**Software Requirements Specification (SRS)**

**1. Introduction**

**1.1 Purpose** This SRS defines the functional and non-functional requirements for a machine learning model that aims to classify the health conditions of various crops. The model will leverage data on cultivation practices and historical Sentinel-2 time series data.

**1.2 Scope** The scope of this project includes:

* **Data Acquisition:** Gathering and preprocessing historical Sentinel-2 satellite imagery and relevant cultivation practice data.
* **Feature Engineering:** Extracting meaningful features from the satellite imagery and cultivation data.
* **Model Development:** Training a machine learning model on the processed data to classify crop health conditions.
* **Model Evaluation:** Assessing the performance of the model using appropriate metrics.
* **Deployment:** Deploying the model for real-time or near-real-time crop health monitoring.

**1.3 Definitions, Acronyms, and Abbreviations**

* **Sentinel-2:** A high-resolution multispectral imaging mission of the European Union's Copernicus Programme.
* **ML:** Machine Learning
* **AI:** Artificial Intelligence
* **NDVI:** Normalized Difference Vegetation Index

**2. Overall Description**

**2.1 Product Perspective** The machine learning model will be integrated into a larger agricultural monitoring system. It will provide valuable insights to farmers, agricultural experts, and policymakers to make informed decisions about crop management and resource allocation.

**2.2 Product Functions** The system will perform the following functions:

1. **Data Acquisition:**
   * Acquire Sentinel-2 satellite imagery for the desired geographic region.
   * Collect relevant cultivation practice data (e.g., sowing dates, irrigation schedules, fertilizer application).
2. **Data Preprocessing:**
   * Preprocess satellite imagery to remove noise and atmospheric effects.
   * Align and resample images to a common spatial and temporal reference system.
   * Extract relevant features from the satellite imagery (e.g., NDVI, vegetation indices, spectral bands).
   * Clean and preprocess cultivation practice data.
3. **Model Training:**
   * Train a machine learning model on the preprocessed data to classify crop health conditions (e.g., healthy, stressed, diseased).
   * Experiment with different ML algorithms (e.g., random forest, support vector machine, neural networks) to identify the optimal model.
4. **Model Evaluation:**
   * Evaluate the model's performance using appropriate metrics (e.g., accuracy, precision, recall, F1-score).
5. **Model Deployment:**
   * Deploy the trained model to a cloud-based platform or edge device for real-time or near-real-time monitoring.
6. **User Interface:**
   * Develop a user-friendly interface to visualize crop health maps and provide actionable insights.

**2.3 User Characteristics** The primary users of this system will be:

* **Farmers:** To monitor the health of their crops and make timely interventions.
* **Agricultural Experts:** To analyse large-scale crop health patterns and identify potential risks.
* **Policymakers:** To make informed decisions about agricultural policies and resource allocation.

**3. Specific Requirements**

**3.1 External Interface Requirements**

* **User Interface:** A web-based or mobile application interface to visualize crop health maps and provide insights.
* **Hardware Interface:** Compatible with various hardware platforms (e.g., servers, edge devices) for deployment.
* **Software Interface:** Compatible with standard data formats (e.g., GeoTIFF, CSV) and machine learning frameworks (e.g., TensorFlow, PyTorch, Scikit-learn).

**3.2 Functional Requirements**

* **Data Acquisition:**
  + Automatically download Sentinel-2 imagery based on predefined geographic regions and time intervals.
  + Integrate with data sources to obtain cultivation practice data.
* **Data Preprocessing:**
  + Apply atmospheric correction and radiometric calibration to satellite imagery.
  + Generate vegetation indices and other relevant features.
  + Handle missing data and outliers.
* **Model Training and Evaluation:**
  + Train and evaluate multiple machine learning models to identify the best-performing one.
  + Continuously retrain the model with new data to improve accuracy.
* **Model Deployment:**
  + Deploy the model to a cloud-based platform or edge device for real-time or near-real-time monitoring.
* **User Interface:**
  + Provide interactive maps to visualize crop health conditions.
  + Generate reports and alerts for users.

**3.3 Design Constraints**

* **Computational Resources:** The system should be computationally efficient to handle large datasets and real-time processing.
* **Data Storage:** Sufficient storage capacity is required to store historical and current satellite imagery and cultivation data.
* **Network Connectivity:** Reliable network connectivity is essential for data acquisition, model training, and deployment.

**4. Design Constraints**

* **Performance:** The system should be able to process large datasets and generate accurate predictions in a timely manner.
* **Scalability:** The system should be able to handle increasing data volumes and geographic coverage.
* **Security:** The system should implement appropriate security measures to protect sensitive data.

**5. Additional Specifications**

* **Accuracy:** The model should achieve a high level of accuracy in classifying crop health conditions.
* **Precision:** The model should minimize false positive and false negative predictions.
* **Recall:** The model should identify as many true positive cases as possible.
* **F1-Score:** A balanced metric that considers both precision and recall.

**6. Testing and Verification**

* **Unit Testing:** Test individual components of the system (e.g., data preprocessing, model training, prediction).
* **Integration Testing:** Test the interaction between different components of the system.
* **System Testing:** Test the entire system to ensure it meets functional and non-functional requirements.
* **User Acceptance Testing (UAT):** Involve end-users to validate the system's usability and performance.

**7. Deliverables**

* **Machine Learning Model:** A trained and optimized machine learning model.
* **Data Pipeline:** A robust data pipeline for acquiring, preprocessing, and storing data.
* **User Interface:** A user-friendly interface for visualizing crop health maps and insights.
* **Documentation:** Comprehensive documentation covering system design, implementation, and usage.

By following this SRS, we can develop a robust and effective machine learning model for crop health monitoring, providing valuable insights to stakeholders in the agricultural sector.