meticulous data abstraction and streamlined functionalities, FMS revolutionizes agricultural management, empowering stakeholders with the tools necessary for sustainable and profitable operations. By fostering collaboration, transparency, and efficiency, FMS stands as a beacon of innovation in the agricultural landscape, driving progress and prosperity for farmers, enterprises, and communities alike.

**TABLE OF CONTENTS**

1. **INTRODUCTION**

1.1 INTRODUCTION----------------------------------------------------------------------**5**

1.2 OBJECTIVES---------------------------------------------------------------------------**6**

1.3 MODULES -----------------------------------------------------------------------------**8**

**2. SURVEY OF TECHNOLOGIES**

2.1 SOFTWARE DESCRIPTION------------------------------------------------------**10**

2.2 LANGUAGES-------------------------------------------------------------------------**12**

**3. REQUIREMENTS AND ANALYSIS**

3.1 REQUIREMENT SPECIFICATION----------------------------------------------**13**

3.2 HARDWARE AND SOFTWARE REQUIREMENTS---------------------------**15**

3.3 ARCHITECTURE DIAGRAM-----------------------------------------------------**20**

3.4 ER DIAGRAM-------------------------------------------------------------------------**22**

**4. PROGRAM CODE**--------------------------------------------------------------------**24**

**5. RESULTS** -------------------------------------------------------------------------------**49**

**6.CONCLUSION**-------------------------------------------------------------------------**57**

**7.REFERENCES**----------------------------------------------------------------------**58**

**1. 1.INTRODUCTION**

Introducing the Farm Management System (FMS): Pioneering the Future of Agricultural Excellence

In an era where the demands on agriculture are more complex than ever, the Farm Management System (FMS) emerges as a transformative solution, poised to revolutionize the way farmers and agricultural stakeholders navigate the intricacies of their operations. With its comprehensive suite of advanced functionalities and intuitive interfaces, the FMS represents a paradigm shift in farm management, offering a seamless integration of technology, data analytics, and traditional practices to unlock unprecedented levels of efficiency, sustainability, and profitability.

At its heart, the FMS is a powerful tool designed to empower farmers and industry professionals with the insights and capabilities needed to optimize every aspect of their agricultural endeavors. From crop planning and monitoring to inventory management and financial analysis, the FMS provides a robust framework for decision-making, enabling users to make informed choices that drive productivity and growth.

But the FMS is more than just a software solution – it embodies a holistic approach to agricultural management that extends beyond the confines of the digital realm. By incorporating principles of sustainability, environmental stewardship, and risk mitigation, the FMS equips users with the tools and knowledge needed to cultivate crops responsibly and resiliently in the face of evolving challenges such as climate change, resource scarcity, and market volatility.

Moreover, the FMS serves as a catalyst for collaboration and innovation within the agricultural community, fostering a culture of knowledge sharing, best practices, and continuous improvement. Through its open architecture and customizable features, the FMS encourages users to adapt and innovate, leveraging emerging technologies and novel strategies to address the unique needs of their operations.

As we embark on this journey into the future of agriculture, the Farm Management System (FMS) stands as a testament to the power of technology to transform traditional practices and propel the industry forward. With its unparalleled capabilities and unwavering commitment to excellence, the FMS promises to usher in a new era of agricultural innovation, resilience, and prosperity for generations to come.

**1.2.OBJECTIVE**

The Farm Management System (FMS) endeavors to revolutionize agricultural management by providing a comprehensive and intuitive platform that addresses the diverse needs of stakeholders within the agricultural ecosystem. The primary objectives of FMS are as follows:

Enhance Operational Efficiency: FMS aims to streamline agricultural operations by digitizing and centralizing essential processes, thereby minimizing manual efforts and reducing the likelihood of errors associated with traditional paper-based methods.

Facilitate Informed Decision-Making: FMS empowers farmers, agricultural enterprises, and other stakeholders with access to a vast repository of data on plants and medicines. By providing detailed insights into cultivation practices, medicinal properties, availability, and pricing, FMS enables informed decision-making at every stage of agricultural management.

Optimize Resource Utilization: By facilitating real-time monitoring and management of plant and medicine inventory, FMS enables stakeholders to optimize resource allocation, minimize wastage, and maximize productivity in agricultural endeavors.

Improve Inventory Management: FMS offers robust inventory management capabilities, allowing users to add, delete, and update plant and medicine details seamlessly. By maintaining accurate and up-to-date inventory records, FMS ensures efficient resource utilization and minimizes stockouts or overstocking situations.

Empower Stakeholder Collaboration: FMS serves as a collaborative platform that fosters communication and cooperation among farmers, suppliers, distributors, and other stakeholders. By facilitating data sharing and collaboration, FMS enhances synergy and promotes collective efforts towards achieving common agricultural goals.

Ensure Data Integrity and Security: FMS prioritizes data integrity and security by implementing robust authentication, authorization, and encryption mechanisms. By safeguarding sensitive information and preventing unauthorized access, FMS instills confidence among users regarding the privacy and security of their data.

Enable Scalability and Adaptability: FMS is designed to accommodate the evolving needs of agricultural enterprises, allowing for scalability and adaptability to changing business requirements, technological advancements, and regulatory frameworks.

Promote Sustainable Agricultural Practices: By providing access to information on sustainable cultivation practices and environmentally friendly agricultural inputs, FMS encourages stakeholders to adopt sustainable farming methods that minimize environmental impact and promote long-term ecological resilience.

Drive Innovation and Growth: FMS serves as a catalyst for innovation within the agricultural sector by leveraging emerging technologies such as data analytics, IoT (Internet of Things), and AI (Artificial Intelligence) to unlock new insights, optimize processes, and drive continuous improvement and growth.

Empowerment Through Accessibility: FMS is designed to be accessible and user-friendly, catering to users with varying levels of technological proficiency and ensuring inclusivity across diverse demographics and geographical regions.

In summary, the overarching objective of the Farm Management System is to empower agricultural stakeholders with the tools, insights, and capabilities necessary to realize sustainable growth, productivity, and prosperity in agricultural endeavors.

**1.3 MODULES**

The Farm Management System integrates essential modules for seamless operation. User Management oversees secure access, while Plant and Medicine Management organizes data efficiently. Employee Management ensures staff details are managed effectively, while Admin Privileges maintain system integrity. Robust Security measures protect sensitive data, while Reporting offers valuable insights for informed decision-making. Together, these modules optimize farm management, promoting productivity and streamlined operations.

**1.31 User Authentication Module:**

This module serves as the gatekeeper of the system, managing the authentication process to ensure that only authorized users can access the system. It handles user login, registration, and authentication, verifying the identity of users before granting them access to the system's functionalities.

**1.32 Customer Module:**

The Customer Module caters to the needs of customers interacting with the system. It provides a user-friendly interface for customers to explore information about available plants, medicines, and other products. Customers can browse, search, and filter plants and medicines based on their preferences. Additionally, they can place orders for desired items and manage their profiles, including updating personal information and viewing order history.

**1.33 Employee Module:**

Designed for employees responsible for managing inventory and assisting customers, the Employee Module offers functionalities to add, delete, and update plant and medicine details within the system. Employees can efficiently manage inventory by monitoring stock levels, updating availability status, and handling customer queries and orders through the system.

**1.34 Admin Module:**

The Admin Module grants administrators full access to all system functionalities. Admins have the authority to manage employee accounts, including adding, deleting, and updating employee information. They can also handle customer accounts, view customer lists, and address customer support issues effectively. Furthermore, admins can manage plant and medicine details by adding, deleting, and updating product information as needed.

**1.35 Plant Management Module:**

This module focuses specifically on managing plant-related information within the system. Users can view detailed information about various types of plants available on the farm, including descriptions, categories, and availability status. They can add new plants to the inventory, update existing plant details, and remove plants that are no longer available.

**1.36 Medicine Management Module:**

Similar to the Plant Management Module, the Medicine Management Module is dedicated to managing information related to medicines available on the farm. Users can access detailed information about different medicines, including descriptions, prices, and availability. They can add new medicines, update existing medicine details, and remove medicines from the inventory when necessary.

**1.37 Inventory Management Module:**

The Inventory Management Module is responsible for tracking the availability and stock levels of plants and medicines within the system. It provides functionalities for managing inventory, including receiving new stock, updating stock levels based on sales and returns, and handling product returns or exchanges efficiently.

**1.38 FEEDBACK MODULE:**

The Feedback Module enables users to submit queries, suggestions, or complaints, capturing user details for personalized responses. It categorizes feedback for efficient handling, escalating critical issues and routing to relevant departments. Administrators track and resolve feedback, utilizing analytics for continuous improvement. This closed-loop process fosters transparency and enhances user satisfaction.

**2. SURVEY OF TECHNOLOGIES**

**2.1SOFTWARE DESCRIPTION: FARM MANAGEMENT SYSTEM**

**2.11Overview:**

The Farm Management System, built with PHP, is a comprehensive platform designed to streamline farm operations. It effectively manages various aspects including plant and medicine inventory, customer orders, employee activities, and administrative tasks.

**2.12Key Features:**

**User Authentication and Authorization:**

Secure login system with role-based access control ensures data security and integrity. It handles user login, registration, and authentication, verifying the identity of users before granting them access to the system's functionalities.

**Customer Management:**

Customers can browse products, place orders, and track order status effortlessly. It provides a user-friendly interface for customers to explore information about available plants, medicines, and other products. Customers can browse, search, and filter plants and medicines based on their preferences. Additionally, they can place orders for desired items and manage their profiles, including updating personal information and viewing order history.

**Employee Management:**

Employees can efficiently manage inventory, process orders, and provide support to customers. Employees can efficiently manage inventory by monitoring stock levels, updating availability status, and handling customer queries and orders through the system.

**Admin Dashboard:**

Administrators have centralized control over system functionalities, employee activities, and inventory management. Admins have the authority to manage employee accounts, including adding, deleting, and updating employee information. They can also handle customer accounts, view customer lists, and address customer support issues effectively. Furthermore, admins can manage plant and medicine details by adding, deleting, and updating product information as needed.

**Inventory Tracking:**

Real-time tracking of plant and medicine inventory levels to ensure availability and optimize stock management. It provides functionalities for managing inventory, including receiving new stock, updating stock levels based on sales and returns, and handling product returns or exchanges efficiently.

**Conclusion:**

The Farm Management System offers a robust solution for modern farm management needs. With its user-friendly interface, advanced functionalities, and scalability, it empowers farm administrators, employees, and customers to enhance productivity and efficiency in agricultural operations

**2.2Language Description:**

The Farm Management System (FMS) is a web-based application designed to optimize agricultural operations. Using HTML, CSS, JavaScript, PHP, and SQL, FMS provides a user-friendly platform for customers, employees, and administrators. It supports three user roles: customers access plant and medicine information, employees manage data, and administrators oversee system functionality. FMS integrates advanced technologies to streamline workflows, facilitate informed decision-making, and drive efficiency and sustainability in farming practices.

**2.21 HTML (Hypertext Markup Language):**

HTML is the standard markup language used for creating the structure of web pages. In the Farm Management System (FMS), HTML will be utilized to design the user interface, defining the layout and elements such as buttons, forms, tables, and navigation menus. It provides the foundation for presenting information and interacting with users on the web platform.

