Project Proposal

An AI-Powered Web Application for Plant Disease Detection and Cure Recommendation

An Intelligent Solution to Identify Plant Diseases and Recommend Effective Remedies

Submitted by

Abdullah Javeed (F22BINFT1M01076)

Submitted to

Dr. Syed Ali Nawaz Shah

Department of Information Technology

Faculty of Computing

The Islamia University of Bahawalpur

Title: “CultiKure” An AI-Powered Web Application for Plant Disease Detection and Cure Recommendation

➢ Introduction:

Agriculture is a cornerstone of Pakistan's economy, yet it faces constant threats from plant diseases that cause significant economic losses. Early and accurate disease identification is critical, but traditional methods are often slow and inaccessible to many farmers. "CultiKure" is a project designed to solve this problem by creating a comprehensive, centralized **web-based platform**. This application will be accessible from any device with a web browser, including desktops, laptops, and mobile phones. By leveraging a powerful dual-API strategy, the system will not only provide instant plant disease diagnosis from an uploaded image but will also offer detailed causes, symptoms, and treatment recommendations from a secondary AI.

➢ Objective:

To design an instinctive, secure, and universally accessible **web application** that connects farmers and gardeners with AI-driven diagnostic tools for effective plant health management and the improvement of community agricultural output.

➢ Goals:

1. **Simplify the Diagnosis Process:** To facilitate a seamless process for users to get an instant diagnosis by simply uploading an image of a plant leaf.
2. **Provide Actionable Advice:** To offer immediate, AI-generated information regarding the causes, symptoms, and treatment options (both organic and chemical) for any detected disease.
3. **Enhance User Access:** To ensure the platform is accessible to all users via a single, responsive **web application** that works on both desktop and mobile browsers.
4. **Raise Awareness:** To educate users on the importance of plant health and disease prevention through an integrated knowledge base.

➢ Key Features:

- **User Registration and Dashboard**
  - **User Profile:** A secure sign-up and login system for all users. The user's personal dashboard will manage their profile and scan history.
  - **Scan History:** A personal log where all previous diagnostic scans, including the image and the results, are saved for future reference.
- **AI-Powered Disease Detection**
  - **Image Upload:** A user-friendly interface to upload a leaf image directly from their computer or mobile device's camera/gallery.
  - **Instant Diagnosis:** The system will integrate with a specialized, pre-trained third-party AI API (e.g., Plant.id) to analyze the image and return an accurate disease name and confidence score.

- **AI-Powered Cure and Info Assistant**
    - **Detailed Information:** Upon successful detection, the system will automatically query a second, powerful generative AI API (e.g., Google Gemini).
    - **Cure Recommendation:** This AI assistant will instantly provide comprehensive details about the disease, including its common causes, symptoms, and a list of recommended organic and chemical treatment options.
- **Responsive Web-Based Access**
    - **Mobile-Friendly Design:** The web application will be built using responsive design principles, ensuring it is perfectly usable and looks professional on all screen sizes, from large desktop monitors to small mobile phones.
- **Additional Features**
    - **Educational Resources:** An integrated library of articles and guides on common plant diseases, prevention tips, and best farming practices.
    - **Social Sharing:** An option for users to share their diagnosis results and the provided treatment plan.

➢ Problem Statement:

Plant diseases pose a significant threat to agricultural productivity, causing substantial financial losses annually. The current methods for disease identification often rely on manual, in-person expert analysis, which is slow, costly, subjective, and not scalable for most farmers, especially those in remote areas. This critical delay between symptom appearance and accurate diagnosis prevents timely treatment. This project is focused on solving this problem by providing an **online web application** through which an instant and accurate diagnosis can be obtained. By leveraging a dual-API strategy, the system not only identifies the problem but also provides an immediate, actionable solution, creating a complete end-to-end tool for the user.

➢ Target Audience:

- **Farmers:** Small and large-scale farmers seeking to improve crop health and yield.
- **Home Gardeners:** Hobbyists and individuals maintaining personal gardens.
- **Agricultural Students:** For research, learning, and educational purposes.
- **Community Organizations:** Organizations promoting agricultural awareness.

