EXPLORATORY ANALYSIS OF RAINFALL DATA IN INDIA FOR AGRICULTURE

NALAIYA THIRAN PROJECT BASED LEARNING
On
PROFESSIONAL READINESS FOR INNOVATION, EMPLOYABILITY AND ENTREPRENEURSHIP

A PROJECT REPORT

SRIRAJA B	19104157
SURESH RAJAN M	19104160
THAVAPRAKASH S	19104167
SOLASA NAGA SAI SURENDRA GUPTA	19104153

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

HINDUSTHAN COLLEGE OF ENGINEERING AND TECHOLOGY

Approved by AICTE, New Delhi, Accredited with ‘A’ Grade by NAAC

(An Autonomous Institution, Affiliated to Anna University, Chennai)

COIMBATORE – 641 032

November 2022

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

1.1.1 INTRODUCTION ON MACHINE LEARNING

Machine learning (ML) is the study of computer algorithms that can improve automatically through experience and by the use of data. It is seen as a part of artificial intelligence. Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so. Machine learning algorithms are used in a wide variety of applications, such as in medicine, email filtering, speech recognition, and computer vision, where it is difficult or unfeasible to develop conventional algorithms to perform the needed tasks.

A subset of machine learning is closely related to computational statistics, which focuses on making predictions using computers; but not all machine learning is statistical learning. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning. Data mining is a related field of study, focusing on exploratory data analysis through unsupervised learning. Some implementations of machine learning use data and neural networks in a way that mimics the working of a biological brain.

Machine learning approaches are traditionally divided into three broad categories, depending on the nature of the "signal" or "feedback" available to the learning system:

- Supervised learning: The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.

- Unsupervised learning: No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning).

- Reinforcement learning: A computer program interacts with a dynamic environment in which it must perform a certain goal (such as driving a vehicle or playing a game against an opponent). As it navigates its problem space, the program is provided feedback that's analogous to rewards, which it tries to maximize.

Supervised learning

Supervised learning algorithms build a mathematical model of a set of data that contains both the inputs and the desired outputs. The data is known as training data, and consists of a set of training examples. Each training example has one or more inputs and the desired output, also known as a supervisory signal. In the mathematical model, each training example is represented by an array or vector, sometimes called a feature vector, and the training data is represented by a matrix. Through iterative optimization of an objective function, supervised learning algorithms learn a function that can be used to predict the output associated with new inputs. An optimal function will allow the algorithm to correctly determine the output for inputs that were not a part of the training data. An algorithm that improves the accuracy of its outputs or predictions over time is said to have learned to perform that task.

Types of supervised machine learning algorithms which will the include active learning , classification and regression. Classification algorithms are used when the outputs are restricted to a limited set of values, and regression algorithms are used when the outputs may have any numerical value within a range. As an example, for a classification algorithm that filters emails, the input would be an incoming email, and the output would be the name of the folder in which to file the email.

Unsupervisied Learning

Unsupervised learning algorithms take a set of data that contains only inputs, and find structure in the data, like grouping or clustering of data points. The algorithms, therefore, learn from test data that has not been labelled, classified or categorized. Instead of responding to feedback, unsupervised learning algorithms identify commonalities in the data and react based on the presence or absence of such commonalities in each new piece of data. A central application of unsupervised learning is in the field of density estimation in statistics, such as finding the probability density function.

Cluster analysis is the assignment of a set of observations into subsets (called clusters) so that observations within the same cluster are similar according to one or more predesignated criteria, while observations drawn from different clusters are dissimilar. Different clustering techniques make different assumptions on the structure of the data, often defined by some similarity metric and evaluated, for example, by internal compactness, or the similarity between members of the same cluster, and separation, the difference between clusters. Other methods are based on estimated density and graph connectivity.

Semi-Supervisied Learning

Semi-supervised learning falls between unsupervised learning (without any labelled training data) and supervised learning (with completely labelled training data). Some of the training examples are missing training labels, yet many machine-learning researchers have found that unlabelled data, when used in conjunction with a small amount of labelled data, can produce a considerable improvement in learning accuracy.

In weakly supervised learning, the training labels are noisy, limited, or imprecise; however, these labels are often cheaper to obtain, resulting in larger effective training sets.

Reinforcement Learning

Reinforcement learning is an area of machine learning concerned with how software agents ought to take actions in an environment so as to maximize some notion of cumulative reward. Due to its generality, the field is studied in many other disciplines, such as game theory, control theory, operations research, information theory, simulation-based optimization, multi-agent systems, swarm intelligence, statistics and genetic algorithms. In machine learning, the environment is typically represented as a Markov decision process (MDP). Many reinforcement learning algorithms use dynamic programming techniques. Reinforcement learning algorithms do not assume knowledge of an exact mathematical model of the MDP, and are used when exact models are infeasible. Reinforcement learning algorithms are used in autonomous vehicles or in learning to play a game against a human opponent.

1.1.2 FLASK WEB FRAMEWORK

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools. Applications that use the Flask framework include Pinterest and LinkedIn.

Flask was created by Armin Ronacher of Pocoo, an international group of Python enthusiasts formed in 2004. According to Ronacher, the idea was originally an April Fool's joke that was popular enough to make into a serious application. The name is a play on the earlier Bottle framework. When Ronacher and Georg Brandl created a bulletin board system written in Python, the Pocoo projects Werkzeug and Jinja were developed. In April 2016, the Pocoo team was disbanded and development of Flask and related libraries passed to the newly formed Pallets project. Flask has become popular among Python enthusiasts. As of October 2020, it has second most stars on GitHub among Python web-development frameworks, only slightly behind Django, and was voted the most popular web framework in the Python Developers Survey 2018.

FEATURES

- Development server and debugger
- Integrated support for unit testing
- RESTful request dispatching
- Uses Jinja templating
- Support for secure cookies (client-side sessions)
- 100% WSGI 1.0 compliant
- Unicode-based
- Extensive documentation
- Google App Engine compatibility
- Extensions available to enhance features desired

1.1.3 MACHINE LEARNING ALGORITHM

Random Forest

A random forest is the machine learning technique that is used to solve regression and classification problem. Technique that combines many classifiers to provide solutions to complex problem. A random forest algorithm has consisted of many decision trees. Ensemble meta-algorithm which improves the accuracy of the algorithms. This algorithm outcome predictions of the decision trees. It predicts by taking the average or mean of the output from various trees. Increasing the number of trees increases the accuracy of the outcome. A random forest limits the decision tree algorithm. It generates predictions without requiring many configurations.

Decision Tree

Decision Tree is the Supervised machine learning technique which can be used for both classification and Regression problems, where it is preferred for solving Classification problems only. It is a tree-structured like classifier, where branches represent the decision rules and internal nodes are represent the features of a dataset and each leaf node represents the outcome. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches. In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node. The decisions or the test are performed on the basis of features of the given dataset. It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.

KNN

The algorithm calculates the distance of a new data point to all other training data points. The distance can be of any type, e.g., Euclidean, Manhattan, etc. The algorithm then sorts the calculated Euclidean distances in ascending order. Next, the algorithm selects the k-nearest data points, where k can be any integer. The selection based on the proximity to other data points regardless of what feature the numerical values represent. Finally, the algorithm assigns the data point to the class where similar data points lie.

Linear Regression Algorithm

Linear regression is the machine learning algorithm which is used to predict the relationship between two variables or factors. One variable is independent variable is used to predict the value for another variable called dependent variable. Linear Regression is a supervised machine learning algorithm. It predicts the value between a continuous range of numbers.

Logistic Regression

Logistic regression is a supervised machine learning classification algorithm used to predict the probability of the target variables. Where the nature of dependent variable is dichotomous, which means it would be only two possible classes. In simple. The dependent variable is binary in nature having data coded as either 1 or 0 . Mathematically, a logistic regression model predicts P(Y=1) as a function of X. It is one of the simplest Machine learning algorithms that can be used for various types of classification problems like as spam detection, Diabetes prediction, cancer detection etc.

Support Vector Machines (SVM)

Support vector machines are powerful and flexible supervised machine learning algorithms which are used both for classification and regression. They are used in classification problems. In 1960, SVM were first introduced but later they got refined in 1990. SVMs have the unique way of implementation to other machine learning algorithms. They are extremely popular because of their ability to handle multiple continuous and categorical variables.

1.1.4 INTRODUCTION ON AGRICULTURE AND RAINFALL

Agriculture is the practice of cultivating plants and livestock. Agriculture was the key development in the rise of sedentary human civilization, whereby farming of domesticated species created food surpluses that enabled people to live in cities. The history of agriculture began thousands of years ago. After gathering wild grains beginning at least 105,000 years ago, nascent farmers began to plant them around 11,500 years ago. Pigs, sheep, and cattle were domesticated over 10,000 years ago. Plants were independently cultivated in at least 11 regions of the world. Industrial agriculture based on large-scale monoculture in the twentieth century came to dominate agricultural output, though about 2 billion people still depended on subsistence agriculture. The major agricultural products can be broadly grouped into foods, fibres, fuels and raw materials (such as rubber). Food classes include cereals (grains), vegetables, fruits, oils, meat, milk, eggs and fungi. Over one-third of the world's workers are employed in agriculture, second only to the service sector, although in recent decades, the global trend of a decreasing number of agricultural workers continues, especially in developing countries where smallholding is being overtaken by industrial agriculture and mechanization that brings an enormous crop yield increase.

Modern agronomy, plant breeding, agrochemicals such as pesticides and fertilizers, and technological developments have sharply increased crop yields, but causing ecological and environmental damage. Selective breeding and modern practices in animal husbandry have similarly increased the output of meat, but have raised concerns about animal welfare and environmental damage. Environmental issues include contributions to global warming, depletion of aquifers, deforestation, antibiotic resistance, and growth hormones in industrial meat production. Agriculture is both a cause of and sensitive to environmental degradation, such as biodiversity loss, desertification, soil degradation and global warming, all of which can cause decreases in crop yield. Genetically modified organisms are widely used, although some are banned in certain countries.

Agricultural science is a broad multidisciplinary field of biology that encompasses the parts of exact, natural, economic and social sciences used in the practice and understanding of agriculture. It covers topics such as agronomy, plant breeding and genetics, plant pathology, crop modelling, soil science, entomology, production techniques and improvement, study of pests and their management, and study of adverse environmental effects such as soil degradation, waste management, and bioremediation.

The scientific study of agriculture began in the 18th century, when Johann Friedrich Mayer conducted experiments on the use of gypsum (hydrated calcium sulphate) as a fertilizer. Research became more systematic when in 1843, John Lawes and Henry Gilbert began a set of long-term agronomy field experiments at Rothamsted Research Station in England; some of them, such as the Park Grass Experiment, are still running. In America, the Hatch Act of 1887 provided funding for what it was the first to call "agricultural science", driven by farmers' interest in fertilizers.[216] In agricultural entomology, the USDA began to research biological control in 1881; it instituted its first large program in 1905, searching Europe and Japan for natural enemies of the gypsy moth and brown-tail moth, establishing parasitoids (such as solitary wasps) and predators of both pests in the USA.

1.1.5 MACHINE LEARNING IN AGRICULTURE

Machine learning is evolving along with big data technologies and other fast computing devices. They are growing to create new opportunities to understand the various data processes related to the environmental functions of agriculture. Machine learning can be defined as the scientific method that will allow machines the ability to learn without programming the devices. Machine learning is used in various scientific areas such as Bioinformatics, Biochemistry, Medicines, Meteorology, Economic Sciences, Robotics, Food Security and Climatology.