**2.22 CSS (Cascading Style Sheets):**

CSS is a style sheet language used to enhance the presentation and appearance of HTML elements. In the FMS project, CSS will be employed to customize the visual aspects of the user interface, including colors, fonts, sizes, spacing, and overall layout. By applying CSS rules, we can ensure consistency and aesthetic appeal across the entire application, creating a seamless user experience.

**2.23 JavaScript:**

JavaScript is a versatile programming language primarily used for adding interactivity and dynamic behavior to web pages. In the FMS project, JavaScript will be utilized to implement client-side functionality, such as form validation, interactive elements, and asynchronous requests to the server. By leveraging JavaScript, we can enhance user engagement and responsiveness, enabling seamless interaction with the application without requiring page reloads.

**2.24 PHP (Hypertext Preprocessor):**

PHP is a server-side scripting language commonly used for developing dynamic web applications. In the FMS project, PHP will serve as the backend scripting language responsible for processing user requests, accessing the database, and generating dynamic content. With PHP, we can handle tasks such as user authentication, data manipulation, and business logic implementation, ensuring the smooth operation of the Farm Management System.

**2.25 SQL (Structured Query Language):**

SQL is a standard language used for managing and manipulating relational databases. In the FMS project, SQL will be employed to define the database schema, create tables, insert, update, delete, and retrieve data. By writing SQL queries, we can interact with the underlying database system, storing and retrieving information related to plants, medicines, customers, employees, and other entities within the Farm Management System. SQL ensures data integrity, consistency, and efficient data management, serving as the backbone of the application's data layer.

**3. REQUIREMENTS AND ANALYSIS**

**3.1 REQUIREMENT SPECIFICATION:**

**3.11. User Management:**

**Requirement**: Implement a user authentication system with role-based access control.

**Functionalities:** Registration and login for customers, employees, and administrators.Different levels of access based on user roles: customer, employee, and admin.

**3.12. Customer Management:**

**Requirement**: Enable customers to browse products, place orders, and track order status.

**Functionalities:**View product catalog with detailed descriptions and images.Add items to cart, specify quantities, and proceed to checkout.View order history and track order status.

**3.13. Employee Management:**

**Requirement**: Provide employees with tools to manage inventory, process orders, and handle customer queries.

**Functionalities:**Add, update, and delete plant and medicine details in the inventory.Process customer orders, update order status, and handle returns or refunds.Respond to customer queries and provide support.

**3.14. Admin Dashboard:**

**Requirement**: Offer administrators centralized control over system functionalities and data management.

**Functionalities**:View and manage employee accounts, including adding, updating, and deleting employees.Monitor customer activities, manage customer accounts, and handle customer support issues.Manage plant and medicine details, including adding, updating, and deleting product information.

**3.15. Inventory Management:**

**Requirement**: Track the availability and stock levels of plants, medicines, and other products.

**Functionalities:**Receive new stock, update quantities, and handle product returns.

Generate alerts for low stock levels and provide insights for inventory optimization.

**3.16. Reporting and Analytics:**

**Requirement**: Generate reports on sales performance, inventory levels, and customer behavior.

**Functionalities:**Generate customizable reports on various aspects of the farm management system.Provide data visualization tools for analyzing trends and making informed decisions.

**3.17. Security:**

**Requirement**: Implement security measures to protect sensitive data and prevent unauthorized access.

**Functionalities:**Encrypt user passwords and sensitive information stored in the database.Implement input validation and data sanitization to prevent common security threats such as SQL injection and cross-site scripting (XSS).

**3.18. Scalability and Performance:**

**Requirement**: Ensure that the system can handle a large number of concurrent users and scale up as the user base grows.

**Functionalities**:Optimize code performance and database queries for efficiency.

Implement caching mechanisms to improve system responsiveness and reduce server load.

**3.19. Usability:**

**Requirement**: Create a user-friendly interface that is intuitive and easy to navigate.

**Functionalities:**Design an intuitive user interface with clear navigation and well-organized layout.Provide helpful tooltips, error messages, and guidance to assist users in completing tasks.

**3.110. Compatibility and Integration:**

**Requirement**: Ensure compatibility with various web browsers and devices, and support integration with external systems if needed.

**Functionalities**:Test the system on different web browsers and devices to ensure compatibility.

Provide APIs or integration points for integrating with external systems such as payment gateways or shipping providers

**3.2 HARDWARE REQUIREMENTS**

**3.21. Server:**

A dedicated server to host the Farm Management System application.

Minimum CPU: Dual-core processor (Intel Core i3 or equivalent)

Minimum RAM: 4 GB

Storage: SSD storage recommended for better performance, minimum 50 GB disk space

Operating System: Linux-based operating system (e.g., Ubuntu Server, CentOS)

Network: Stable internet connection with sufficient bandwidth for user access and database transactions.

**3.22. Database Server:**

Separate server or instance to host the database management system (e.g., MySQL, PostgreSQL).

Minimum CPU: Dual-core processor

Minimum RAM: 2 GB

Storage: SSD storage recommended, minimum 20 GB disk space

Operating System: Linux-based operating system

Network: Connected to the application server with low latency for efficient data retrieval and storage.

**3.23. Backup System:**

Regular backup solution to ensure data integrity and disaster recovery.

Automated backup schedule for both application and database servers.

Off-site backup storage for redundancy and protection against data loss.

**3.24. Networking Equipment:**

Router: High-speed router with firewall capabilities to manage network traffic.

Switch: Gigabit Ethernet switch for connecting servers and client devices.

Cabling: Cat6 Ethernet cables for reliable network connectivity.

**3.25. Client Devices:**

Desktop Computers or Laptops: For accessing the Farm Management System application.

Minimum Requirements: Dual-core processor, 4 GB RAM, modern web browser (Chrome, Firefox, Safari, Edge).

Mobile Devices: Optional for accessing the system via mobile-responsive web interface.

Internet Connectivity: Stable internet connection for accessing the application remotely.

**3.3 SOFTWARE REQUIREMENTS**

**3.311 Operating System Compatibility:**

The system should be designed to run on various operating systems, including Windows, macOS, and Linux distributions, to accommodate a wide range of users.

**3.312 Web Server Software:**

A reliable web server software such as Apache HTTP Server or Nginx is necessary to host the PHP-based application and serve web pages to users.

**3.313 Database Management System (DBMS):**

The system requires a robust and scalable relational database management system (RDBMS) like MySQL, PostgreSQL, or MariaDB to store and manage data efficiently.

**3.314 Programming Language:**

PHP, being a server-side scripting language, is fundamental for developing dynamic web applications. It enables the execution of server-side scripts and interaction with the database.

**3.315 Web Browser Compatibility:**

The system should be compatible with popular web browsers such as Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari to ensure optimal user experience across different platforms.

**3.316 Version Control System (VCS):**

Utilizing a version control system like Git is essential for managing source code, tracking changes, and collaborating with a team effectively, ensuring codebase integrity and version management.

**3.317 Integrated Development Environment (IDE):**

Developers benefit from using feature-rich integrated development environments (IDEs) like PhpStorm, Visual Studio Code, or Sublime Text for coding, debugging, and project management tasks.

**3.318 Frontend Frameworks/Libraries:**

Employing frontend frameworks or libraries such as Bootstrap, jQuery, or React.js can enhance the user interface, interactivity, and responsiveness of the system, improving user experience and engagement.

**3.319 Dependency Management:**

Composer, a dependency manager for PHP, simplifies the installation and management of PHP packages and libraries, ensuring efficient dependency resolution and version control.

**3.320 Authentication and Authorization Libraries:**

Integration of authentication and authorization libraries like Laravel Passport or PHP's built-in session management is vital for securing user access, protecting sensitive data, and managing user roles and permissions effectively.

**3.321 Security Measures:**

Implementing security measures such as SSL/TLS certificates, encryption techniques, and secure coding practices is crucial for safeguarding data integrity, protecting against cyber threats, and ensuring compliance with regulatory requirements.

**3.322 Backup and Recovery Mechanisms:**

Establishing robust backup and recovery mechanisms, including regular backups of the database and system files, along with a well-defined recovery plan, is essential for data protection, disaster recovery, and business continuity.

**3.323 Performance Monitoring Tools:**

Utilizing performance monitoring tools like New Relic, Datadog, or Apache JMeter enables continuous monitoring of system performance, identification of bottlenecks, and optimization of resource usage to ensure optimal system performance and scalability.

**3.324 Documentation Tools:**

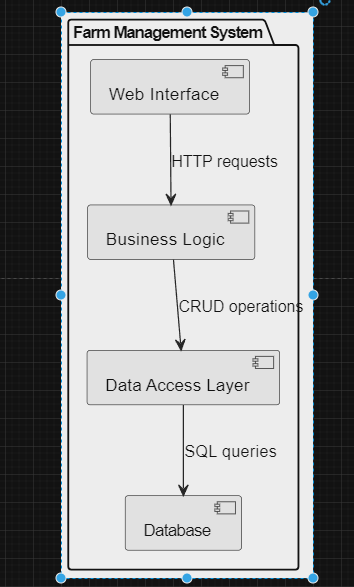
Employing documentation tools such as Markdown, Sphinx, or Doxygen facilitates the documentation of code, APIs, system architecture, and user manuals, ensuring comprehensive and well-structured documentation for future reference, maintenance, and knowledge transfer.

**3.325 Testing Frameworks:**

Incorporating testing frameworks like PHPUnit for PHP unit testing or Selenium for web browser automation enables developers to conduct automated testing, ensuring the reliability, functionality, and quality of the system throughout the development lifecycle.

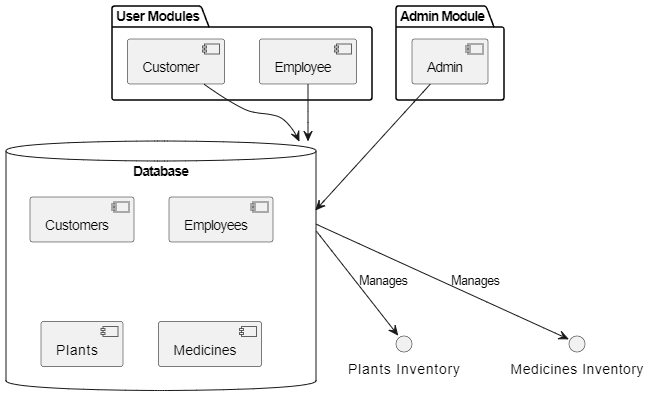
These comprehensive software requirements lay the groundwork for the development, deployment, and maintenance of the Farm Management System, ensuring compatibility, security, efficiency, and scalability across all stages of the software development lifecycle

**3.4 ARCHITECTURE DIAGRAM**



**Fig:3.41**

This architecture diagram illustrates the high-level components and their interactions within the Farm Management System, providing a foundation for system development and understanding.