➢ Project Scope:

The scope of this project includes:

- **User Registration and Authentication:** Secure sign-up and login for the web application.
- **Dual AI API Integration:** Integration of a specialized image recognition API (e.g., Plant.id) for diagnosis and a generative AI API (e.g., Google Gemini) for cure information.
- **Web Application Development:** Development of a single, responsive web application using a Python-based backend (Flask) and a modern frontend (HTML/CSS/JavaScript).

- **Data Management:** A secure database to manage user profiles and diagnostic scan history.

➢ Methodology:

The project will be executed using the **Agile Development Methodology**, allowing for iterative development and flexibility. The workflow will be broken down into sprints, focusing on delivering specific functionalities in each cycle.

- **Major Phases:**
    1. **Phase 1: Planning and System Design:** This initial phase involves finalizing requirements, defining the system architecture, designing the database schema, and creating UI/UX wireframes for the web application.
    2. **Phase 2: Backend Development and Core Logic:** The focus will be on building the central RESTful API using **Python (Flask)**. This includes user authentication endpoints and the core logic for integrating with the two external AI APIs (Plant.id and Google Gemini).
    3. **Phase 3: Frontend Development:** Development of the responsive user interface for the web application using **HTML, CSS, and JavaScript**. This frontend will communicate with the backend API to fetch and display data.
    4. **Phase 4: Integration, Testing, and Feedback:** All components (frontend, backend, database) will be integrated. Rigorous testing (Unit, Integration, and User Acceptance Testing) will be conducted to ensure quality and performance.
    5. **Phase 5: Deployment and Final Documentation:** The complete web application will be deployed to a cloud hosting service (e.g., Heroku, PythonAnywhere, etc) to be publicly accessible. Final project documentation will be completed.

➢ Technical Specifications:

Platform: Web-based application accessible via all modern desktop and mobile browsers.

Technology Stack:

- **Frontend (Web):** HTML5, CSS3, JavaScript
- **Backend:** Python (using the Flask Framework)
- **Database:** Firebase Firestore or PostgreSQL
- **AI APIs:** Plant.id Health Assessment API, Google Gemini API
- **Hosting:** Cloud-based platforms like Heroku, PythonAnywhere, AWS, or Azure **etc.**
- **Security:** Implementation of SSL encryption and secure, token-based authentication methods for API endpoints.

➢ Expected Outcome

The primary outcomes will be:

1. **User Satisfaction:** Feedback from users regarding the web application's ease of use, speed, and accuracy of information.
2. **Response Time:** The total time taken from image upload to the final display of diagnosis and cure recommendations.
3. **System Reliability:** The performance and uptime of the deployed web application under various user loads.

The system's success will be evaluated based on the following measurable criteria:

1. **Functionality:** All key features, from user registration to the dual-API diagnosis process, operate without errors.
2. **Performance:** The average response time from image upload to displaying the final result (including cure information) should be under 10 seconds.
3. **Usability:** User feedback will be collected to assess the ease of use and overall satisfaction with the platform. A target satisfaction score of over 80% will be aimed for.
4. **Reliability:** The deployed application will maintain a target uptime of 99%.

➢ Project Timeline

This is expected to take six months and the major milestones would include:

Month 1: Foundation and Planning

Month 2: Backend Development and Core

Month 3: Web Application Frontend (UI/UX)

Month 4: Frontend-Backend Integration

Month 5: Testing, Integration, and Refinement

Month 6: Final Deployment and Project Delivery

Conclusion:

The "CultiKure" platform represents a strategic and modern approach to solving a critical agricultural problem. By leveraging an "API-first" methodology, the project prioritizes the rapid delivery of a complete and practical **web application**. The innovation lies in its dual-API workflow, which creates a seamless "diagnosis-to-remedy" experience. This platform will serve as an accessible and powerful tool for farmers and gardeners, demonstrating the real-world value of AI in enhancing food security and promoting sustainable agriculture.