Software Requirements Specification
Version 1.0

EcoShield

Theme: Save Planet

Project Name: EcoShield

Category: Transforming Big Data with Data Science

Contents

1.1 Background and Necessity for the Application

Understanding the influence of climatic patterns on the global food chain is crucial for addressing sustainability and food security challenges. The impact of climatic change leads to several significant challenges to food systems globally, affecting both food production and availability. With the advent of technology, there has been a growing interest in leveraging Data Science and Big Data processing to address this challenge. By collecting and analyzing vast amounts of data from various sources such as temperature changes and food security concerns, it is possible to develop accurate results for detecting climatic changes. An application can be developed as a solution to address this.

In the coming decades, the agricultural sector may face many challenges stemming from growing global populations, land degradation, and loss of cropland due to urbanization. In this century climate change is one factor that could affect food production and availability in many parts of the world, particularly those most prone to drought and famine. The use of Big Data processing and Data Science in this application is not just a technological advancement but a necessity for modern agriculture.

1.2 Proposed Solution

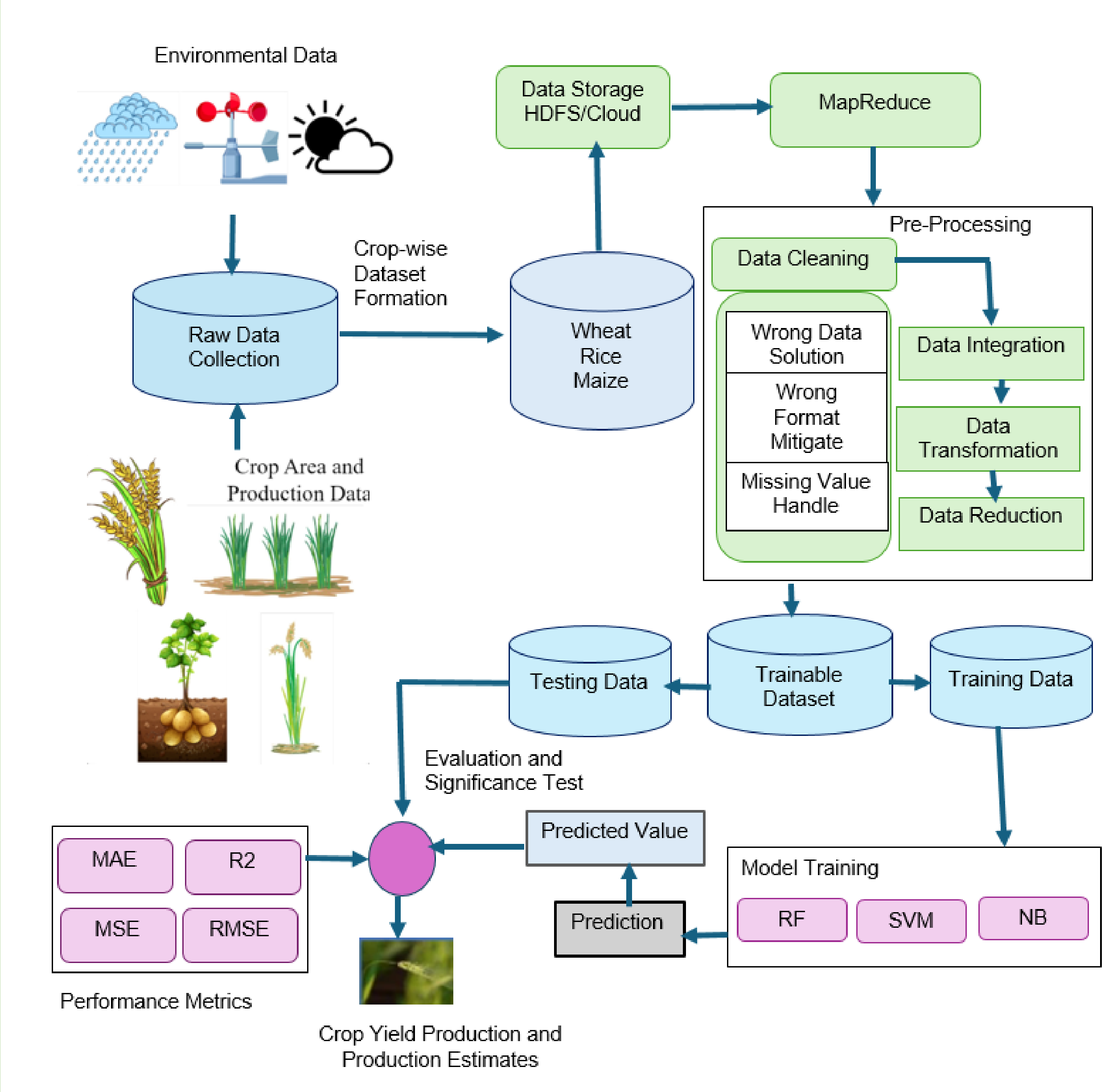
The proposed application, '**EcoShield**', aims to address the impact of climate change on global food chain through a comprehensive strategy that combines scientific research, technological advancements, and community involvement. By utilizing cutting-edge Big Data Processing with Data Science algorithms, it is easy to identify factors to save the planet. Integrating Hadoop Distributed File System (HDFS) allows for efficient, distributed data storage, while MapReduce enables fast and parallel data processing. Together, they enhance the analysis of large datasets for better climate impact monitoring, policy measures, and decision-making.