**Fig:3.42**

**3.41 Web Browser (Client):** Represents the user interface where users interact with the system.

**3.42 Web Server (Apache/Nginx):** Hosts the PHP-based application and serves web pages to users.

**3.43 PHP Engine (Server-side):** Executes server-side scripts, handles user requests, and interacts with the database.

**3.44 Database (MySQL, PostgreSQL, etc.):** Stores and manages data related to plants, medicines, users, orders, etc.

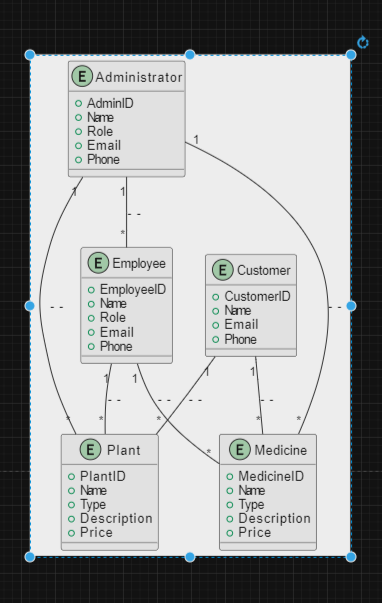
**3.45 Application Logic Layer:** Contains the business logic of the system, including authentication, authorization, order processing, etc.

**3.46 Data Access Layer (DAL):** Facilitates communication between the application logic layer and the database, handling data retrieval, storage, and manipulation.

**3.47 User Interface Presentation Layer (UI):** Renders the user interface components and interacts with the application logic layer based on user inputs.

**3.48 Database Abstraction Layer (ORM):** Provides an abstraction layer over the database, allowing seamless interaction with database entities using object-oriented programming principles.

**3.5 ER DIAGRAM**



**Fig:3.51**

**In this ER diagram:**

**3.51 Entities:**

Customer: Represents individuals who register on the system to interact with the farm. They have unique IDs, names, and email addresses.

**3.511 Employee**: Represents individuals who work within the farm. They also have unique IDs, names, and positions within the farm.

**3.512 Admin**: Represents administrators who oversee the entire system. They have unique IDs, names, and email addresses.

**3.513 Plant**: Represents plants available in the farm. Each plant has a unique ID, name, description, and price.

**3.514 Medicine**: Represents medicines available in the farm. Each medicine has a unique ID, name, description, and price.

**3.515 Feedback**: Represents feedback provided by customers. Each feedback has a unique ID and is associated with a specific order. It includes the feedback date, description, and the email of the customer providing the feedback.

**3.52 Relationships:**

**3.521 Feedback-Customer:** Represents an optional relationship between feedback and customer. A feedback may or may not be associated with a customer. This is depicted by the diamond shape, indicating optionality.

**3.522 Employee-Plant/Medicine:** Represents the fact that an employee can manage plants and medicines. They can add, delete, or update plant and medicine details. This is an association relationship.

**3.523 Attributes:**

Each entity has its own set of attributes that define its properties. These attributes are listed within the rectangles representing the entities in the diagram.

**3.524 Cardinalities:**

The cardinality of each relationship indicates the number of instances of one entity that can be associated with the number of instances of another entity. For example, in the Customer-Order relationship, it's depicted as one customer having multiple orders. Overall, this ER diagram provides a structured overview of the entities, their attributes, relationships, and cardinalities within your Farm Management System. It serves as a blueprint for designing and implementing the database schema for your system**.**

**4. PROGRAM CODE**

**<!DOCTYPE html>**

**<?php**

**include('func.php');**

**$con=mysqli\_connect("localhost","root","","fms");**

**$username = $\_SESSION['username'];**

**?>**

**<html lang="en">**

**<head>**

**<title>ADMT Farm Management</title>**

**<link rel="shortcut icon" type="image/x-icon" href="logo.jpeg"/>**

**<!-- Required meta tags -->**

**<meta charset="utf-8">**

**<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">**

**<link rel="stylesheet" type="text/css" href="font-awesome-4.7.0/css/font-awesome.min.css">**

**<link rel="stylesheet" href="style.css">**

**<!-- Bootstrap CSS -->**

**<link rel="stylesheet" href="vendor/fontawesome/css/font-awesome.min.css">**

**<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/css/bootstrap.min.css" integrity="sha384-/Y6pD6FV/Vv2HJnA6t+vslU6fwYXjCFtcEpHbNJ0lyAFsXTsjBbfaDjzALeQsN6M" crossorigin="anonymous">**

**<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">**

**<link href="https://fonts.googleapis.com/css?family=IBM+Plex+Sans&display=swap" rel="stylesheet">**

**<nav class="navbar navbar-expand-lg navbar-dark bg-primary fixed-top">**

**<a class="navbar-brand" href="#"><i class="fa fa-user-plus" aria-hidden="true"></i> ADMT Farm Management </a>**

**<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">**

**<span class="navbar-toggler-icon"></span>**

**</button>**

**<style >**

**.btn-outline-light:hover{**

**color: #25bef7;**

**background-color: #f8f9fa;**

**border-color: #f8f9fa;**

**}**

**</style>**

**<style >**

**.bg-primary {**

**background: -webkit-linear-gradient(left, #3931af, #00c6ff);**

**}**

**.list-group-item.active {**

**z-index: 2;**

**color: #fff;**

**background-color: #342ac1;**

**border-color: #007bff;**

**}**

**.text-primary {**

**color: #342ac1!important;**

**}**

**</style>**

**<div class="collapse navbar-collapse" id="navbarSupportedContent">**

**<ul class="navbar-nav mr-auto">**

**<li class="nav-item">**

**<a class="nav-link" href="logout1.php"><i class="fa fa-sign-out" aria-hidden="true"></i>Logout</a>**

**</li>**

**<li class="nav-item">**

**<a class="nav-link" href="#"></a>**

**</li>**

**</ul>**

**</div>**

**</nav>**

**</head>**

**<style type="text/css">**

**button:hover{cursor:pointer;}**

**#inputbtn:hover{cursor:pointer;}**

**</style>**

**<body style="padding-top:50px;">**

**<div class="container-fluid" style="margin-top:50px;">**

**<h3 style = "margin-left: 40%; padding-bottom: 20px;font-family:'IBM Plex Sans', sans-serif;"> Welcome &nbsp<?php echo $\_SESSION['username'] ?> </h3>**

**<div class="row">**

**<div class="col-md-4" style="max-width:18%;margin-top: 3%;">**

**<div class="list-group" id="list-tab" role="tablist">**

**<a class="list-group-item list-group-item-action active" href="#list-dash" role="tab" aria-controls="home" data-toggle="list">Dashboard</a>**

**<a class="list-group-item list-group-item-action" href="#list-plant" id="list-plant-list" role="tab" data-toggle="list" aria-controls="home">Plants</a>**

**<a class="list-group-item list-group-item-action" href="#list-med" id="list-med-list" role="tab" data-toggle="list" aria-controls="home">Medicine</a>**

**<a class="list-group-item list-group-item-action" href="#list-method" id="list-method-list" role="tab" data-toggle="list" aria-controls="home">Methods</a>**

**</div><br>**

**</div>**

**<div class="col-md-8" style="margin-top: 3%;">**

**<div class="tab-content" id="nav-tabContent" style="width: 950px;">**

**<div class="tab-pane fade show active" id="list-dash" role="tabpanel" aria-labelledby="list-dash-list">**

**<div class="container-fluid container-fullw bg-white" >**

**<div class="row">**

**<div class="col-sm-4">**

**<div class="panel panel-white no-radius text-center">**

**<div class="panel-body">**

**<span class="fa-stack fa-2x"> <i class="fa fa-square fa-stack-2x text-primary"></i> <i class="fa fa-list fa-stack-1x fa-inverse"></i> </span>**

**<h4 class="StepTitle" style="margin-top: 5%;"> View Plants</h4>**

**<script>**

**function clickDiv(id) {**

**document.querySelector(id).click();**

**}**

**</script>**

**<p class="links cl-effect-1">**

**<a href="#list-plant" onclick="clickDiv('#list-plant-list')">**

**Plant List**

**</a>**

**</p>**

**</div>**

**</div>**

**</div>**

**<div class="col-sm-4" style="left: -3%">**

**<div class="panel panel-white no-radius text-center">**

**<div class="panel-body">**

**<span class="fa-stack fa-2x"> <i class="fa fa-square fa-stack-2x text-primary"></i> <i class="fa fa-list-ul fa-stack-1x fa-inverse"></i> </span>**

**<h4 class="StepTitle" style="margin-top: 5%;"> View Medicines</h4>**

**<p class="links cl-effect-1">**

**<a href="#list-med" onclick="clickDiv('#list-med-list')">**

**Medicines List**

**</a>**

**</p>**

**</div>**

**</div>**

**</div>**

**<div class="col-sm-4">**

**<div class="panel panel-white no-radius text-center">**

**<div class="panel-body">**

**<span class="fa-stack fa-2x"> <i class="fa fa-square fa-stack-2x text-primary"></i> <i class="fa fa-list-ul fa-stack-1x fa-inverse"></i> </span>**

