SmartFarm with Plant disease perdition

Abstract:

The SmartFarm project aims to revolutionize traditional farming practices by introducing a smart and automated system that addresses various challenges faced by farmers. The system employs cutting-edge technologies, including Django for web development and Python for machine learning, to provide farmers with expert opinions, access to farming products, and a platform to market their products to a wider customer base. The proposed system includes four main entities: User, Farmer, Expert, and Admin, each with specific roles and functionalities. Through seamless integration of modules, farmers can register, manage their farms, seek advice from experts, list and sell products, while users can conveniently purchase organic products directly from farmers. The key features of the system include farmer registration and login, image upload for disease prediction, recommendations and remedies, an interactive frontend, essential library integration, machine learning models for disease prediction, and automated communication to enhance farmer engagement.

Module Description:

- User Module:
    - Registration and Login: Users can register with their credentials and confirm their status as customers upon prompt. Secure authentication ensures data confidentiality.
    - Product Viewing and Purchase: Users can view products listed by farmers and make purchases directly through the platform.

- Farmer Module:

- Registration and Login: Farmers can register and log in securely with their credentials.
- Farm Management: Farmers can add and update farm details, view farm status, and ask questions to experts.
- Product Listing: Farmers can list their farmed products in the database, view bookings by customers, and update booking statuses.
- Expert Interaction: Farmers can receive advice from experts, view farm status, and receive instructions for farm improvement.

- Expert Module:
  - Login: Experts can securely log in to the system.
  - Farm Monitoring: Experts can monitor farmers' farm statuses, provide guidance, and update information on the platform.
  - Product Listing: Experts can list recommended products for farmers, enhancing collaboration.
  - Booking Management: Experts can view bookings made by farmers, update statuses, and provide feedback.

- Admin Module:
  - Expert Management: Admins can create and manage expert accounts.
  - Database Access: Admins have access to all database tables for efficient management.
  - CRUD Operations: Admins can perform create, read, update, and delete operations on all database tables.

System Requirements:

- Programming Languages: Python, HTML, CSS, JavaScript
- Web Framework: Django
- Machine Learning Framework: PyTorch

- Libraries: OpenCV, Pandas, Joblib, NumPy, Torch
- Database: Utilizes a relational database (not specified in the provided information)
- Web Server: Django Integrated Server (for development) or a production-ready server (e.g., Apache, Nginx)
- Operating System: Compatible with various operating systems (Windows, Linux, macOS)
- Security: Implements secure authentication mechanisms for user and farmer login
- User Interface: An interactive frontend developed using HTML, CSS, and JavaScript
- Communication: Automated notifications for farmers containing disease predictions, recommendations, and suggested remedies.
- Scalability: The system is designed to scale with increasing user and data loads.