Artificial Intelligence is being used in various sectors from home to office and now in the agriculture sectors. Machine learning in agriculture used to improve the productivity and quality of the crops in the agriculture sector.

Retailers: The seed retailers use this agriculture technology to churn the data to create better crops. While the pest control companies are using them to identify the various bacteria's, bugs and vermin.

AI is used to boost the yield of crops: The AI technologies are used to determine which corn and which conditions will produce the best yield. It will also determine which weather condition will give the highest return.

AI helps to identify bug hunters: One of the companies named Rentokil is using AI to kill all the bugs and vermin. Other companies are making use of Android app which is developed by Accenture to find bugs. The app takes the pictures of the bug and runs the app called as Peptide. When a bug is found app will provide an immediate solution which helps the technician to take further actions. It will also recommend the chemical to be used to kill the bugs.

Agriculture Robot: Most of the companies are now programming and designing robots to handle the essential task related to agriculture. This includes harvesting crops and works faster than then human laborers. This is the best example of machine learning in agriculture.

Companies are now making use of technologies and deep learning algorithms. The data are then collected using the drones and other software to monitor the crops and also the soil. They also use the software to control the fertility of the soil.

- The agricultural farmers are now taking advantage of the machine learning models and their innovations. Using AI and machine learning is good for the food tech segments.
- The Farmers Business Network that is being created for the farmers a social network will make use of the ML and the analytic tools to drive the results of data on pricing.
- Robots are now managing the crops and also monitoring them.
- Sensors are helping to collect the data related to crops.
- According to research if AI and ML are being used in agriculture, then the agriculture sector will grow in the coming years.

1.2 PURPOSE

In India, Agriculture plays a major role in Indian economy. It almost 18% of total GDP. Agriculture played a major role in every human life. From the ancient times, agriculture is considered to be the main practices in India. In ancient times, people used to cultivate their own crops in order to meet their requirement. Most of the crop are mainly dependent on soil and climate factors. Due to change in climate, there is lot of crops are destroying. On other hand, the available land of agriculture is decreasing continuously. Moreover, the food is wasted due to various factors like climate change, More Rainfall, pest, selection of unsuitable crop.

In recent years, various innovations are used in agriculture field for prediction and decision based on weather and soil factor. More methods can be categories into statistics model like multiple linear regression method and machine learning algorithms that input data to output and learns from data itself. Machine Learning is used for solving various issues such as weather prediction, crop disease prediction, selection of suitable crops, crop yields prediction, developing automated irrigation system, etc.

This comparative study is conducted concentrating on the following aspects: modeling inputs, Visualizing the data, modeling methods, and pre-processing techniques. The results provide a comparison of various evaluation metrics of these machine learning techniques and their reliability to predict rainfall by analyzing the weather data.

We will be using classification algorithms such as Decision tree, Random forest, KNN, and xgboost. We will train and test the data with these algorithms. From this best model is selected and saved in pkl format. Once the model is saved, we integrate it with flask application and also deploy the model in IBM.

CHAPTER 2

LITERATURE SURVEY

2.1 EXISTING PROBLEM

Landslides are considered to be calamitous natural hazards commonly recurring in the Indian Himalayas. Majority of landslides are induced by prolonged or heavy rainfall. Rainfall forecasting helps in identifying the precipitation conditions responsible for landslide occurrence. The proposed research work provides the performance comparison of various machine learning algorithms such as linear regression, back propagation neural network (BPNN), support vector regression (SVR) and long short term memory network (LSTM) used to forecast rainfall which can be compared with the rainfall thresholds to predict landslide occurrence. The analysis is performed using antecedent rainfall data obtained from Narendra Nagar, a small town in the Tehri Garhwal district of Uttarakhand for the period of 1901-2015. Owing to the limited predictability of instantaneous state of the weather, daily rainfall observations are aggregated into monthly indexes. The proposed algorithms use preprocessing techniques followed by data normalization to increase the accuracy of forecasting models. The developed models have the ability to predict rainfall intensity one month in advance or for a specific month of the upcoming year depending upon the dataset used. The study concludes that the BPNNs are able to outperform and provide optimal inferences stating the aptness of artificial neural networks (ANNs) in estimating rainfall and hence predicting the possibility of landslide occurrence well in advance. The study is conducted explicitly for regions highly vulnerable to landslides near Narendra Nagar but may be implemented to any landslide prone area.

Rainfall forecasting is a technologically and scientifically a challenging task around the world. Rainfall is one of the most important weather conditions in a given area. Forecasting possible rainfall can help to solve several problems related to the tourism industry, natural disaster management, agricultural industry etc. As the Sri Lankan rural economy is mostly based on agriculture, it is important to forecast rainfall as well as other weather conditions accurately. The weather patterns are localized and hence, generalization of weather prediction models is very difficult. Therefore, this project proposes three data mining models to forecast rainfall, and compares the prediction performances of those models. To that end the data mining models linear regression, SMO regression, and M5P model were trained from rainfall data collected from the Badulla district, Sri Lanka, during the period 2002 to 2017, to forecast weekly rainfall for the following five months leadtime. Each model was evaluated using Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Root

Relative Squared Error (RRSE), Root Absolute Error (RAE), Direction Accuracy (DA) and residual analysis. According to the findings, the M5P model tree provided the lowest error value, highest direction accuracy, highest correlation between actual and predicted rainfall values, and better randomness of the error values compared to the linear and SMO regression models.

Amount of Rainfall prediction is a major issue for the weather department as it is associated with the human's life and the economy. Excess rainfall is the major cause of natural disasters such as drought and flood which are encountered by the people every year across the world. The time series machine learning model is used for forecasting rainfall at Tamilnadu. Forecasting data required for the analysis is available in the Indian meteorological department. To model the monthly rainfall in Tamilnadu for the period from January 1990 to December 2017, the seasonal ARIMA (Auto Regressive Integrated Moving Average) technique is applied. Using the SARIMA (Seasonal Auto Regressive Integrated Moving Average), the stationarity of the time series flow was demonstrated by the rainfall prediction model and then seasonal correlogram assessed. In relation to the Mean Squared Error (MSE) and Root Mean Squared Error, the output of this model is assessed (RMSE). Therefore, it reveals that the ARIMA model accurately forecasts the Rainfall with less error and the derived model could be used to forecast Monsoon rainfall for the upcoming years

2.2 REFERENCES

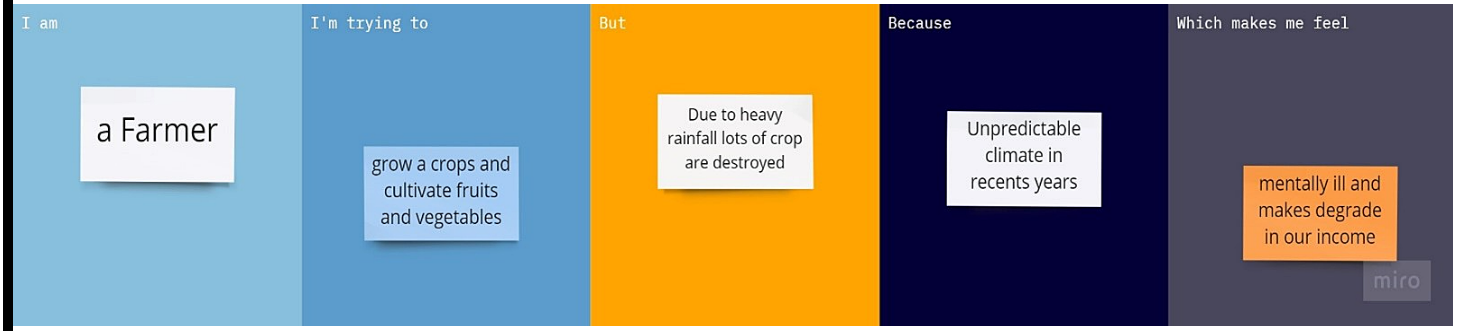
[1] Shikha Srivastava , Nishchay Anand , Sumit Sharma , Sunil Dhar , Lokesh Kumar Sinha, "**Monthly Rainfall Prediction Using Various Machine Learning Algorithms for Early Warning of Landslide Occurrence**" in 2020 International Conference for Emerging Technology (INCET) Belgaum, India

[2] T. Dananjali , S. Wijesinghe, J. Ekanayake, "**Forecasting Weekly Rainfall Using Data Mining Technologies**" in 2020 International Conference for Emerging Technology (INCET) Belgaum, India

[3] Ashwini, U., Kalaivani, K., Ulagapriya, K., & Saritha, A. (2021),"**Time Series Analysis based Tamilnadu Monsoon Rainfall Prediction using Seasonal ARIMA** on 2021 6th International Conference on Inventive Computation Technologies (ICICT).

2.3 REFERENCES PROBLEM STATEMENT DEFINITION

Rainfall has been a major concern these days. Weather conditions have been changing for time being. Rainfall forecasting is important otherwise, it may lead to many disasters. Irregular heavy rainfall may lead to the destruction of crops, heavy floods that can cause harm to human life. It is important to exactly determine the rainfall for effective use of water resources, crop productivity, and pre-planning of water structures.This comparative study is conducted concentrating on the following aspects: modeling inputs, Visualizing the data, modeling methods, and pre-processing techniques. The results provide a comparison of various evaluation metrics of these machine learning techniques and their reliability to predict rainfall by analyzing the weather data.We will be using classification algorithms such as Decision tree, Random forest, KNN, and xgboost. We will train and test the data with these algorithms. From this best model is selected and saved in pkl format. Once the model is saved, we integrate it with flask application and also deploy the model in IBM.



I am	I'm trying to	But	Because	Which makes me feel
a Farmer	grow a crops and cultivate fruits and vegetables	Due to heavy rainfall lots of crop are destroyed	Unpredictable climate in recents years	mentally ill and makes degrade in our income

Problem Statement	PS1
I am	Farmer
I’m trying to	Grow a crop, cultivate fruit and vegetables and supply food to peoples
But	Due to heavy rainfall in recent days cause lots of crops are destroyed
Because	Climate is unpredictable and there is heavy rainfall in recent days due to climate changes
Which makes me feel	Mentally ill and affect my income and food supply market