**<h4 class="StepTitle" style="margin-top: 5%;"> View Methods</h4>**

**<p class="links cl-effect-1">**

**<a href="#list-method" onclick="clickDiv('#list-method-list')">**

**Methods List**

**</a>**

**</p>**

**</div>**

**</div>**

**</div>**

**</div>**

**</div>**

**</div>**

**<div class="tab-pane fade" id="list-plant" role="tabpanel" aria-labelledby="list-plant-list">**

**<table class="table table-hover">**

**<thead>**

**<tr>**

**<th scope="col">Plant Id</th>**

**<th scope="col">Plant Name</th>**

**<th scope="col">Plant Type</th>**

**<th scope="col">Plant Description</th>**

**<th scope="col">Soil Type</th>**

**</tr>**

**</thead>**

**<tbody>**

**<?php**

**$con=mysqli\_connect("localhost","root","","fms");**

**global $con;**

**$query = "select \* from plant;";**

**$result = mysqli\_query($con,$query);**

**while ($row = mysqli\_fetch\_array($result)){**

**?>**

**<tr>**

**<td><?php echo $row['plant\_id'];?></td>**

**<td><?php echo $row['plant\_name'];?></td>**

**<td><?php echo $row['plant\_type'];?></td>**

**<td><?php echo $row['plant\_desc'];?></td>**

**<td><?php echo $row['soil\_type'];?></td>**

**<td>**

**</td>**

**</tr></a>**

**<?php } ?>**

**</tbody>**

**</table>**

**<br>**

**</div>**

**<div class="tab-pane fade" id="list-method" role="tabpanel" aria-labelledby="list-method-list">**

**<table class="table table-hover">**

**<thead>**

**<tr>**

**<th scope="col">Method ID</th>**

**<th scope="col">Method Name</th>**

**<th scope="col">Method Type</th>**

**<th scope="col">Method Description</th>**

**</tr>**

**</thead>**

**<tbody>**

**<?php**

**$con=mysqli\_connect("localhost","root","","fms");**

**global $con;**

**$query = "select \* from method;";**

**$result = mysqli\_query($con,$query);**

**if(!$result){**

**echo mysqli\_error($con);**

**}**

**while ($row = mysqli\_fetch\_array($result)){**

**?>**

**<tr>**

**<td><?php echo $row['method\_id'];?></td>**

**<td><?php echo $row['method\_name'];?></td>**

**<td><?php echo $row['method\_type'];?></td>**

**<td><?php echo $row['method\_desc'];?></td>**

**</tr>**

**<?php }**

**?>**

**</tbody>**

**</table>**

**</div>**

**<div class="tab-pane fade" id="list-med" role="tabpanel" aria-labelledby="list-med-list">**

**<table class="table table-hover">**

**<thead>**

**<tr>**

**<th scope="col">Plant ID</th>**

**<th scope="col">Medicine ID</th>**

**<th scope="col">Medicine Name</th>**

**<th scope="col">Medicine Type</th>**

**<th scope="col">Medicine Cost</th>**

**<th scope="col">Medicine Description</th>**

**</tr>**

**</thead>**

**<tbody>**

**<?php**

**$con=mysqli\_connect("localhost","root","","fms");**

**global $con;**

**$query = "select \* from medicines;";**

**$result = mysqli\_query($con,$query);**

**if(!$result){**

**echo mysqli\_error($con);**

**}**

**while ($row = mysqli\_fetch\_array($result)){**

**?>**

**<tr>**

**<td><?php echo $row['plant\_id'];?></td>**

**<td><?php echo $row['med\_id'];?></td>**

**<td><?php echo $row['med\_name'];?></td>**

**<td><?php echo $row['med\_type'];?></td>**

**<td><?php echo $row['med\_cost'];?></td>**

**<td><?php echo $row['med\_desc'];?></td>**

**</tr>**

**<?php }**

**?>**

**</tbody>**

**</table>**

**</div>**

**</div>**

**</div>**

**</div>**

**<!-- Optional JavaScript -->**

**<!-- jQuery first, then Popper.js, then Bootstrap JS -->**

**<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN" crossorigin="anonymous"></script>**

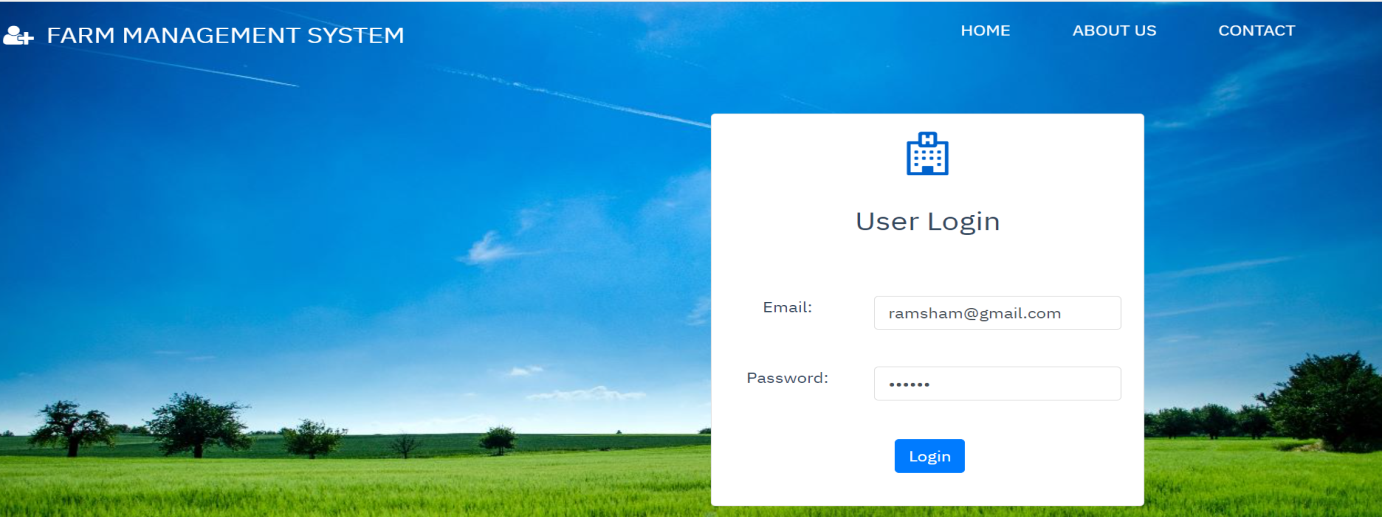
**<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.11.0/umd/popper.min.js" integrity="sha384-b/U6ypiBEHpOf/4+1nzFpr53nxSS+GLCkfwBdFNTxtclqqenISfwAzpKaMNFNmj4" crossorigin="anonymous"></script>**

**<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/js/bootstrap.min.js" integrity="sha384-h0AbiXch4ZDo7tp9hKZ4TsHbi047NrKGLO3SEJAg45jXxnGIfYzk4Si90RDIqNm1" crossorigin="anonymous"></script>**

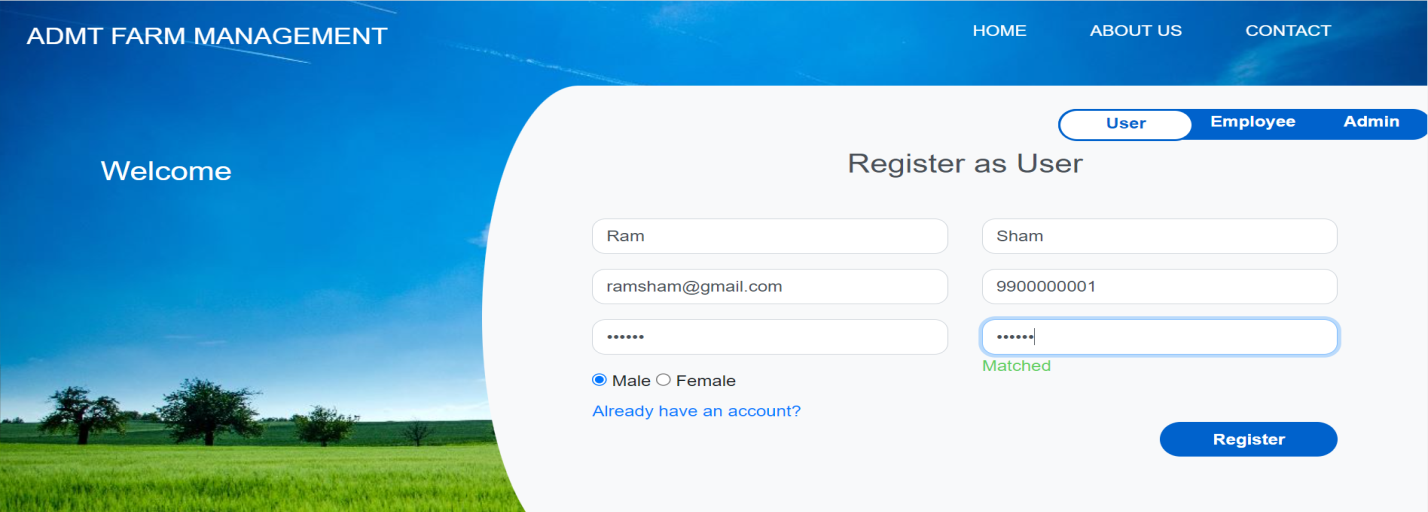
**<script src="https://cdnjs.cloudflare.com/ajax/libs/limonte-sweetalert2/6.10.1/sweetalert2.all.min.js"></script>**