The first step is to collect a dataset for understanding the impacts of climate change and developing effective solutions. The dataset is provided with you. Alternatively, you can download the dataset from: http://sedac.ciesin.columbia.edu/mva/cropclimate/

Hint: The sample of the dataset for 'EcoShield' for implementation purpose is as follows:

BLS 2 Countries (SRES) ABE	Fips_cod	ISO3v10	WH_2000	RI_2000	MZ_2000	WHA1F2020	RIA1F2020	MZA1F2020	ActChWHA1F2020	ActChRIAIF2020	ActChMZ
Afghanistan	AF	AFG	2750142.86	337285.71	265285.71	-1.22	-3.22	-5.54			
Alaska (same as Canada)	02	USA				6.53	4.53	-0.03			
Albania	AL	ALB	274685.71		212814.29	4.19	2.19	-1.68	11520.95	0.00	
Algeria	AG	DZA	2157083.86	300.00	1249.14	1.35	-0.65	-4.30			
Angola	AO	AGO	4000.00	7791.57	543685.29	1.35	-0.65	-4.30			
Argentina	AR	ARG	14503088.71	914794.86	15968048.57	3.42	1.42	-5.08	496591.53	13027.04	-8
Armenia	AM	ARM	236083.86		12146.57	2.69	0.69	-3.07	6339.97	0.00	
Australia	AS	AUS	20069730.43	891258.57	367102.14	2.67	0.54	-0.33	535510.99	4830.62	
Austria	AU	AUT	1430718.14		1776373.29	4.19	2.19	-1.68	60007.61	0.00	-
Azerbaijan	AJ	AZE	1486676.43	13816.43	134517.86	2.69	0.69	-3.07	39924.29	94.71	
Bangladesh	BG	BGD	1370010.71	38483787.71	203940.71	-7.55	-2.23	-7.75	-103488.04	-857738.18	-
Belarus	BO	BLR	1002379.43		68088.00	-4.72	-6.72	-7.72	-47268.75	0.00	
Belgium	BE	BEL	1675068.29		541648.00	4.19	2.19	-1.68	70256.21	0.00	
Belize	BH	BLZ		12547.86	33505.00	-3.43	-5.43	0.23	0.00	-681.29	
Benin	BN	BEN		62641.43	746581.86	0.14	-1.86	-4.18	0.00	-1164.35	-
Bhutan	BT	BTN	6512.43	51147.29	68433.43	-2.80	-0.70	-1.01	-182.35	-355.49	
Bolivia	BL	BOL	120754.57	346376.14	712158.29	3.42	1.42	-5.08	4134.68	4932.53	-
Bosnia and Herzegovina	BK	BIH	260590.57		811312.57	-4.91	-6.91	-7.91	-12782.40	0.00	-
Botswana	BC	BWA	542.86		8914.71	0.14	-1.86	-4.18	0.77	0.00	
Brazil	BR	BRA	3892588.57	11437436.57	39665359.43	-3.43	-5.43	0.23	-133496.86	-620997.18	9
Brunei	BX	BRN		590.57		-2.80	-0.70	-1.01	0.00	-4.10	
Bulgaria	BU	BGR	3389509.57	20770.14	1346074.00	-4.91	-6.91	-7.91	-166261.07	-1434.21	-1
Burkina Faso	UV	BFA		97038.57	642223.43	0.14	-1.86	-4.18	0.00	-1803.71	-
Burundi	BY	BDI	8089.14	62470.29	123804.71	0.14	-1.86	-4.18	11.43	-1161.17	

The sample architecture for **EcoShield** application can be as follows:



Sample Architecture of the Application

The project **'EcoShield'** involves integrating vast datasets collected from various sources such as agricultural reports to capture a wide range of climatic and agricultural variables. Data science techniques, including Big Data and predictive analytics, are utilized to analyze historical data and identify patterns and trends in crop yields under different climatic conditions.