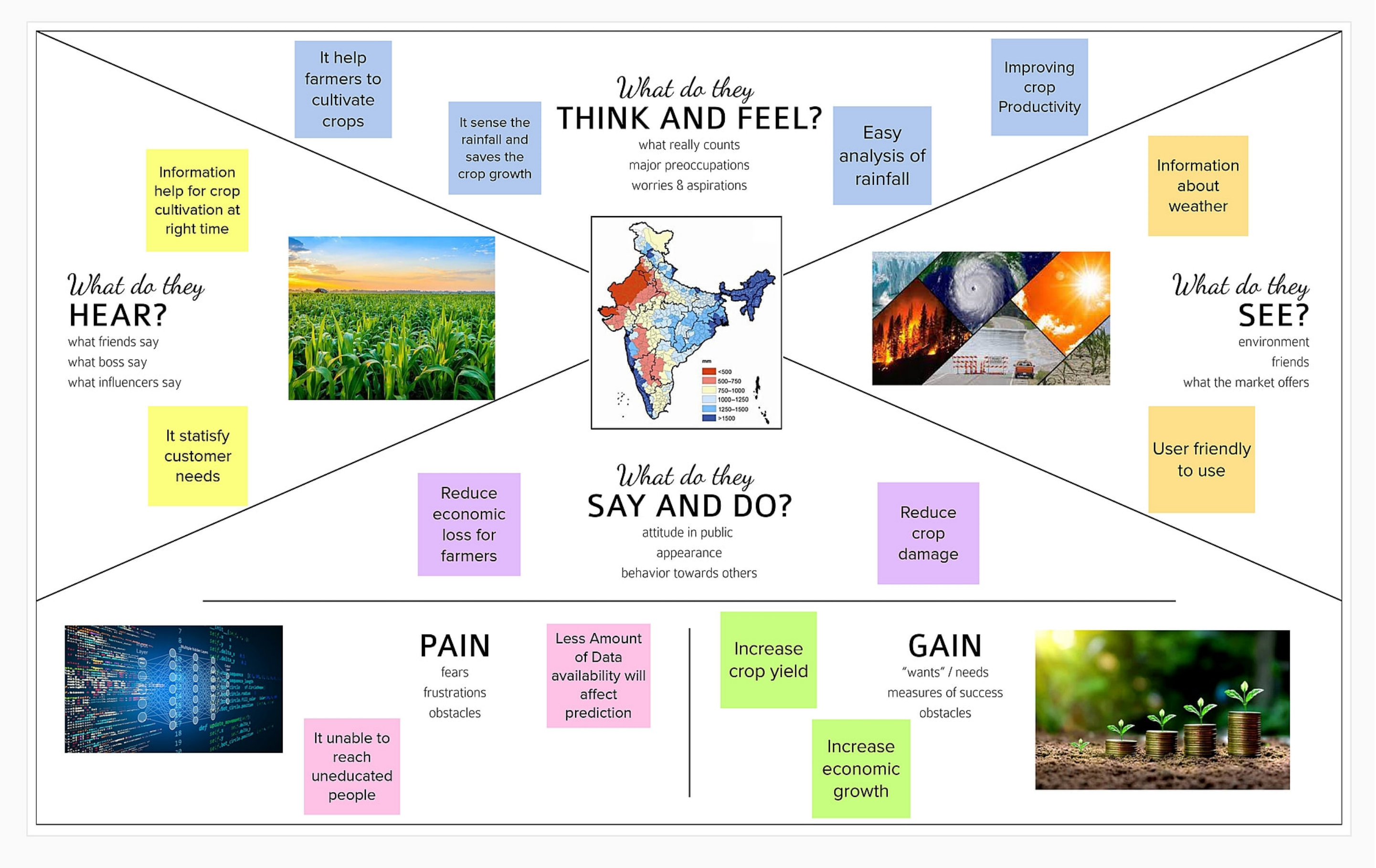
CHAPTER 3

IDEATION AND PROPOSED SOLUTION

3.1 EMPATHY MAP

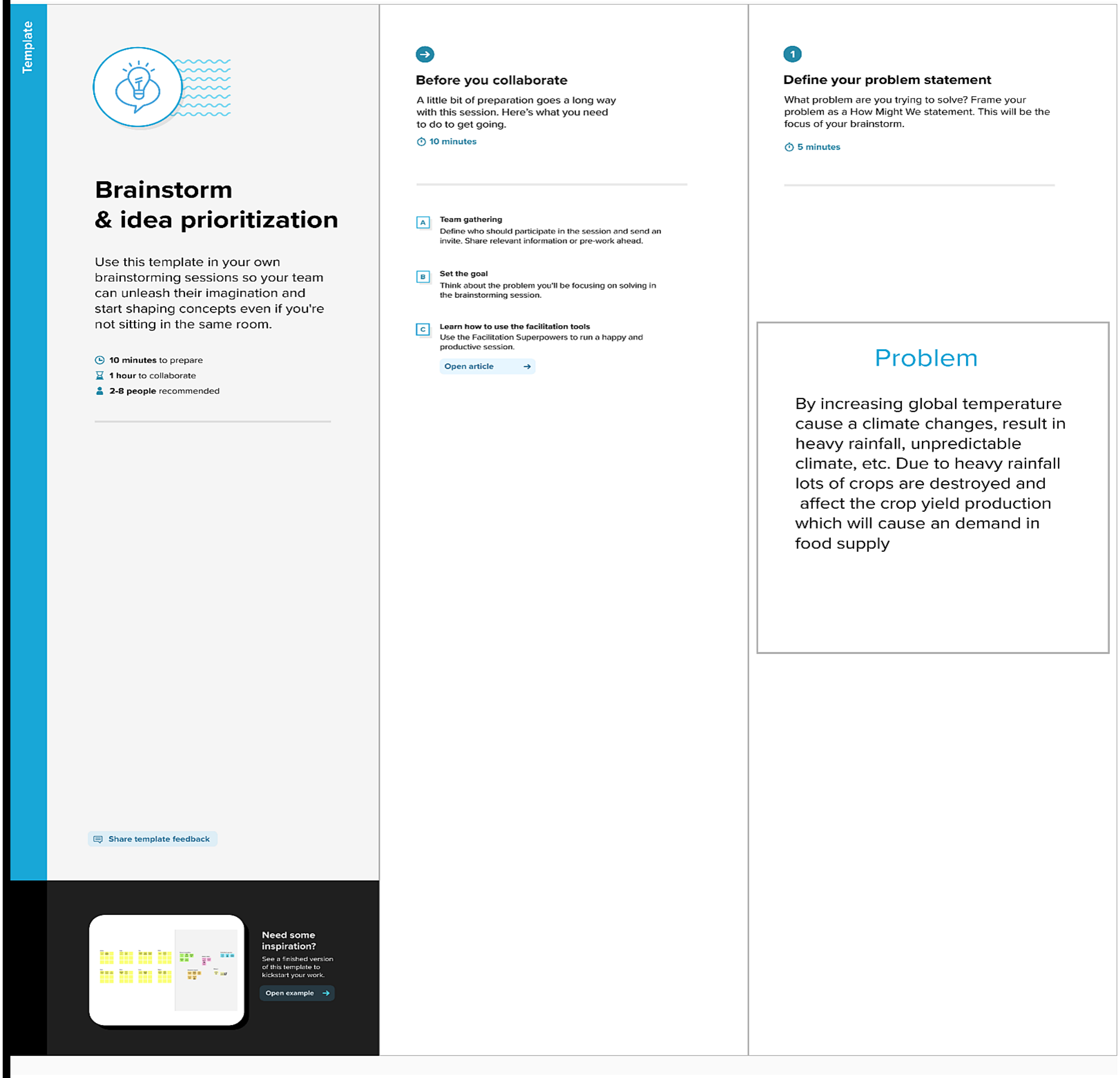
An empathy map is a simple, easy-to-digest visual that captures knowledge about a user’s behaviours and attitudes. It is a useful tool to helps teams better understand their users.

Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user’s perspective along with his or her goals and challenges.



3.2 IDEATION & BRAINSTORMING

In this activity you are expected to list the ideas (at least 4 per each team member) by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.



Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

- **10 minutes** to prepare
- **1 hour** to collaborate
- **2-8 people** recommended

Need some inspiration?

See a finished version of this template to kickstart your work.

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A **Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B **Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.

C **Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

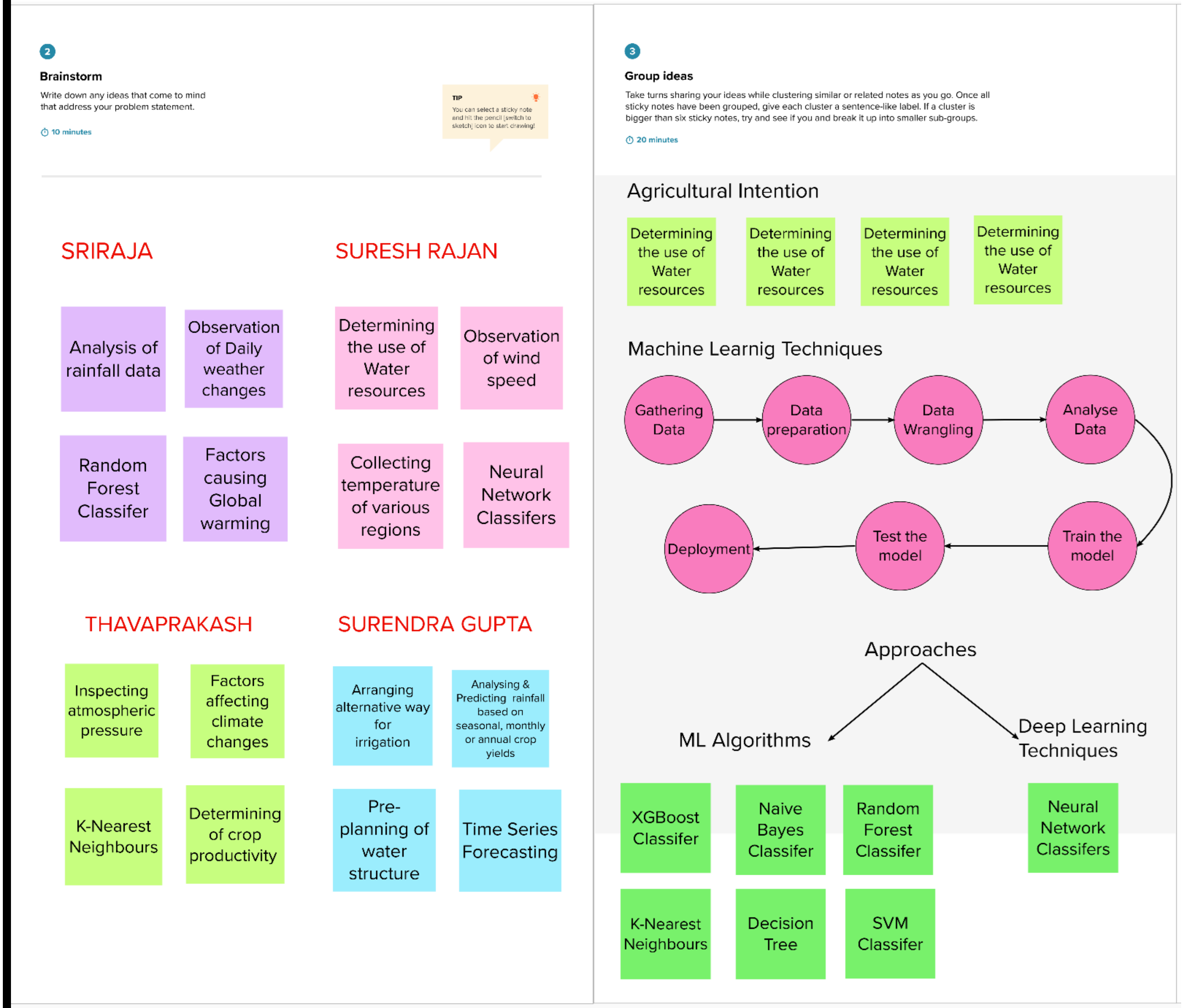
1 Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

Problem

By increasing global temperature cause a climate changes, result in heavy rainfall, unpredictable climate, etc. Due to heavy rainfall lots of crops are destroyed and affect the crop yield production which will cause an demand in food supply



2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

SRIRAJA

- Analysis of rainfall data
- Observation of Daily weather changes
- Random Forest Classifer
- Factors causing Global warming

SURESH RAJAN

- Determining the use of Water resources
- Observation of wind speed
- Collecting temperature of various regions
- Neural Network Classifers