**</body>**

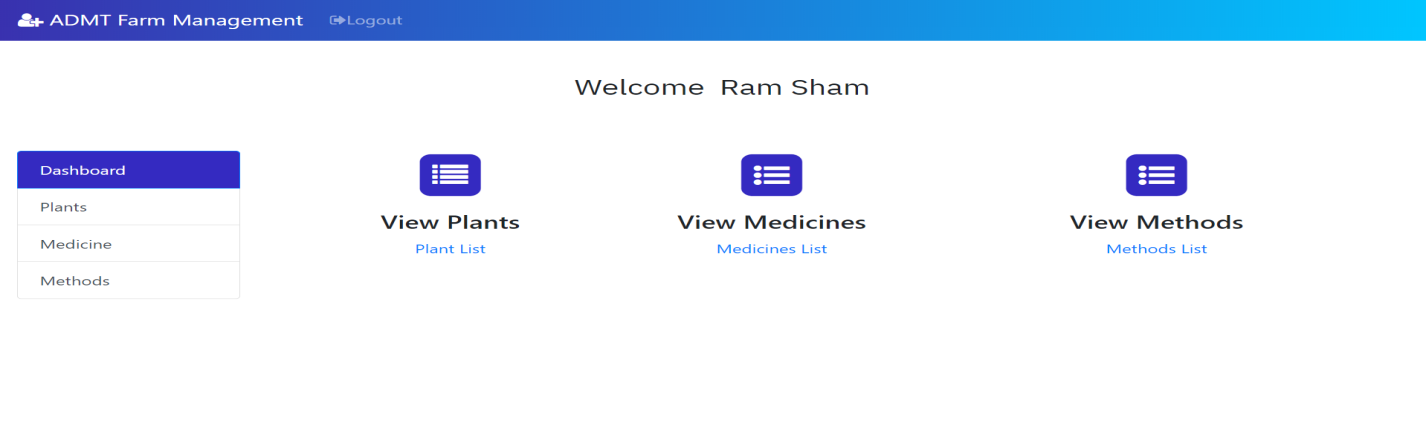
**</html>**

**5 RESULT:**

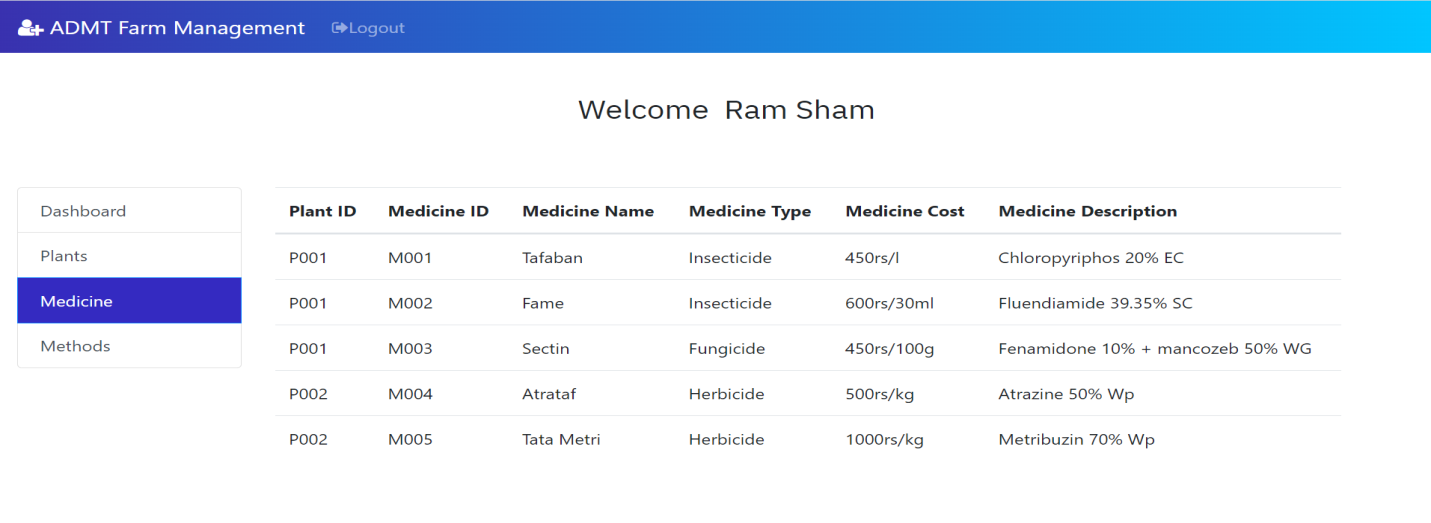
**FIG5.1:User login page**

****

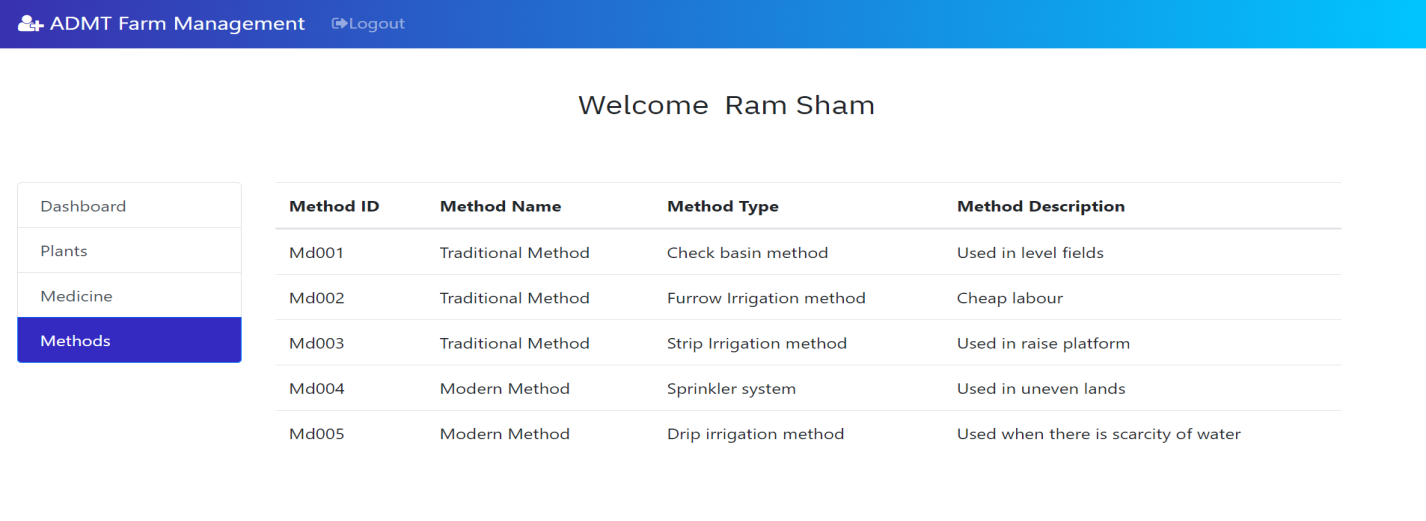
**FIG5.2:User Registeration**

****

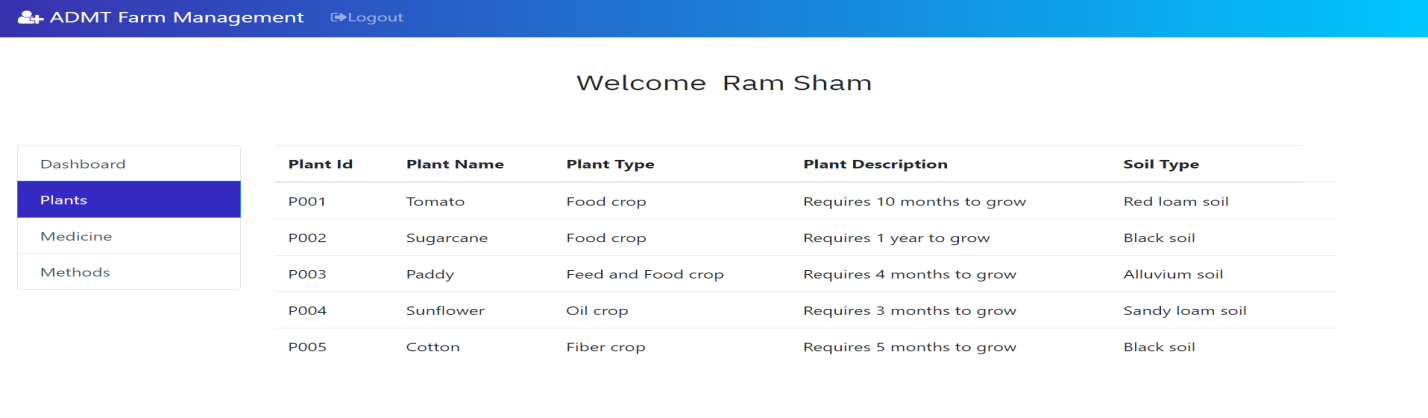
**FIG5.3:User Module**

****

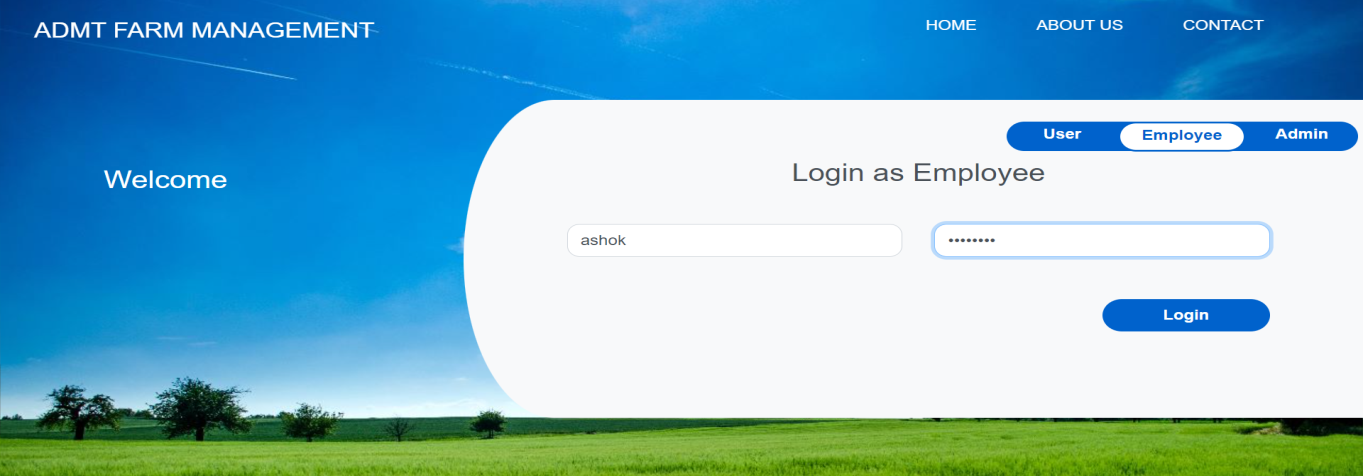
**FIG5.4:User medicine module**

****

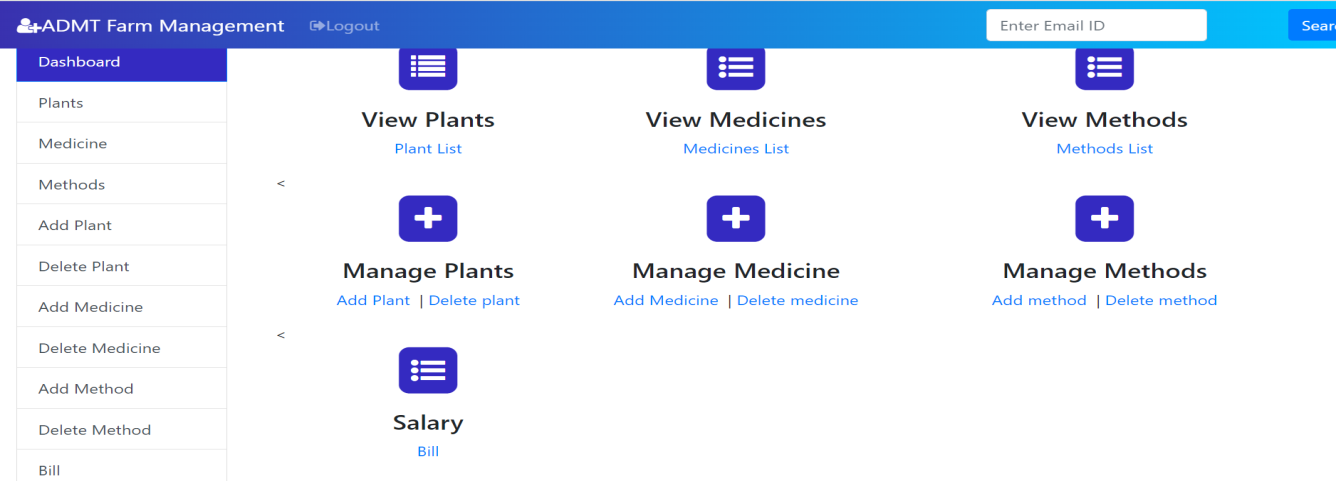
