SVKM’s NMIMS

Mukesh Patel School of Technology Management & Engineering

**Project Report**

**On**

**Agromate: AI powered disease detection software**

**Sub: Artificial Intelligence**

Faculty Mentor: Submitted by:

Dr. Surendra Kumar Shukla 1. Jayaditya Harish Arora (B.Tech CS (E284))

Assistant Professor 2.

**Department of Computer Science**

**2024-25**

## Table of Contents

| Sr No | Table Contents | Page No |
| --- | --- | --- |
| 1 | Introduction | 3 |
| 2 | Project Overview | 3-4 |
| 3 | Features of the Project | 4 |
| 4 | System Architecture and Workflow | 4-5 |
| 5 | Installation & Setup | 5-6 |
| 6 | Challenges & Solution | 6 |
| 7 | Future Enhancement | 6 |
| 8 | Conclusion | 6 |
| 9 | References | 7 |

## 1. Introduction

AgroMate is an AI-powered platform designed to empower farmers with real-time agricultural insights, weather data, and disease detection capabilities. The project integrates advanced technologies like computer vision, natural language processing (NLP), and data analytics to provide actionable recommendations for crop management.

### 1.1 Motivation

Modern farming faces challenges such as unpredictable weather, pest outbreaks, and lack of expert advice. AgroMate addresses these issues by offering a one-stop solution for farmers to monitor environmental conditions, diagnose crop diseases, and receive expert guidance via an AI chatbot.

### 1.2 Objectives

* Provide real-time weather and pollution data to farmers.
* Enable AI-driven disease detection for crops.
* Offer multilingual chatbot support for agricultural advice.
* Ensure accessibility and usability for farmers in diverse regions.

## 2. Project Overview

AgroMate is a full-stack web application combining frontend dashboards with backend AI models. It serves as a decision-support tool for farmers, helping them optimize crop health and yield.

### 2.1 Technologies Used

| Technology | Purpose |
| --- | --- |
| HTML/CSS/JavaScript | Frontend dashboard and UI design |
| TensorFlow/Keras | Computer vision models for disease detection |
| OpenAI GPT-3.5 Turbo | NLP for the Agri Expert Chatbot |
| Node.js | Backend server and API integration |
| Firebase/Cloud Services | Data storage and real-time updates |

## 3. Features of the Project

### 3.1 Real-Time Interactive Dashboard

* Displays live weather (temperature, humidity, rainfall) and air quality (PM2.5, CO, NO₂) data.
* Visualizes trends using line graphs, bar charts, and maps.

### 3.2 AI-Powered Disease Detection

* Uses 9 trained computer vision models to detect 21 diseases across 9 crops.
* Farmers upload plant images for instant diagnosis and treatment suggestions.

### 3.3 Agri Expert Chatbot

* Multilingual support for farmers to ask questions about crop diseases, treatments, and best practices.
* Integrates with image uploads for contextual advice.

### 3.4 Exportable Reports

* Farmers can download weather/disease reports as PDFs for record-keeping.

## 4. System Architecture and Workflow

### 4.1 System Architecture

1. Frontend:
   * Dashboard for data visualization (built with React/JavaScript).
   * Image upload interface for disease detection.
   * Chatbot UI for interactive Q&A.
2. Backend:
   * Node.js server handling API requests.
   * TensorFlow models for image classification.
   * Firebase for real-time data storage.
3. AI Components:
   * Hybrid model (CNN + Decision Tree) for disease detection.
   * GPT-3.5 Turbo for chatbot responses.

### 4.2 Workflow

1. Farmer logs in and views real-time dashboard.
2. Uploads a crop image for disease diagnosis.
3. Chatbot provides treatment advice or answers follow-up questions.
4. Data is saved for future reference.

## 5. Installation and Setup

### 5.1 Prerequisites

* Node.js, npm, Python 3.x, TensorFlow.
* Modern web browser (Chrome/Firefox).

### 5.2 Steps to Run Locally

1. Clone the repository:
2. bash
3. Copy

git clone https://github.com/HareKrishna4747/AgroMate.git

1. cd AgroMate
2. Install dependencies:
3. bash
4. Copy

npm install

1. pip install -r requirements.txt
2. Start the backend server:
3. bash
4. Copy
5. node server.js
6. Launch the frontend:
7. bash
8. Copy
9. npm start
10. Access the app at http://localhost:3000.

## 6. Challenges and Solutions

| Challenge | Solution |
| --- | --- |
| Real-time data synchronization | Used Firebase for live updates. |
| Multilingual chatbot support | Integrated GPT-3.5 Turbo with prompt engineering. |
| Disease detection accuracy | Hybrid CNN + Decision Tree model. |

## 7. Future Enhancements

1. Mobile App Development for offline use in rural areas.
2. IoT Integration with soil sensors for granular data.
3. Expanded Crop Library to cover more crops and diseases.

## 8. Conclusion

AgroMate bridges the gap between traditional farming and modern AI, offering farmers a scalable, user-friendly tool to combat agricultural challenges. Its real-time insights and AI-driven diagnostics can significantly improve crop management and productivity.

## 9. References

* TensorFlow Documentation: <https://www.tensorflow.org/>
* OpenAI API: <https://openai.com/>
* Firebase: <https://firebase.google.com/>

Note: Replace placeholders like [Your Name] and GitHub links with actual details before submission.