THAVAPRAKASH

- Inspecting atmospheric pressure
- Factors affecting climate changes
- K-Nearest Neighbours
- Determining of crop productivity

SURENDRA GUPTA

- Arranging alternative way for irrigation
- Analysing & Predicting rainfall based on seasonal, monthly or annual crop yields
- Pre-planning of water structure
- Time Series Forecasting

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

Agricultural Intention

- Determining the use of Water resources
- Determining the use of Water resources
- Determining the use of Water resources
- Determining the use of Water resources

Machine Learnig Techniques

Gathering Data → Data preparation → Data Wrangling → Analyse Data → Train the model → Test the model → Deployment

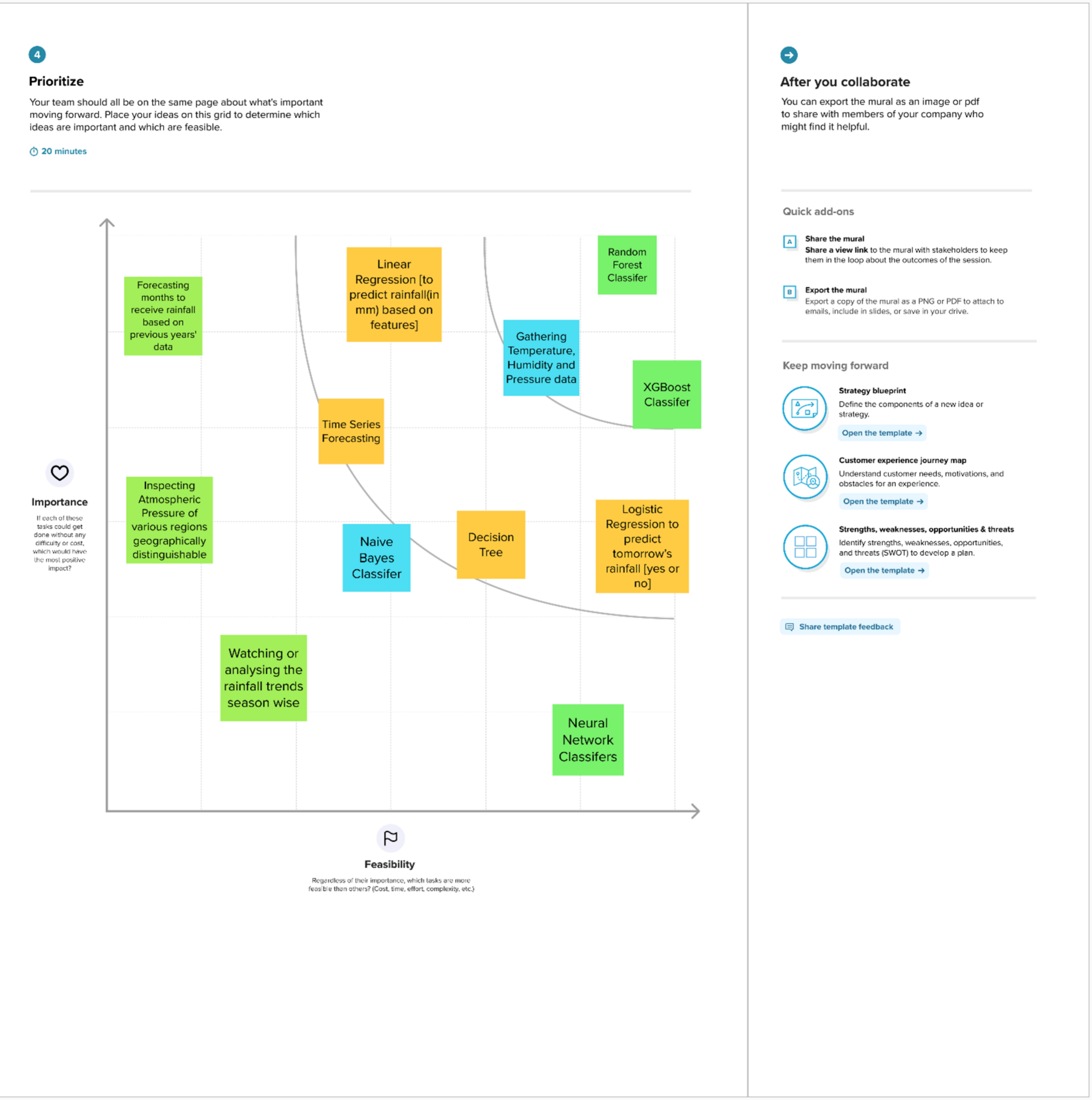
Approaches

ML Algorithms

- XGBoost Classifer
- Naive Bayes Classifer
- Random Forest Classifer
- K-Nearest Neighbours
- Decision Tree
- SVM Classifer

Deep Learning Techniques

- Neural Network Classifers



4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

Forecasting months to receive rainfall based on previous years' data
Linear Regression [to predict rainfall(in mm) based on features]
Random Forest Classifer
Gathering Temperature, Humidity and Pressure data
XGBoost Classifer
Time Series Forecasting
Inspecting Atmospheric Pressure of various regions geographically distinguishable
Naive Bayes Classifer
Decision Tree
Logistic Regression to predict tomorrow's rainfall [yes or no]
Watching or analysing the rainfall trends season wise
Neural Network Classifers
Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?
Feasibility
Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

After you collaborate

You can export the mural as an image or pdf to share with members of your company who might find it helpful.

Quick add-ons

A **Share the mural**
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.

B **Export the mural**
Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save in your drive.

Keep moving forward

Strategy blueprint
Define the components of a new idea or strategy.
Open the template →

Customer experience journey map
Understand customer needs, motivations, and obstacles for an experience.
Open the template →

Strengths, weaknesses, opportunities & threats
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.
Open the template →

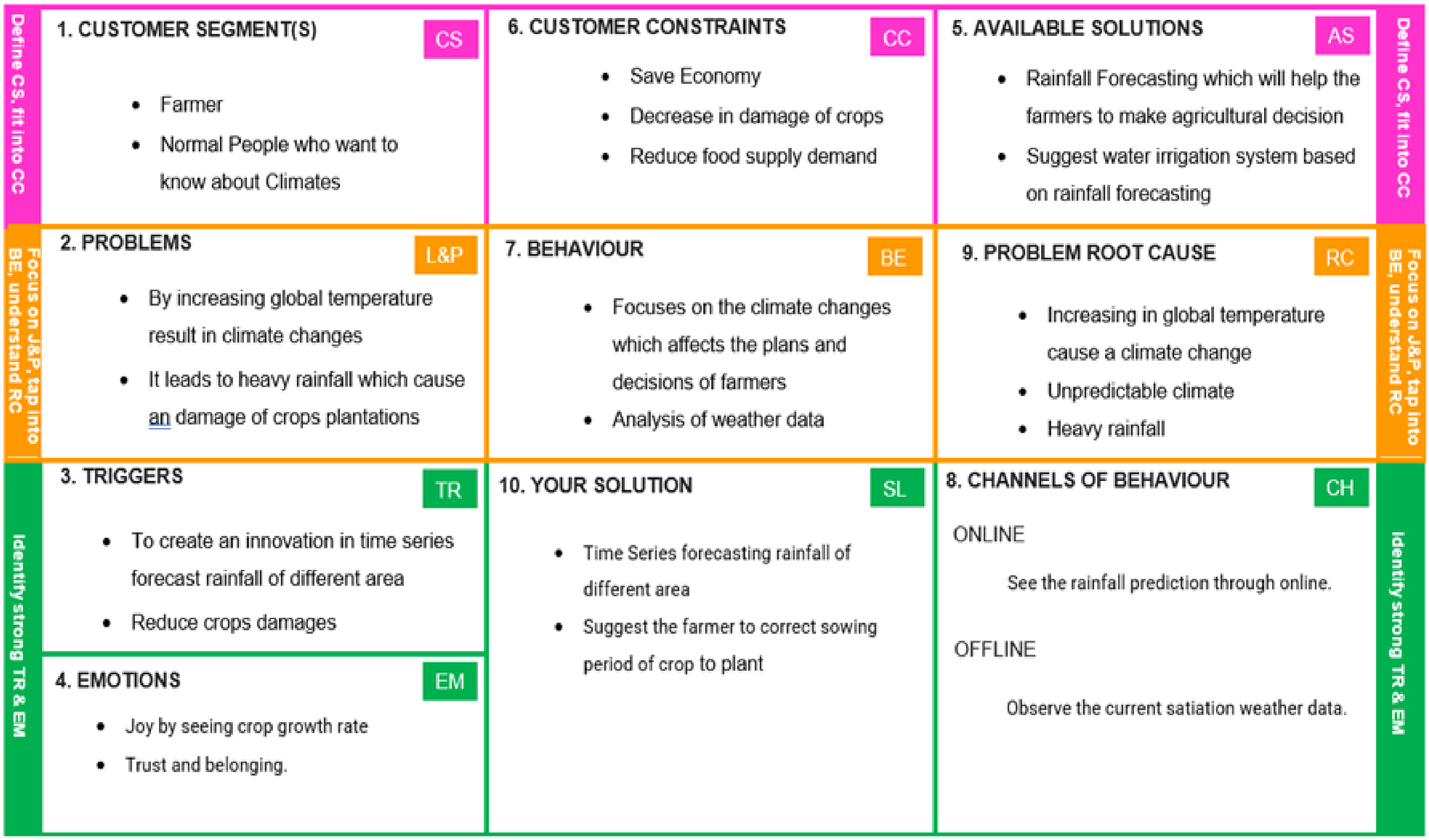
Share template feedback

3.3 PROPOSED SOLUTION

S.No.	**Parameter**	**Description**
1.	Problem Statement (Problem to be solved)	By increasing global temperature cause a climate changes result in heavy rainfall, unpredictable weathers, etc. Due to heavy rainfall lots of crops are destroyed and affect the crop yield production which will cause a demand in food supply.
2.	Idea / Solution description	1. Analysing rainfall using previous years rainfall data 2. Forecast the time series rainfall
3.	Novelty / Uniqueness	1. Forecasting time series rainfall 2. Warn the farmers about the climate change and rainfall level
4.	Social Impact / Customer Satisfaction	1. Agriculture plays major role in Indian GDP almost 18% 2. It will raise our Indian GDP 3. It will help the farmer to identify the correct period of time to sowing a crop and cultivating a crop 4. It will increase the food production
5.	Business Model (Revenue Model)	5. It will raise our India's GDP and will increase the crop production 6. It will reduce the food supply demand

6.	Scalability of the Solution	This will help the farmers in scheduling their agricultural process and increase crop yield production rate.

3.4 PROBLEM SOLUTION FIT



Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

- Farmer
- Normal People who want to know about Climates

6. CUSTOMER CONSTRAINTS CC

- Save Economy
- Decrease in damage of crops
- Reduce food supply demand

5. AVAILABLE SOLUTIONS AS

- Rainfall Forecasting which will help the farmers to make agricultural decision
- Suggest water irrigation system based on rainfall forecasting

Define CS, fit into CC

Focus on J&P, tap into BE, understand RC

2. PROBLEMS L&P

- By increasing global temperature result in climate changes
- It leads to heavy rainfall which cause an damage of crops plantations

7. BEHAVIOUR BE

- Focuses on the climate changes which affects the plans and decisions of farmers
- Analysis of weather data

9. PROBLEM ROOT CAUSE RC

- Increasing in global temperature cause a climate change
- Unpredictable climate
- Heavy rainfall

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS TR

- To create an innovation in time series forecast rainfall of different area
- Reduce crops damages

4. EMOTIONS EM

- Joy by seeing crop growth rate
- Trust and belonging.