**FIG5.5:User method module**

****

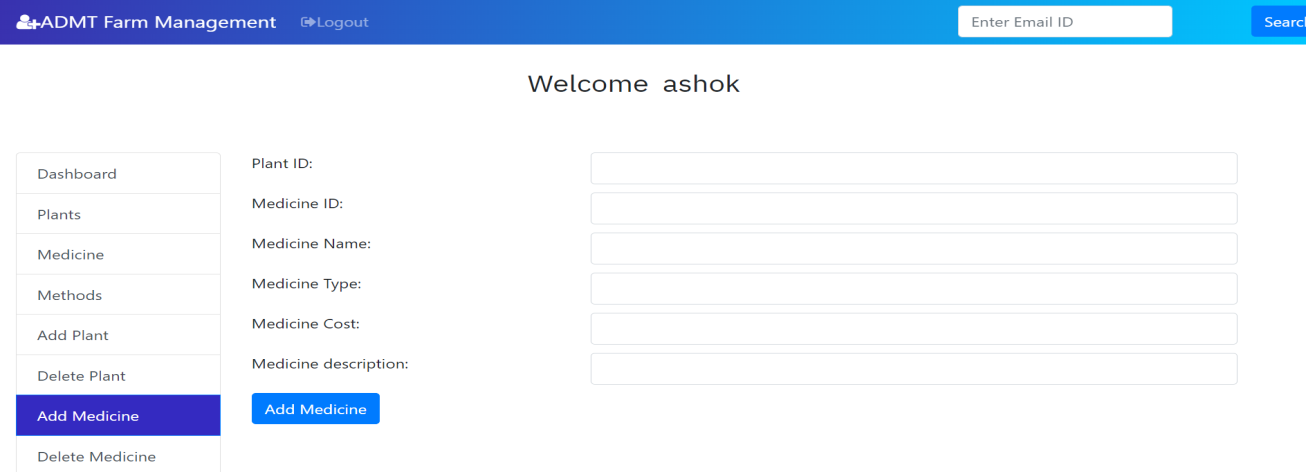
**FIG5.6:User plants module**

****

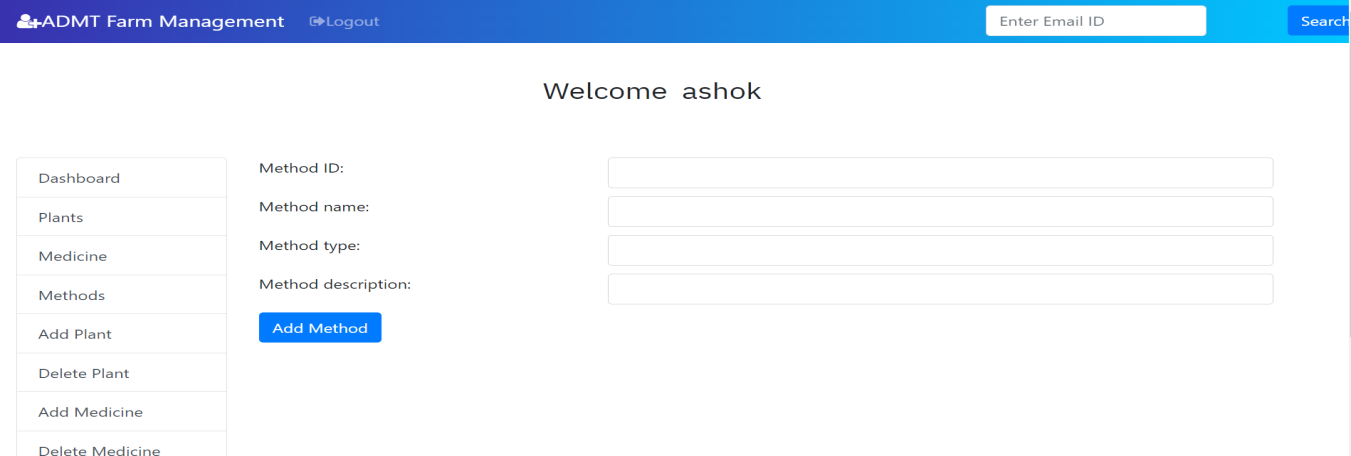
**FIG5.7:Employee Login page**

****

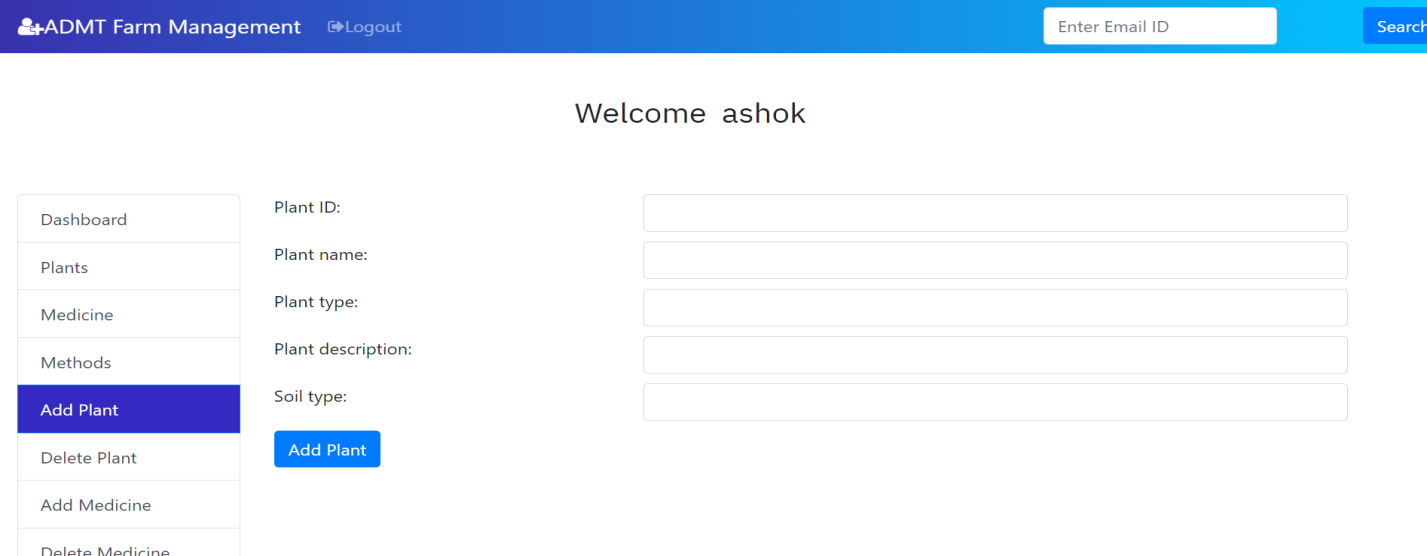
**FIG5.8:Employee Module**

****

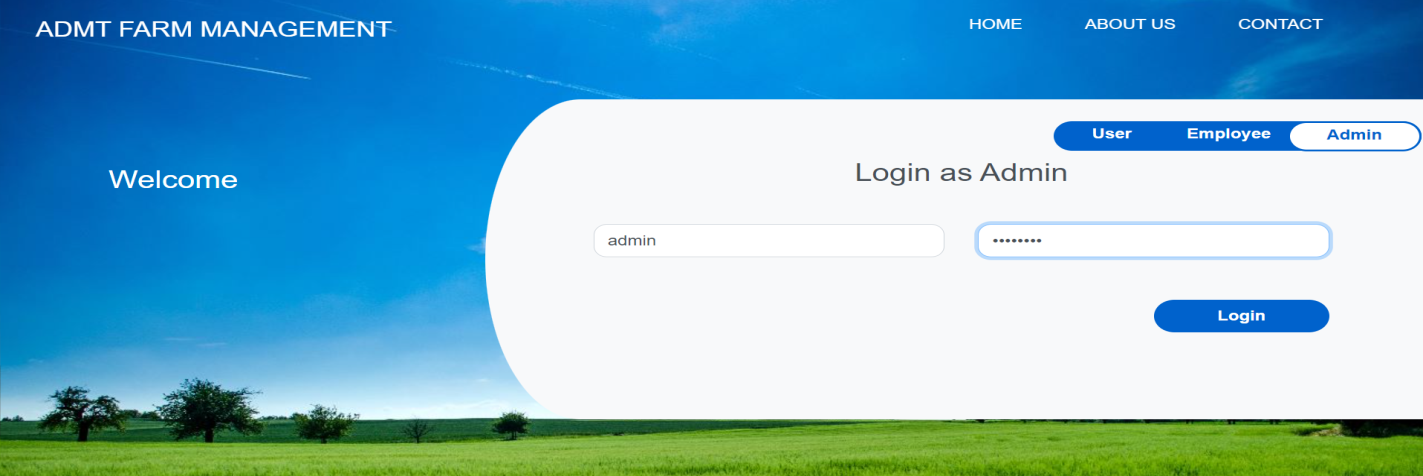
**FIG5.9:Employee-Add Medicine module**

****

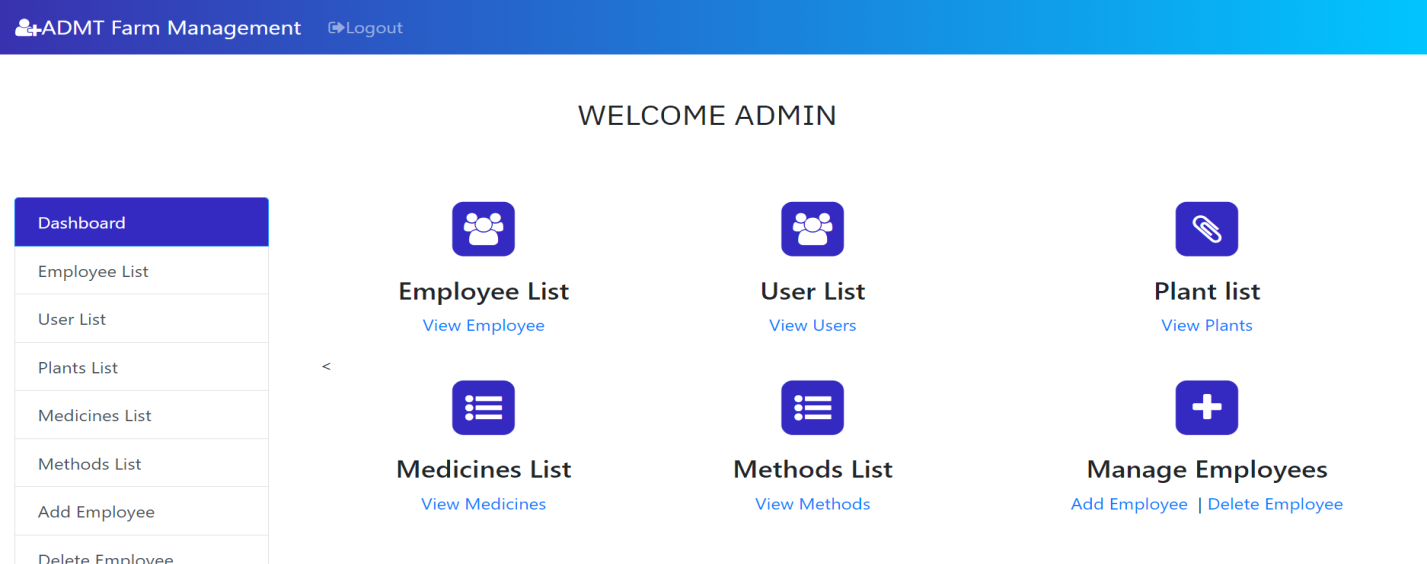
