|  |
| --- |
| Software Requirement Specification |
|  |

**Agro Vision**

Farmer’s Companion Application

***Team Members :-***

Balendhu S Ajay 20 bsacreations@gmail.com

Kevin Tom Varghese 36 kevintomishere@gmail.com

Philip John 56 rahulphilipjoh@gmail.com

Pranav Krishna P 57 fifapranav3404@gmail.com

Contents

1. Introduction
   1. Purpose of the SRS
   2. Scope of the Software
   3. Definitions, Acronyms, and Abbreviations
   4. Target Audience
2. Overall Description
   1. Product Perspective
   2. Product Functions
   3. User Characteristics
   4. Constraints
   5. Assumptions and Dependencies
3. Specific Requirements

3.1 External Interface Requirements

3.1.1. User Interfaces

3.1.2 Hardware Interfaces

3.1.3 Software Interfaces

3.1.4 Communications Interfaces

3.2 Functional Requirements

3.2.1 Crop Disease Detection

3.2.2 Crop Management tools

3.2.3 Chatbot

3.2.4 Locating nearby shops providing farming supplements

3.3 Non-functional Requirements

3.3.1 Performance Requirements

3.3.2 Security Requirements

3.3.3 Maintainability Requirements

3.3.4 Reliability Requirements

1. Acceptance Criteria
   1. Crop Scanning
   2. Identify Growth Rates
   3. Scalability
   4. User-friendly
   5. Secure
2. Introduction
   1. Purpose of the SRS

The purpose of this document is to define the requirements for the development of a software system that scans crops, detects crop diseases, recommends pesticides to cure crop diseases based on the disease detected, identifies crop growth patterns based on weather conditions, and suggests crops that can be grown based on types of soil. This system will use machine learning and image recognition techniques to perform the required functionalities.

* 1. Scope of the Software

The scope of the system includes the development of a software application that will be used by farmers and agricultural experts to scan crops, detect crop diseases, recommend pesticides, identify crop growth patterns, and suggest crops that can be grown based on types of soil. The system will utilize machine learning and image recognition algorithms to perform the functionalities.

* 1. Definitions, Acronyms, and Abbreviations

SRS: Software Requirements Specification

ML: Machine Learning

AI: Artificial Intelligence

CNN: Convolutional Neural Network

GUI: Graphical User Interface

* 1. Target Audience

The main consumer base consists of: -

1. Farmers
2. Agricultural researchers
3. Plant pathologists
4. Agricultural extension agents
5. Companies that produce pesticides
6. Overall Description
   1. Product Perspective

The software system is a standalone application that will be used by farmers and agricultural experts to scan crops, detect crop diseases, recommend pesticides, identify crop growth patterns, and suggest crops that can be grown based on types of soil.

* 1. Product Features
* The software system will include the following features:
* Crop scanning using image recognition.
* Crop disease detection using machine learning algorithms.
* User-friendly GUI for ease of use.
  1. User Classes and Characteristics

The users of the system are farmers and agricultural experts who have basic knowledge of computer systems and mobile devices.

* 1. Constraints

The software system will be constrained by the following factors:

* The accuracy of crop disease detection and pesticide recommendation is limited by the quality of the images provided by the user.
* The software system will require significant computational resources for machine learning and image recognition.
  1. Assumptions and Dependencies

The following assumptions have been made during the development of the system:

* The mobile device or personal computer will have an internet connection.
* The user will be able to capture clear and accurate images of the crops.

1. Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interface

The software system will have a user-friendly interface that allows users to:

* Upload images of crops for analysis
* View analysis results, including disease detection, pesticide recommendations, growth patterns, and suggested crops.

3.1.2 Hardware Interface

The software system should support the following hardware interface requirements:

* Cameras or mobile devices: The system should be able to capture images of crops using a camera or mobile device.
* Storage devices: The system should be able to store the captured images and other data on storage devices such as hard disks, USB drives, or cloud storage.
* Network devices: The system should be able to communicate with other devices and systems over wired or wireless networks.

3.1.3 Software Interface

The software system should support the following software interface requirements:

* Operating system: The software should be compatible with different operating systems, such as Windows, MacOS, and Linux.
* Programming language: The system should be developed using programming languages suitable for machine learning and image recognition, such as Python and MATLAB.
* Frameworks: The system should use machine learning frameworks, such as TensorFlow or Keras, for building and training the models.
* Database: The system should use a database to store information about crops, diseases, and pesticides.
* Image processing libraries: The system should use image processing libraries, such as OpenCV, to process the images and extract relevant features.

3.1.4 Communication Interface

The software system should support the following communication interface requirements:

* APIs: The system should provide APIs to allow other systems to access its functionalities, such as crop disease detection and pesticide recommendation.
* Web interface: The system should provide a web interface to allow users to access its functionalities through a web browser.
* Network protocols: The system should use standard network protocols, such as HTTP or MQTT, to communicate with other devices and systems over the network.

3.2 Functional Requirements

3.2.1. Crop Scanning and Disease Detection

* The software shall be able to scan crops using image recognition technology to detect any diseases or abnormalities.
* The software shall be able to identify the type of crop and the specific disease based on the analysis of the scanned image.
* The software shall provide a report on the severity of the disease detected, including the percentage of the crop affected.

3.2.2. Crop Management tool

* This functionality provides a suite of management tools, including irrigation scheduling, pest management strategies, weather patterns, and fertilization recommendations.
* Farmers can view suggestions regarding time of irrigation, harvesting and sowing, based on weather conditions.
* They can set process/stage specific alarms or reminders as and when required.
* This tool does not make use of IoT.

3.2.3. Chat-bot to answer crop related queries

* The software shall be able to enhance customer experience and engagement by providing 24/7 assistance, quick answers to simple questions, and reducing the need for human intervention.
* Thus helping farmers make informed decisions, increase crop productivity, and prevent financial loss.

3.2.4. Locating nearby shops providing farming supplements

* The system should have a geolocation feature that allows users to locate nearby shops that sell crop disease treatment products, such as pesticides or fungicides, based on their current location.
* The system should provide in-app navigation to guide users to the nearest shop that sells the recommended product.

3.3 Non-Functional Requirements

3.3.1. Performance

* The software shall be able to process the scanned images and provide recommendations within a reasonable time frame.
* The software shall be able to handle a large volume of data without any performance issues.

3.3.2. Accuracy

* The software shall have a high level of accuracy in detecting crop diseases and identifying suitable pesticides and crops.
* The software shall be able to provide reliable recommendations based on the analysis of the scanned images.

3.3.3. Usability

* The software shall be user-friendly, with a simple and intuitive interface.
* The software shall be easy to install and use, with clear instructions provided for users.

3.3.4. Security

* The software shall have a secure login system, and user data shall be encrypted and stored securely.
* The software shall be protected from cyber-attacks and other security threats.

1. Acceptance Criteria
   1. Crop Scanning

The software shall be able to scan and analyze crop images accurately and provide recommendations for suitable pesticides and crops.

* 1. Identify Growth Rates

The software shall be able to identify crop growth patterns based on weather conditions and provide reliable predictions for expected crop yield.

* 1. Scalability

The software shall be able to handle a large volume of data without any performance issues.

* 1. User-friendly

The software shall be user-friendly, with a simple and intuitive interface.

* 1. Secure

The software shall be secure, with a reliable login system and data encryption.