Initially, raw data is collected from various sources, including environmental data which can be impacted by climate change (temperature, precipitation, humidity, sunlight, CO_2 levels, and extreme weather events). The data is also collected from crop area and production (yield, planting and harvesting dates, soil quality, and farming practices). This data is then organized into a structured, crop-wise dataset and stored in a suitable data storage solution such as Hadoop Distributed File System (HDFS) or cloud storage.

The pre-processing stage involves data cleaning, integration, transformation, and reduction to ensure the dataset is accurate and manageable. MapReduce techniques are applied to process large datasets efficiently. The cleaned and integrated data is then split into training, testing, and trainable datasets.

Multiple Machine Learning models, including Random Forest (RF), Support Vector Machine (SVM), and Naive Bayes (NB), are trained using the training data. These models predict crop yields and production estimates based on the testing data. The performance of the models is evaluated using metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared (R^2).

Significance tests are conducted to validate the predictions. The final output, the predicted crop yield and production estimates, provides valuable insights into how climate change might impact crop production, aiding policymakers, farmers, and researchers in developing mitigation strategies. These insights can be visualized in form of graphs, heatmaps focusing on assessing the models' accuracy, precision, and other metrics ensuring that they meet the desired performance standards.

1.3 Purpose of the Document

The purpose of this document is to present a trained and tested interactive model titled '**EcoShield**'.

This document explains the purpose and features of '**EcoShield**' and the constraints under which it must operate. This document is intended for both stakeholders and developers of the application.

1.4 Scope of Project

The scope of this project encompasses the collection, integration, and analysis of environmental and crop production data to assess the impacts of climate change on global crop yields. Leveraging Big Data and Data Science techniques, the project aims to develop predictive models to estimate future crop production under various climate scenarios.

This includes preprocessing raw data, training machine learning models, and validating predictions with key metrics. The insights gained will inform stakeholders, including policymakers and farmers, enabling them to make data-driven decisions to mitigate climate change effects on food security and optimize agricultural practices.

1.5 Constraints

‘**EcoShield**’ faces several constraints, including data quality and availability, computational resource demands, and model accuracy challenges. Ensuring scalability and interpretability of complex Machine Learning models is critical, as is accounting for temporal dynamics and geospatial variability.

Additionally, the project may require to integrate interdisciplinary expertise and consider policy, economic, ethical, and social factors to provide reliable and actionable insights. Addressing these constraints requires robust data management, advanced analytics, and careful planning to ensure the project's success.

1.6 Functional Requirements

Functional requirements for climatic pattern on food chain system are crucial to ensure its effectiveness in agricultural applications. These requirements define the specific capabilities and behaviors that the application must exhibit to accurately identify and manage the planet’s climatic conditions. Key functional requirements of the application will contain following:

i. **Raw Data Collection** – Collect environmental data (temperature, rainfall, solar radiation), crop area, and production data for various crops (example, wheat, rice, maize) from reliable and up-to-date sources.

ii. **Data Storage and Management** - Store collected data in a distributed storage system (example, HDFS and cloud storage) and use MapReduce for efficient data processing and management. Ensure that the storage system can handle the volume of data and provide quick access for analysis.

iii. **Pre-Processing** – Implement data cleaning procedures to handle missing values, incorrect formats, and erroneous data entries. Integrate data from different sources to create a unified dataset. Perform data transformation to convert data into appropriate formats for analysis. Apply data reduction techniques to remove redundant and irrelevant data, ensuring a more manageable dataset size.

iv. **Model Training** – Use Machine Learning algorithms such as Random Forest (RF), Support Vector Machine (SVM), and Naive Bayes (NB) to train models on the pre-processed data. Split the dataset into training and testing subsets to evaluate model performance. Fine-tune model parameters to improve prediction accuracy and robustness.