**FIG5.10:Employee-Add Method module**

****

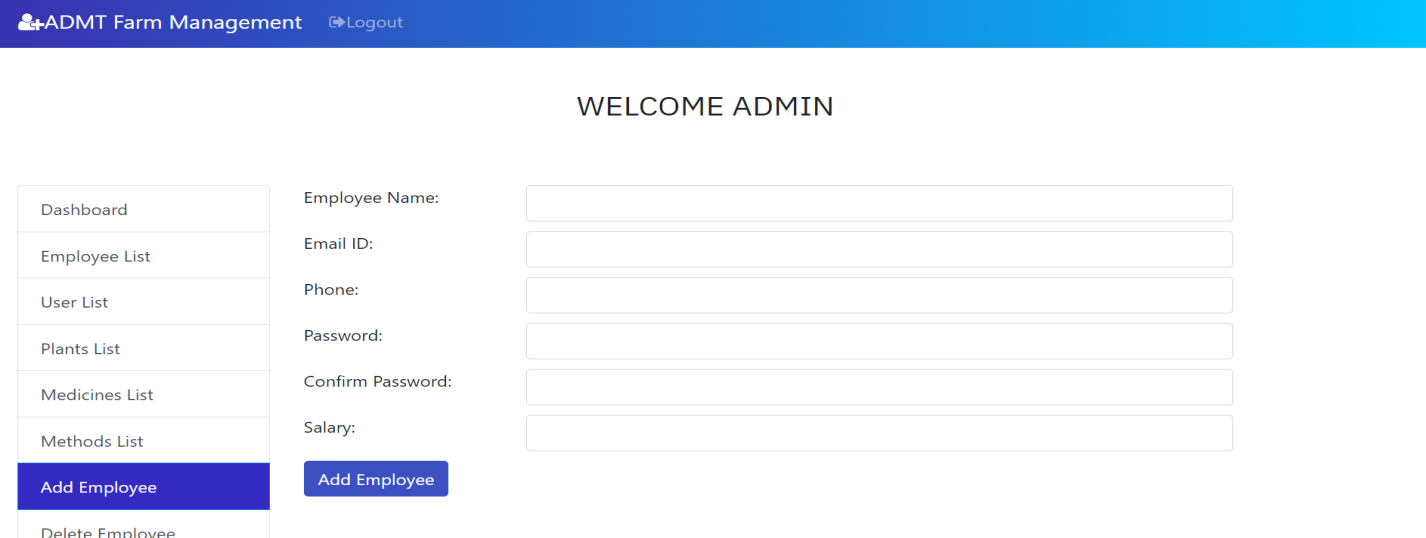
**FIG5.11:Employee-Add Plants module**

****

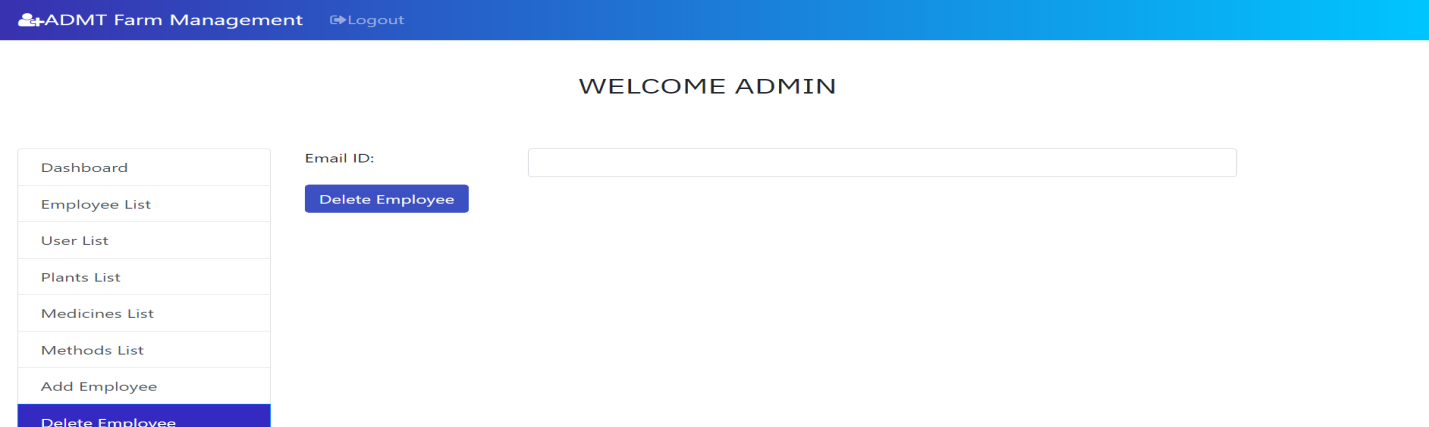
**FIG5.12:Admin login page**

****

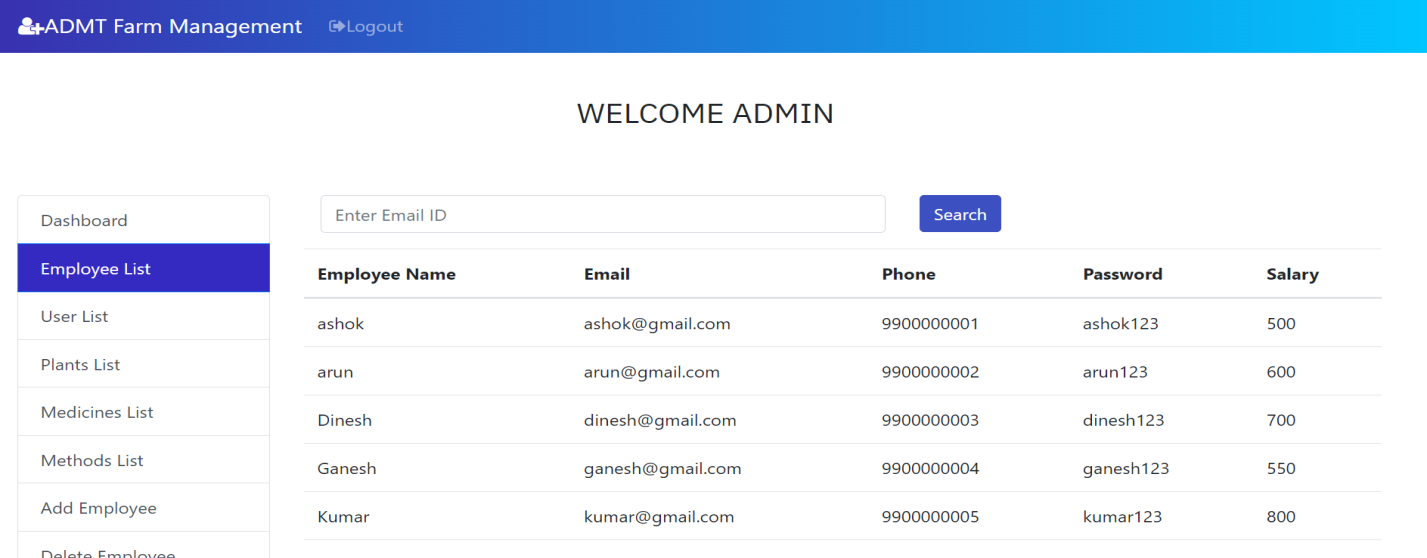
**FIG5.13:Admin Module**

****

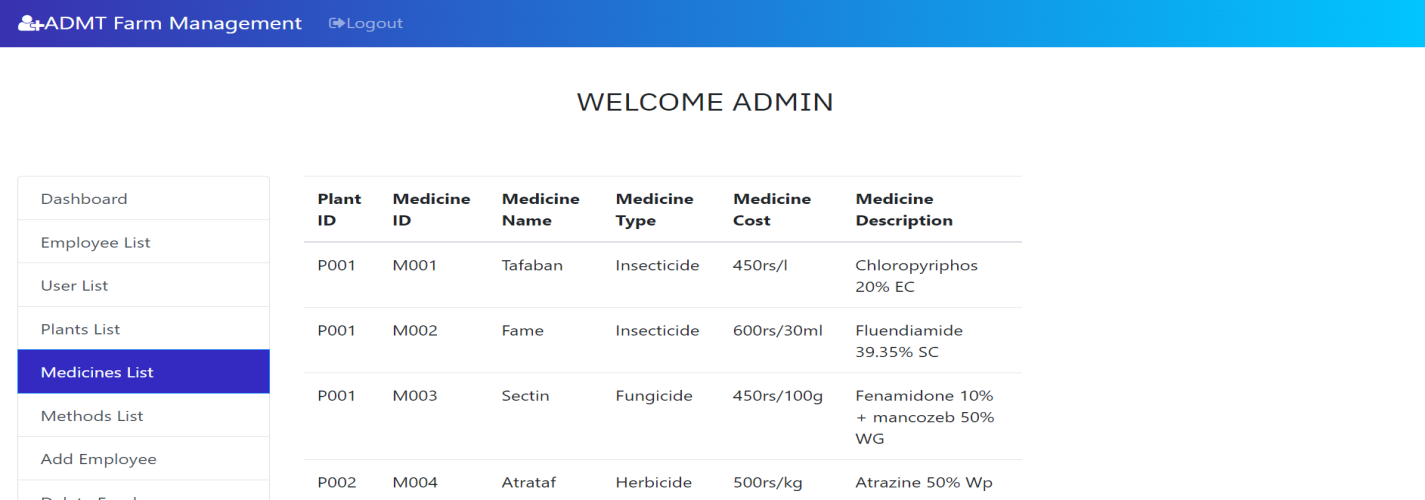
**FIG5.14:Admin-Add Employee module**

****

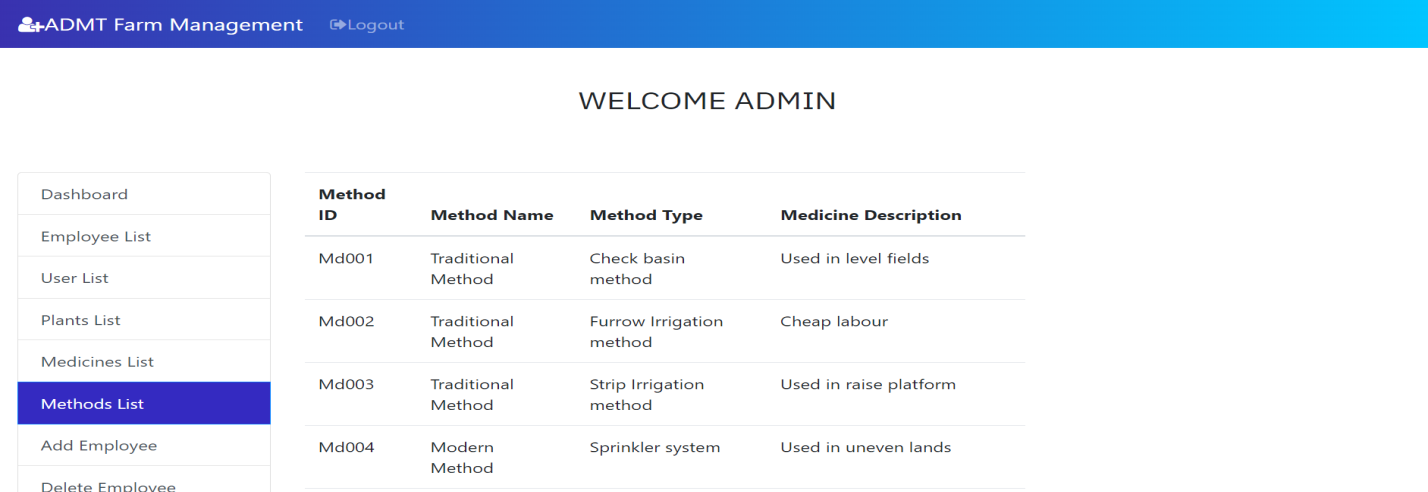
**FIG5.15:Admin-Delete Employee module**