10. YOUR SOLUTION SL

- Time Series forecasting rainfall of different area
- Suggest the farmer to correct sowing period of crop to plant

8. CHANNELS OF BEHAVIOUR CH

ONLINE

See the rainfall prediction through online.

OFFLINE

Observe the current satiation weather data.

Identify strong TR & EM

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

Following are the functional requirements of the proposed solution.

FR No.	**Functional Requirement (Epic)**	**Sub Requirement (Story / Sub-Task)**
FR-1	User Registration	Registration through Registration Form or Google account
FR-2	User Confirmation	Confirmation via Email and OTP
FR-3	User Login	Using the registered email id and password as login credentials
FR-4	Dashboard	Display current rainfall forecasting
FR-5	Rainfall Prediction	Train and predict the rainfall
FR-6	Crop recommendation	Train and recommend the sowing crop based on soil and weather parameter

4.2 NON-FUNCTIONAL REQUIREMENT

Following are the non-functional requirements of the proposed solution.

FR No.	**Non-Functional Requirement**	**Description**
NFR-1	**Usability**	The Farmer and other people can easily use the application and it is user friendly to prior knowledge is required for using it
NFR-2	**Security**	Providing secure system requirements and then determine authenticity, originality and security
NFR-3	**Reliability**	The system will provide the prediction without any errors and failures for a specific time
NFR-4	**Performance**	Predictions are as same as the true values, so the performance is higher

NFR-5	**Availability**	Available to different group of farmers for 24/7
NFR-6	**Scalability**	The Application should be in the way of adding new functionalities or modules without affecting the existing functionalities

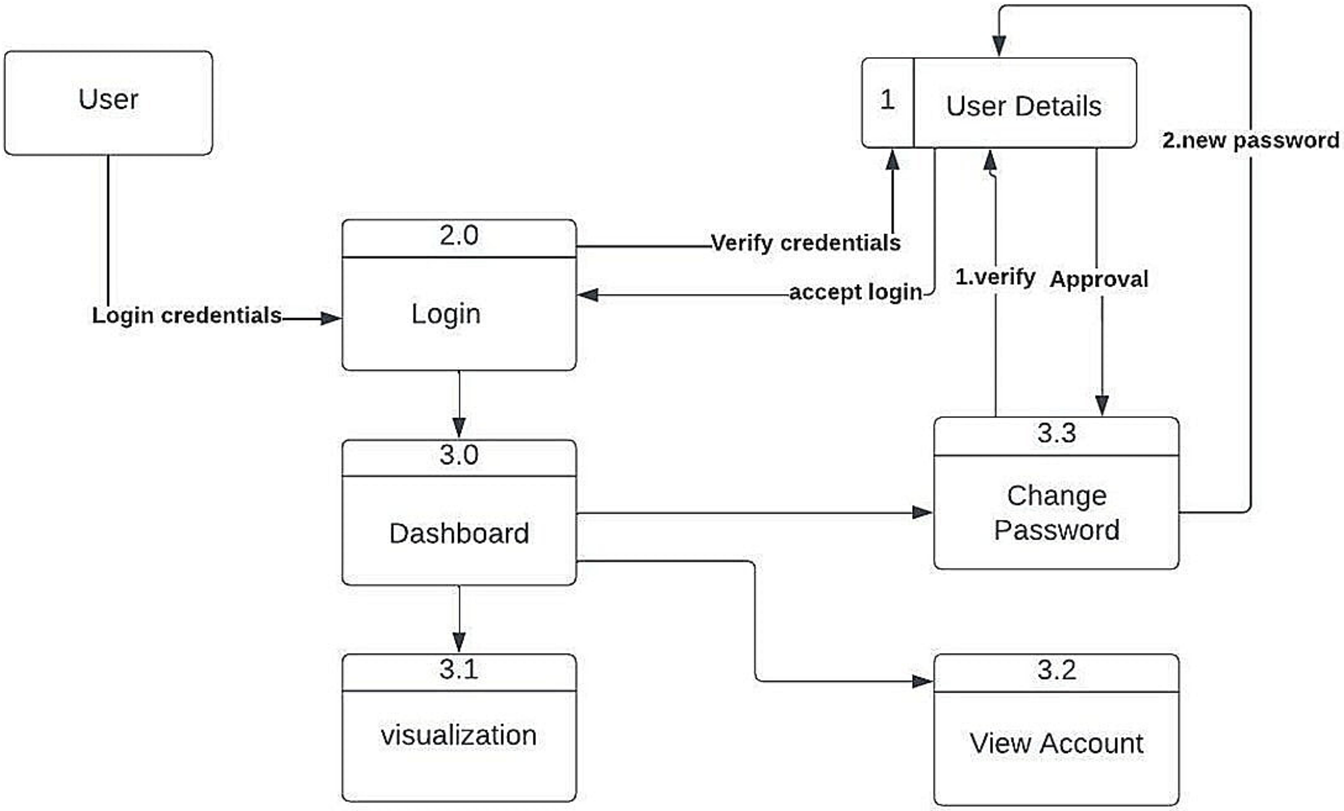
CHAPTER 5

PROJECT DESIGN

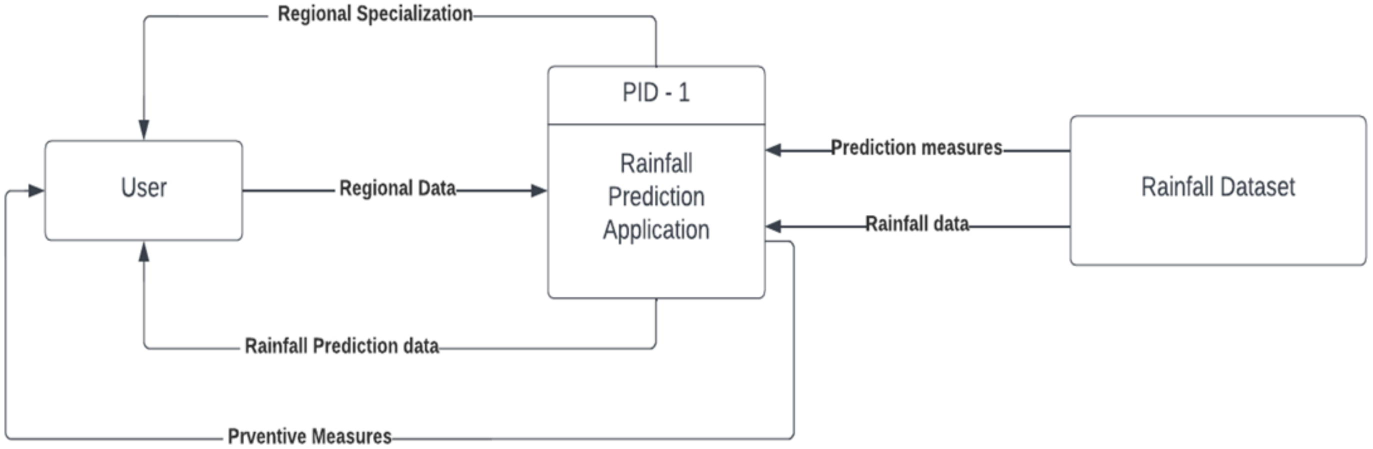
5.1 DATAFLOW DIAGRAMS

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

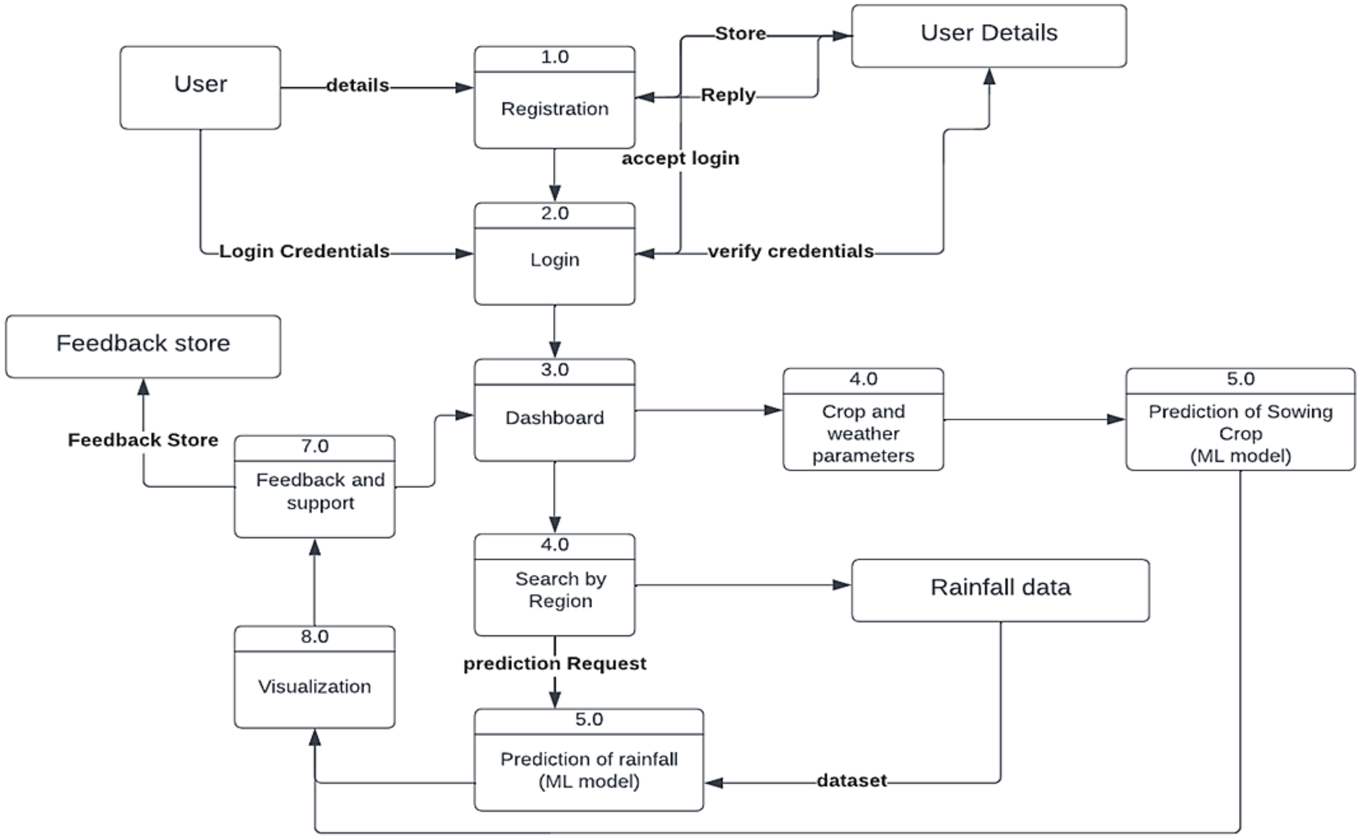
Data flow diagram level 0:



Data flow diagram level1:



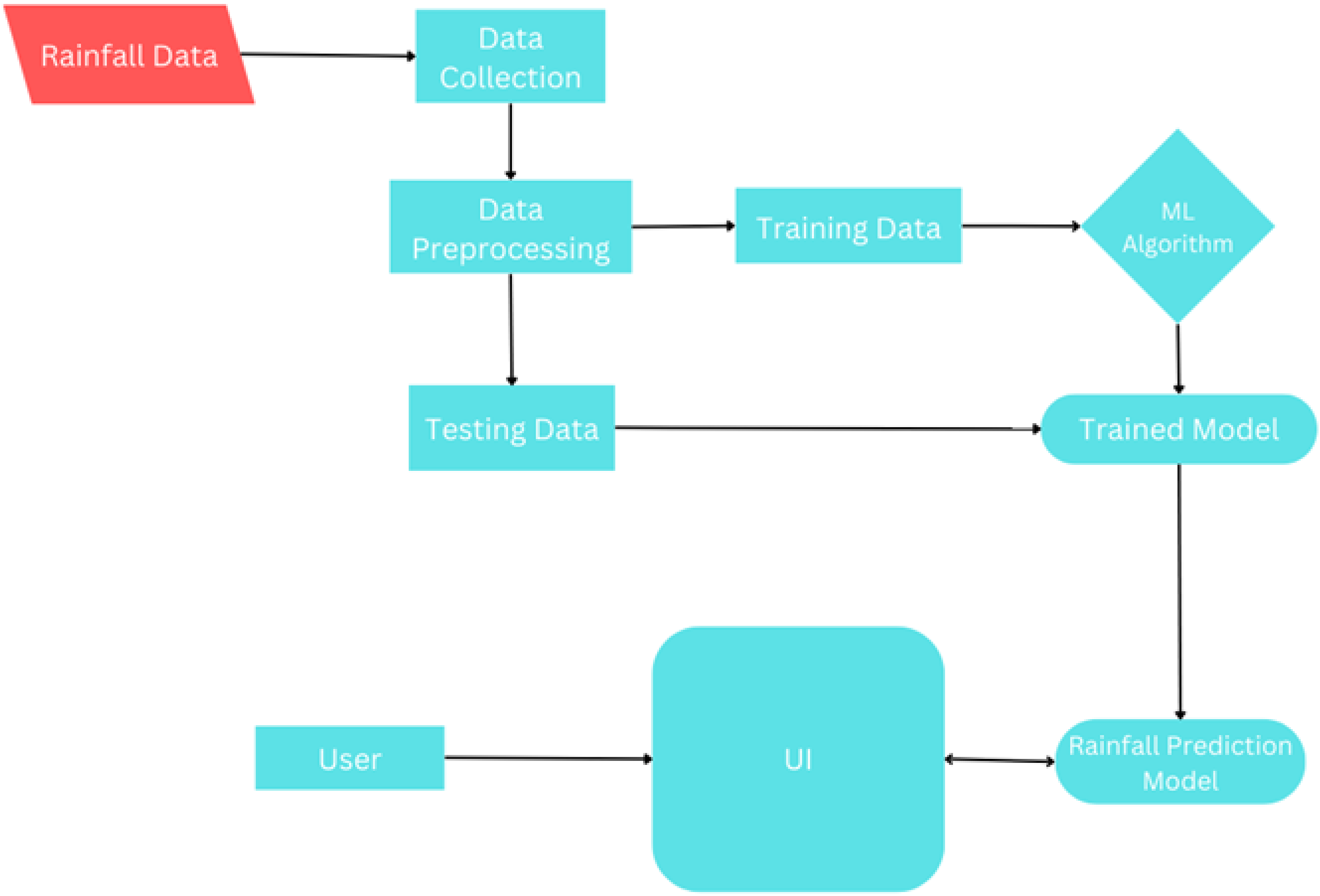
Data flow diagram level2:



5.2 SOLUTION & TECHNICAL ARCHITECTURE

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed, and delivered.



TECHNICAL ARCHITECTURE:

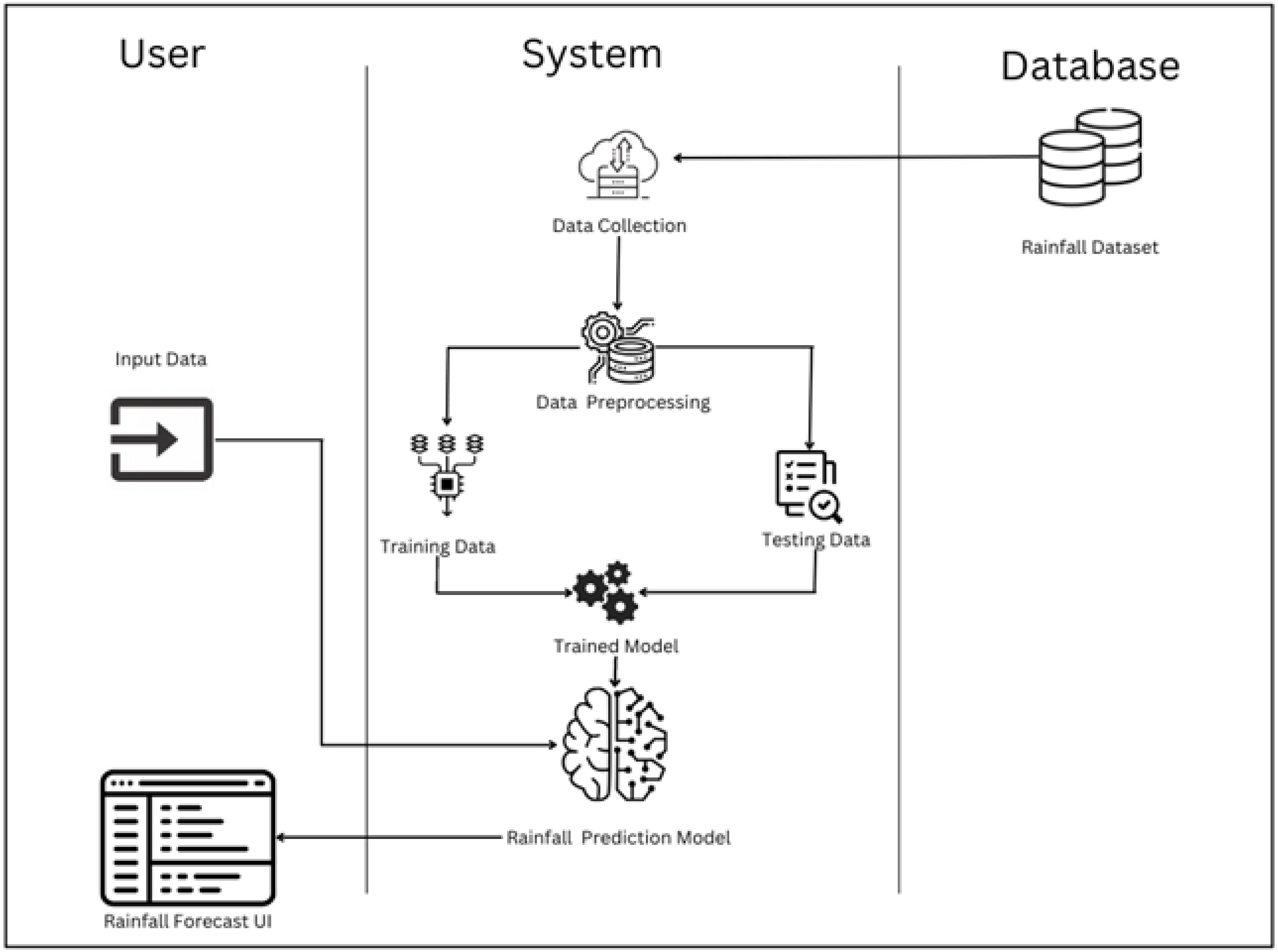


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	User interacts with application by Web UI	HTML, CSS, Bootstrap,JavaScript
2.	Application Logic-1	Integrate website with machine learning	Flask, Python

3.	Application Logic-2	Forecasting rainfall	Machine learning
4.	Application Logic-3	Cloud Application Server	IBM Watson STT service
5.	Database	Data Type, Configurations etc.	MySQL, Pickle,etc
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
7.	External API	Accessing current climate data	IBM Weather API, etc.
8.	Machine Learning Model	Forecasting rainfall and crop recommending	Rainfall Model, Crop Recommending Model.

Table-2: Application Characteristics:

S.No	**Characteristics**	**Description**	**Technology**
1.	Open-Source Frameworks	Open-source frameworks used	Python, Flask, Pickle, Machine learning
2.	Security Implementations	Request authentication using JWT Tokens	HS-256, Encryptions, SSL Certs
3.	Scalable Architecture	Support for Multiple Sample prediction	Pandas, Numpy , Scipy, Scikit-learn
4.	Availability	Availability is increased by Distributed Servers in Cloud VPS	IBM Cloud
5.	Performance	The application is expected to handle multiple predictions per second	Load Balancers, Distributed Server

5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard.	High	Sprint-2
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-2
		USN-3	As a user, I can register for the application through Google Account	I can register & access the dashboard with google	Low	Sprint-2
	Login	USN-4	As a user, I can log into the application by entering email & password	Registered username or email and password	High	Sprint-2

	Dashboard	USN-5	As a user, I can see the current rainfall forecasting	Location and date	Medium	Sprint-3
	Rainfall Prediction	USN-6	As a user, I can enter the input data to predict the rainfall	Location and date	High	Sprint-3
	Crop Recommendation	USN-7	As a user, I can enter the soil and weather parameter to predict the crop recommendation	Soil and weather parameter	High	Sprint-3
Customer Care Executive	Contact	USN-8	As a user, I can ask queries regarding the system	I can clarify my doubts	Medium	Sprint-4
	Feedback	USN-9	As a user, I can send feedback and queries to the Administrator	I can review the application and suggest updates	Low	Sprint-4
Administrator	Login	USN-10	As a Administrator, I can login to the application by entering my email, password, and confirming	I can view and update the system	High	Sprint-1

			my password			
	Rainfall Prediction Model	USN-11	As a Administrator, I train the rainfall prediction model and save the best model	I train the rainfall prediction Model	High	Sprint-1
	Crop Recommendation Model	USN-12	As a Administrator, I train and test the crop recommendation model and save the best model	I train the sowing crop prediction model	High	Sprint-1

CHAPTER 6

PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Product Backlog, Sprint Schedule, and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Rainfall Prediction Model	USN-1	Collecting weather dataset, data preprocessing the data and do a data visualization	5	High	Suresh, Thavaprakash
Sprint-1		USN-2	Train rainfall Prediction model using different machine learning algorithms	5	Medium	Sriraja, Surendra
Sprint-1		USN-3	Test the best model and save best model by pickle library	5	High	Sriraja, Surendra
Sprint-1	Crop Recommendation Model	USN-4	Collecting sowing crop dataset, data preprocessing the data and do a data visualization	5	High	Suresh, Thavaprakash

Sprint-1		USN-5	Train crop recommendation model using different machine learning algorithms	5	Medium	Sriraja, Surendra
Sprint-1		USN-6	Test the best model and save best model by pickle library	5	High	Sriraja, Surendra
Sprint-2	Registration	USN-7	User can register for the application by entering his or her email, password, and confirming the password.	5	Medium	Suresh, Thavaprakash
Sprint-2		USN-8	User will receive confirmation email or message once registered for the application	5	Low	Suresh, Thavaprakash
Sprint-2	Login	USN-9	Enter the username and password to login to the application	5	Medium	Suresh, Thavaprakash
Sprint-2		USN-10	The existing credentials should be used for login on multiple systems	5	Medium	Suresh, Thavaprakash
Sprint-2	Dashboard	USN-11	Forecast the today weather	10	Low	Sriraja, Surendra
Sprint-3	Rainfall Prediction	USN-12	User can enter the weather parameters like min temp, max temp, etc	5	High	Surendra, Thavaprakash