v. **Model Test and Evaluation** – Test the trained models on the testing dataset to evaluate their performance. Use metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Mean Squared Error (MSE), and R-squared (R^2) to assess the accuracy and reliability of the models. Conduct significance tests to ensure the statistical validity of the model predictions.

vi. **Prediction and Output Generation** – Use the trained models to make predictions on new data. Generate predicted values for crop yield production and production estimates. Provide the predicted output in a user-friendly format for further analysis and decision-making.

vii. **Result Visualization** - Develop visualizations to represent the predicted results and model performance metrics. Create graphs and charts to help users understand the impact of different variables on crop yield and production.

viii. **System Integration** - Ensure seamless integration of all components, from data pre-processing to model training and prediction. Implement workflows to automate the entire process, ensuring efficient data handling and analysis.

ix. **User Interface** - Provide a user-friendly interface for users to interact with the system, input data, and view results. Ensure the interface is intuitive and provides clear instructions and feedback to the users.

By following these functional requirements, users can focus on the critical aspects of data pre-processing, model training, evaluation, and prediction. This will further lead to a comprehensive understanding of the impact of climate change on crop production using Big Data and Data Science techniques.

1.7 Non-Functional Requirements

There are several non-functional requirements that should be fulfilled by the application. The application should be:

1. **__Usable__**: The application should be designed with a clear and intuitive interface, making it easy for users to understand.
2. **__Scalable__**: The system should be able to handle increasing amounts of data and users without a significant drop in performance. The architecture should support horizontal scaling to accommodate additional storage and computational resources as required.
3. **__Efficient__**: The application should process large datasets efficiently using Big Data technologies to ensure timely data analysis and model training. The response time for data queries and model predictions should be minimized to provide quick insights.
4. **__Available__**: The system should have high availability to ensure that users can access and use it whenever required. Redundancy and failover mechanisms should be implemented to minimize downtime and ensure continuous operation.
5. **__Reliable__**: The application should provide consistent and accurate results even under varying conditions and data loads. Mechanisms should be in place to handle and recover from hardware or software failures to ensure continuous operation.

These are the bare minimum expectations from the project. It is a must to implement the FUNCTIONAL and NON-FUNCTIONAL requirements given in this SRS.

Once they are complete, you can use your own creativity and imagination to add more features if required.

1.8 Interface Requirements

1.8.1 Hardware

Intel Core i5/i7 Processor or higher
8 GB RAM or higher
Color SVGA
500 GB Hard Disk space
Mouse
Keyboard

1.8.2 Software

Technologies to be used:

1. **Data Store**: HDFS, Apache HBase, MongoDB, CSV
2. **Backend**: Apache Spark, Apache Hive
3. **Programming/IDE**: R/Python 3.11.4 or higher, Jupyter Notebook, Anaconda 23.1.0 or higher, Google Colab
4. **Libraries**: OpenCV, TensorFlow, scikit-learn, pandas, numpy, PyTorch, Matplotlib, and Seaborn
5. **Visualization**: Tableau Desktop

1.9 Project Deliverables

You will design and build the project and submit it along with a complete project report that includes:

- Problem Definition
- Design Specifications
- Diagrams such as User Flow Diagram/User Journey Map
- Test Data Used in the Project
- Project Installation Instructions
- Link of GitHub for accessing the uploaded project code (Link should have public access)
- Detailed Steps to execute the project
- Link of Published Blog

The consolidated project must be submitted on GitHub with a ReadMe.doc file listing assumptions (if any) made at your end. Provide the GitHub URL where the project has been uploaded for sharing. The repository on GitHub should have public access. Documentation is very important part of the project hence, all crucial aspects of the project must be documented properly. Ensure that documentation is complete and comprehensive.

The source code, including .ipynb files for Jupyter Notebook and Google Colab, should be shared via GitHub. Appropriate access permissions should be granted to users to allow testing for Jupyter Notebook and Google Colab. The consolidated project must be submitted on GitHub with a ReadMe.doc file listing assumptions (if any) made at your end.

You should publish a blog of minimum 2000 words on any free blogging Website such as Blogger, Tumblr, Ghost or any other blogging Website. The link of the published blog should be submitted along with the project documentation.

Submit a video (.mp4 file) demonstrating the working of the application, including all the functionalities of the project. This is MANDATORY.

~~~ End of Document ~~~