****

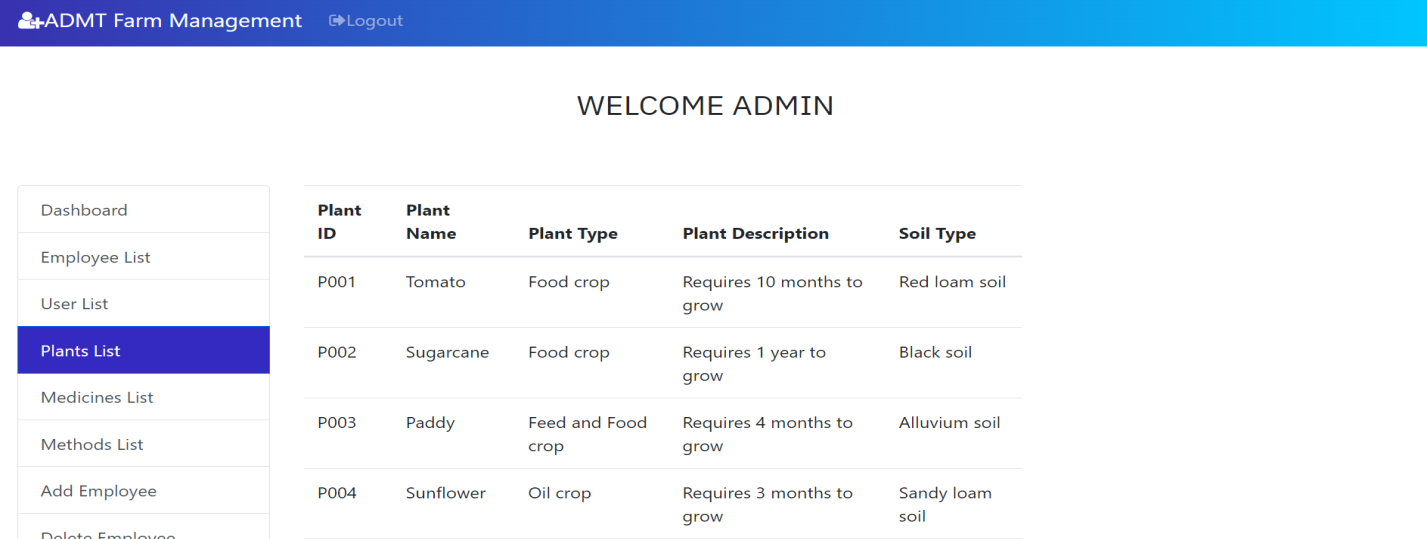
**FIG5.16:Admin-view Employee list**

****

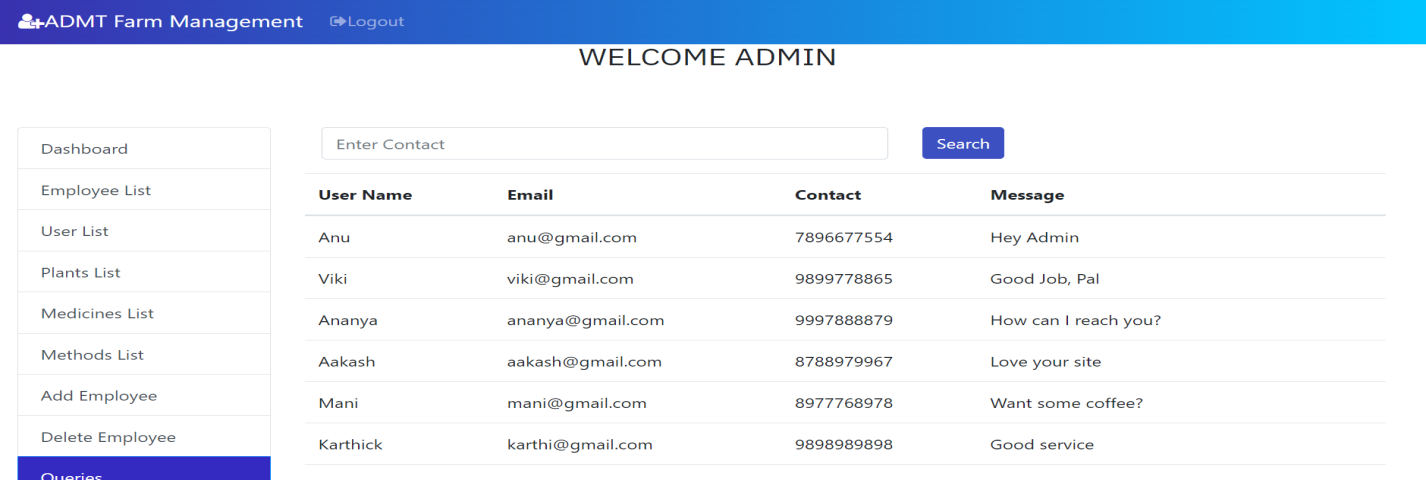
**FIG5.17:Admin-view Medicine list**

****

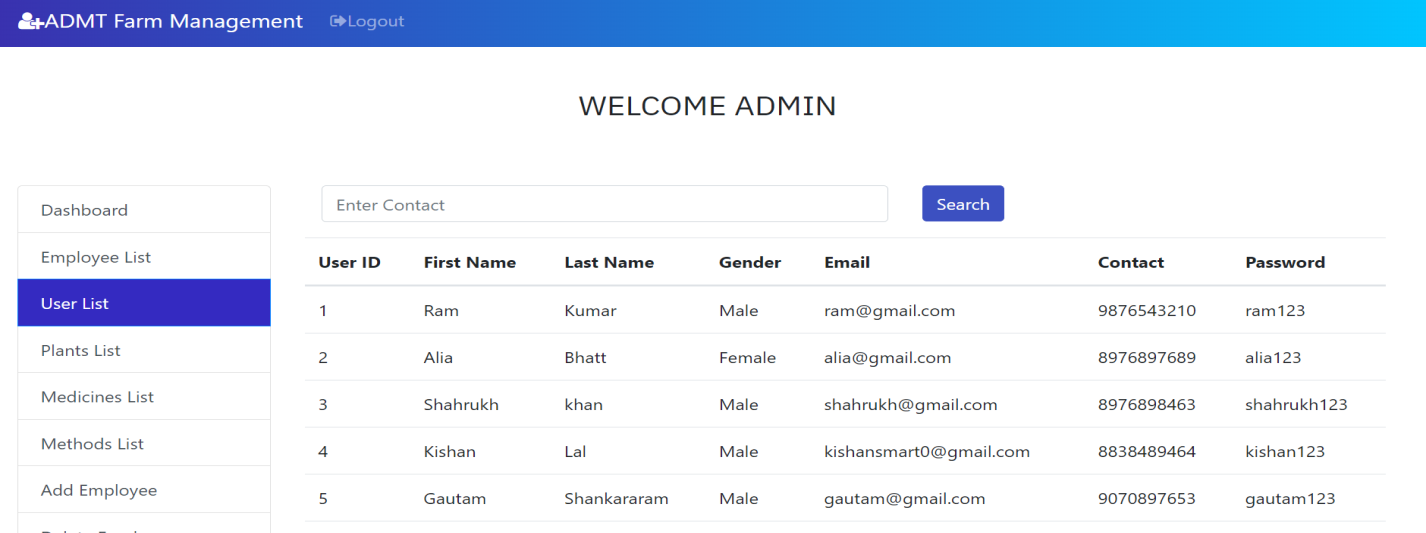
**FIG5.18:Admin-view Method list**

****

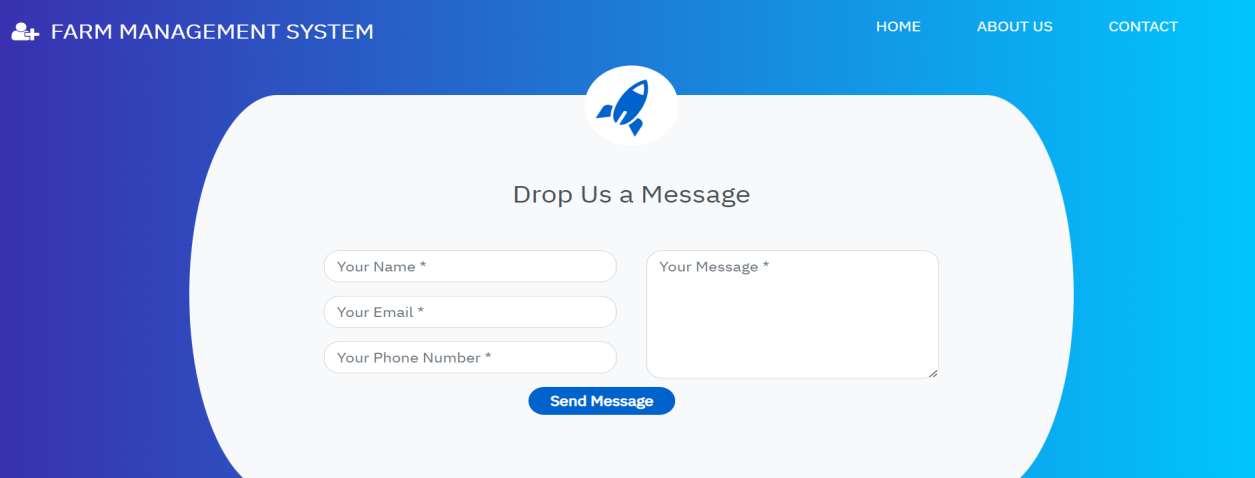
**FIG5.19:Admin-view Plants list**

****

**FIG5.20:Admin-view Feedbacks**

****

**FIG5.21:Admin-view User list**

****

**FIG5.22:User Feedback page**

**6.CONCLUSION**

In conclusion, the Farm Management System (FMS) developed within this project offers a comprehensive solution for streamlining operations within the agricultural domain. By leveraging a robust database management system, the FMS provides distinct functionalities tailored to the needs of customers, employees, and administrators.

Customers can seamlessly access plant and medicine details, facilitating informed decision-making regarding purchases and treatments. Through a user-friendly interface, they can explore a diverse range of products, enhancing their overall experience and satisfaction.

Employees are empowered with the tools necessary for efficient plant and medicine management. With the ability to add, delete, and modify details, they can ensure the accuracy and relevance of the database, thereby contributing to operational efficiency and customer satisfaction.

Administrators wield overarching control over the system, with capabilities to manage employee rosters and oversee database integrity. This administrative oversight ensures compliance with organizational standards and facilitates strategic decision-making.

Overall, the Farm Management System represents a significant advancement in agricultural technology, offering a scalable and adaptable solution for businesses of varying sizes. Its integration of customer-centric features, employee management functionalities, and administrative controls positions it as a valuable asset in optimizing farm operations and driving sustainable growth.

As the agricultural landscape continues to evolve, the Farm Management System stands poised to meet the challenges and opportunities of tomorrow, serving as a cornerstone in the quest for agricultural innovation and efficiency

**7.REFERENCES**

**"Farm Management Information Systems**: A Case Study on a German Large-scale Farm"

This paper provides insights into the implementation and use of farm management information systems, offering a real-world case study.

<https://www.researchgate.net/publication/316753826_Investigations_of_Feedback_Schedules_on_Speech_Motor_Learning>

**"Farm Management Systems: Technology Trends and Future Directions"**

This article discusses emerging technologies and trends in farm management systems, providing valuable information for the design and development of your FMS.

<https://ieeexplore.ieee.org/document/9276827>

**"Smart Farming: Including the Farm Management System in the Internet of Things"**

This paper explores the integration of farm management systems into the Internet of Things (IoT) ecosystem, highlighting the benefits and challenges.

<https://www.mdpi.com/>

These references should provide you with valuable insights and information specific to farm management systems, aiding in the development and documentation of our project.