Sprint-3		USN-13	Predict the rainfall and display the result	5	High	Sriraja, Suresh
Sprint-3	Crop Recommendation	USN-14	User can enter the soil parameters like nitrogen, ph. value, etc	5	High	Surendra, Thavaprakash
Sprint-3		USN-15	Predict the sowing crop and display the result	5	High	Sriraja, Suresh
Sprint-4	Testing	USN-16	Test the application	10	High	Surendra, Thavaprakash
Sprint-4	Deploy Model	USN-17	deploy the model in IBM cloud to make user friendly application	10	High	Sriraja, Suresh

Project Tracker, Velocity & Burndown Chart:

Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Date (Planned)**	**Story Points Completed (as on Planned End Date)**	**Sprint Release Date (Actual)**
Sprint-1	30	6 Days	24 Oct 2022	29 Oct 2022	30	30 Oct 2022
Sprint-2	30	6 Days	31 Oct 2022	05 Nov 2022	30	06 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	13 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	20 Nov 2022

Velocity:

We have a 6-day sprint duration, and the velocity of the team is 20 to 30 (points per sprint). Let’s calculate the team’s average velocity (AV) per iteration unit (story points per day)

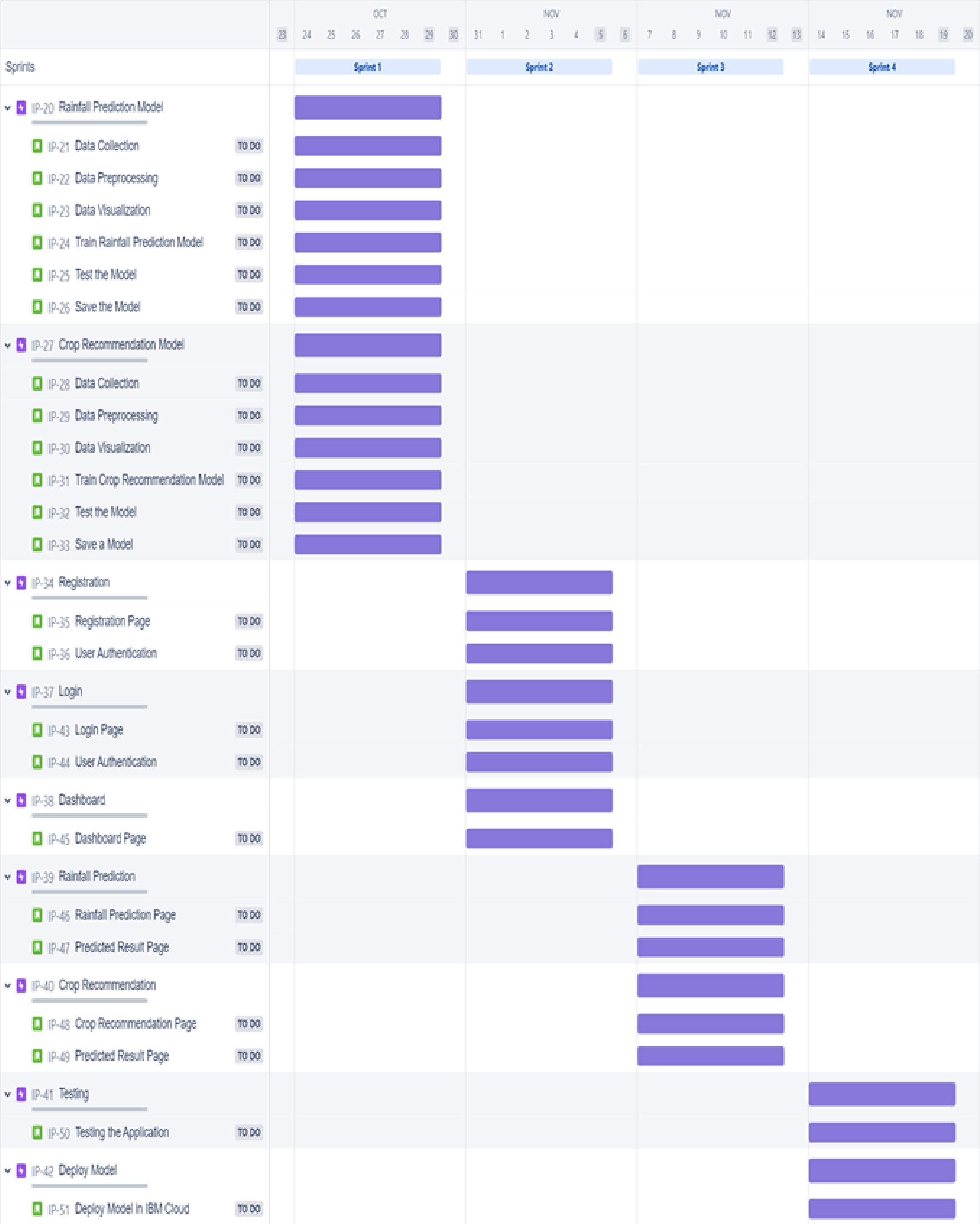
$$AV = \frac{sprint\ duration}{velocity}$$

Sprint	**Average Velocity**
Sprint-1	**5**
Sprint-2	**5**
Sprint-3	**3.33**
Sprint-4	**3.33**

Total Average Velocity = 4.16

6.2 SPRINT DELIVERY SCHEDULE

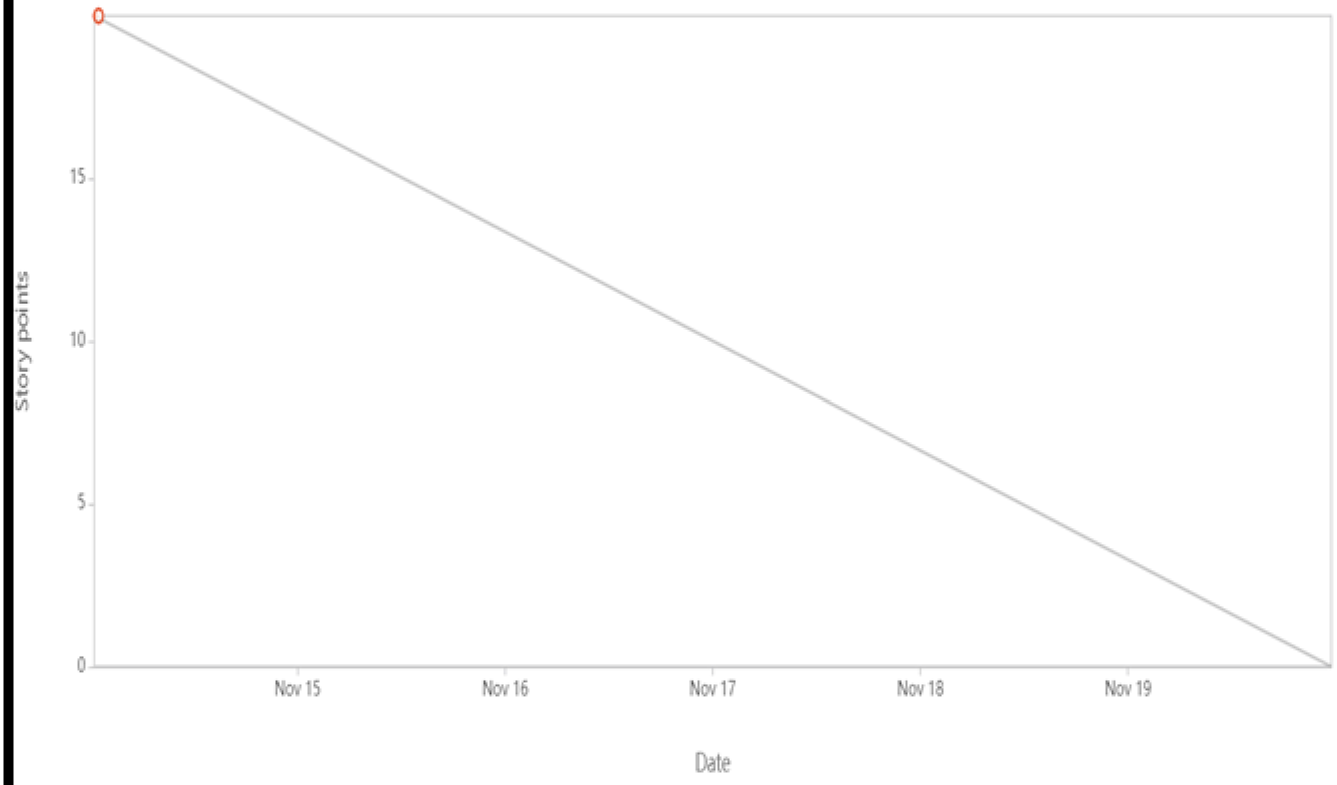
RoadMap



Sprints

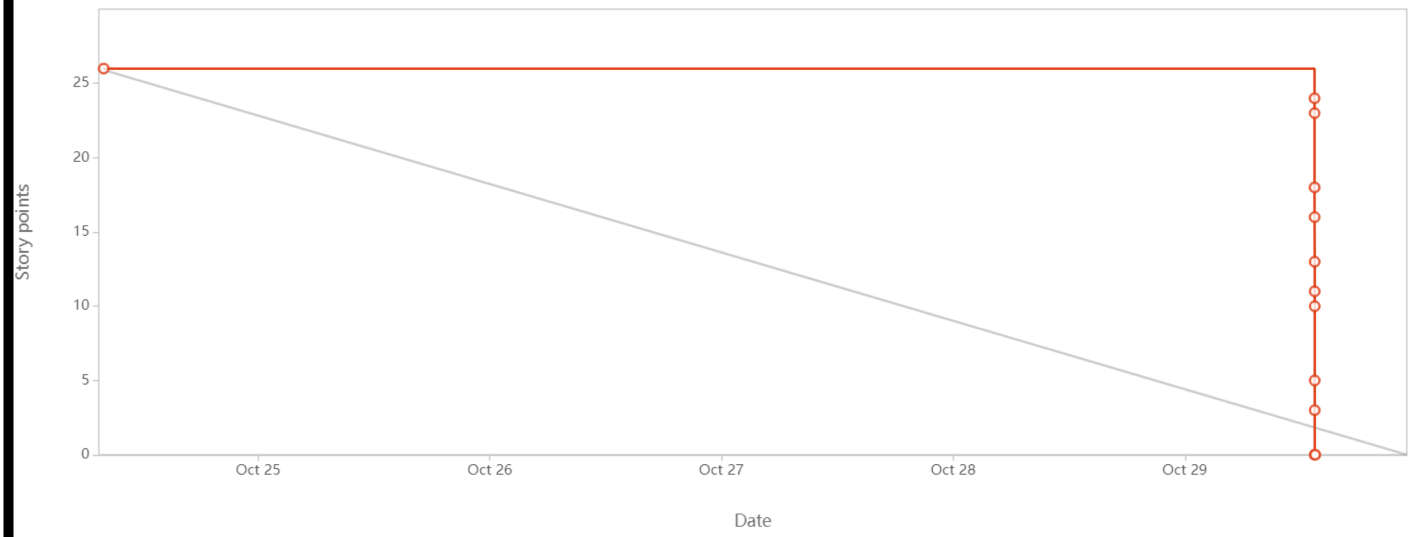
IP-20	Rainfall Prediction Model	
IP-21	Data Collection	TO DO
IP-22	Data Preprocessing	TO DO
IP-23	Data Visualization	TO DO
IP-24	Train Rainfall Prediction Model	TO DO
IP-25	Test the Model	TO DO
IP-26	Save the Model	TO DO
IP-27	Crop Recommendation Model	
IP-28	Data Collection	TO DO
IP-29	Data Preprocessing	TO DO
IP-30	Data Visualization	TO DO
IP-31	Train Crop Recommendation Model	TO DO
IP-32	Test the Model	TO DO
IP-33	Save a Model	TO DO
IP-34	Registration	
IP-35	Registration Page	TO DO
IP-36	User Authentication	TO DO
IP-37	Login	
IP-43	Login Page	TO DO
IP-44	User Authentication	TO DO
IP-38	Dashboard	
IP-45	Dashboard Page	TO DO
IP-39	Rainfall Prediction	
IP-46	Rainfall Prediction Page	TO DO
IP-47	Predicted Result Page	TO DO
IP-40	Crop Recommendation	
IP-48	Crop Recommendation Page	TO DO
IP-49	Predicted Result Page	TO DO
IP-41	Testing	
IP-50	Testing the Application	TO DO
IP-42	Deploy Model	
IP-51	Deploy Model in IBM Cloud	TO DO

Burndown Chart:

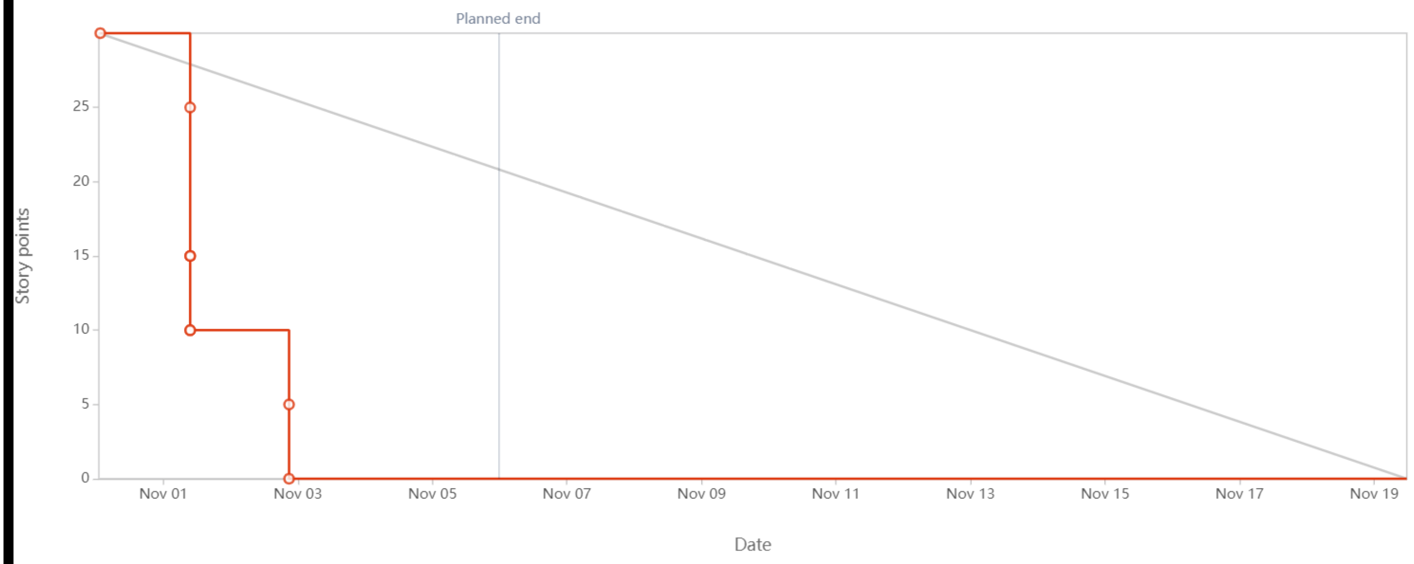


6.3 REPORTS FROM JIRA

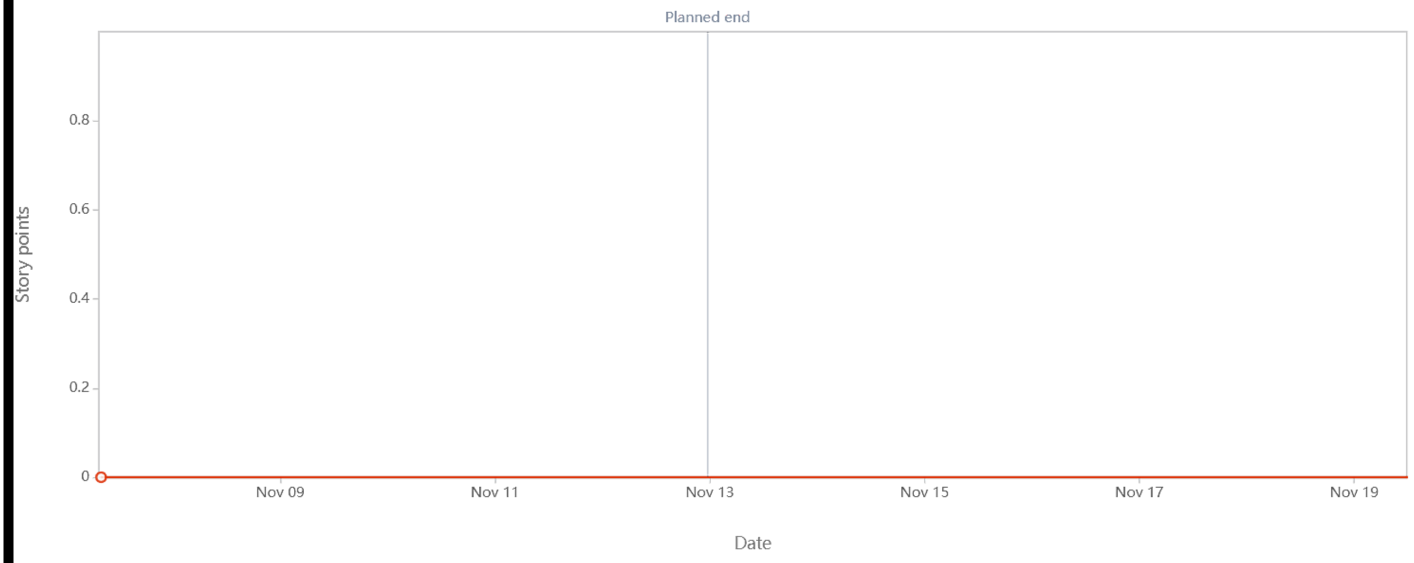
Sprint 1 :



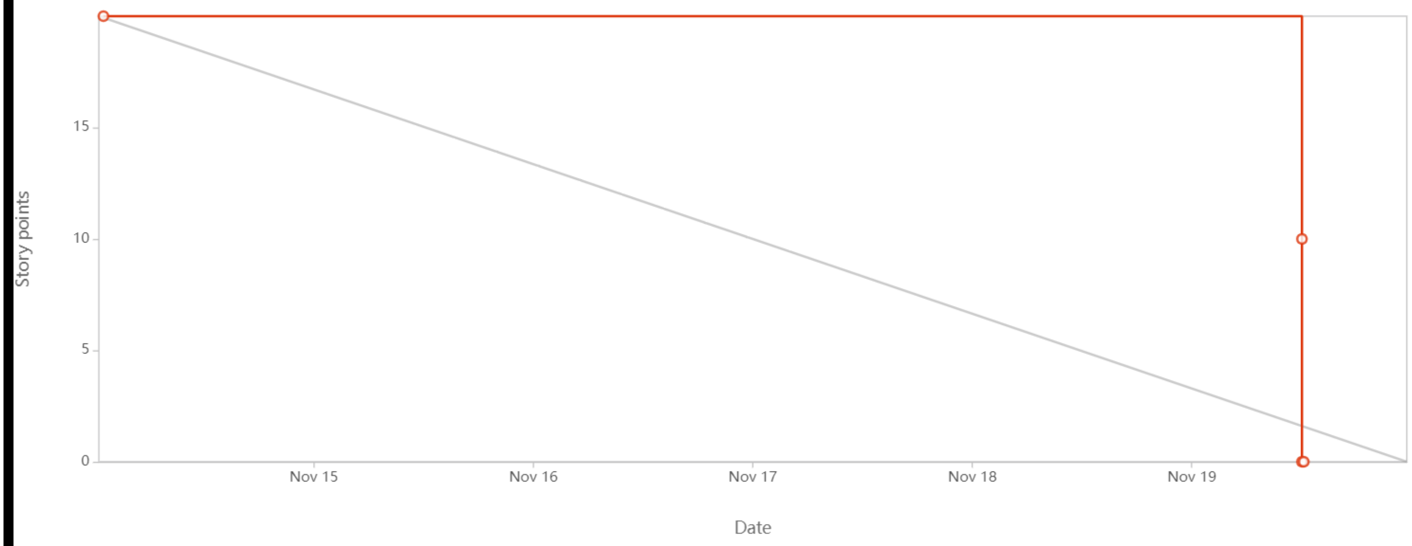
Sprint 2 :



Sprint 3 :



Sprint 4 :



CHAPTER 7

CODING & SOLUTIONING

7.1 FEATURE 1

Rainfall Prediction:

Rainfall prediction is used to predict the rainfall based on the given condition, then it predicted and say whether it rain or not in mm

Rainfall prediction Source Code :

```
def rainfall_prediction():
    if request.method=='POST':
        mintemp=float(request.form['mintemp'])
        meantemp=float(request.form['meantemp'])
        maxtemp=float(request.form['maxtemp'])
        pressure=float(request.form['pressure'])
        preceptions=float(request.form['preceptions'])
        windspeed=float(request.form['windspeed'])

        data=np.array([[0,1,maxtemp,mintemp,meantemp,preceptions,pressure,windspeed]])
        my_prediction=rainfall_model.predict(data)
        final_prediction=my_prediction[0]
        #ws=["static/image/"+final_prediction+".jpg",final_prediction,'myfunc()']
        return render_template('Rainfall.html',rainfall=[final_prediction,"pre()"])
    else:
            return render_template('Rainfall.html',error="Can't predicted Some Error. Try Again")

if __name__ == '__main__':
   app.run(debug = True)
```

7.2 FEATURE 2

Crop Recommendation:

Crop recommendation is used to predict the sowing crop based on the given condtions, it predict and say the sowing crop

Crop Recommendation Source Code :

```
def crop_prediction():
    if request.method=='POST':
        N=int(request.form['nitrogen'])
        P=int(request.form['phosphorous'])
        K=int(request.form['pottasium'])
        ph=float(request.form['ph'])
        rainfall=float(request.form['rainfall'])
        city=request.form.get("city")

        if weather_fetch(city) != None:
            temperature,humidity = weather_fetch(city)
            data=np.array([[N,P,K,temperature,humidity,ph,rainfall]])
            my_prediction=crop_recommedation_model.predict(data)
            final_prediction=my_prediction[0]
            ws=["static/image/"+final_prediction+".jpg",final_prediction,'myfunc()']
            return render_template('crop.html',crops=ws)
        else:
            return render_template('crop.html',crops=[0,0,"error()"])
    else:
            return render_template('crop.html',crops=[0,0,"error()"])
```

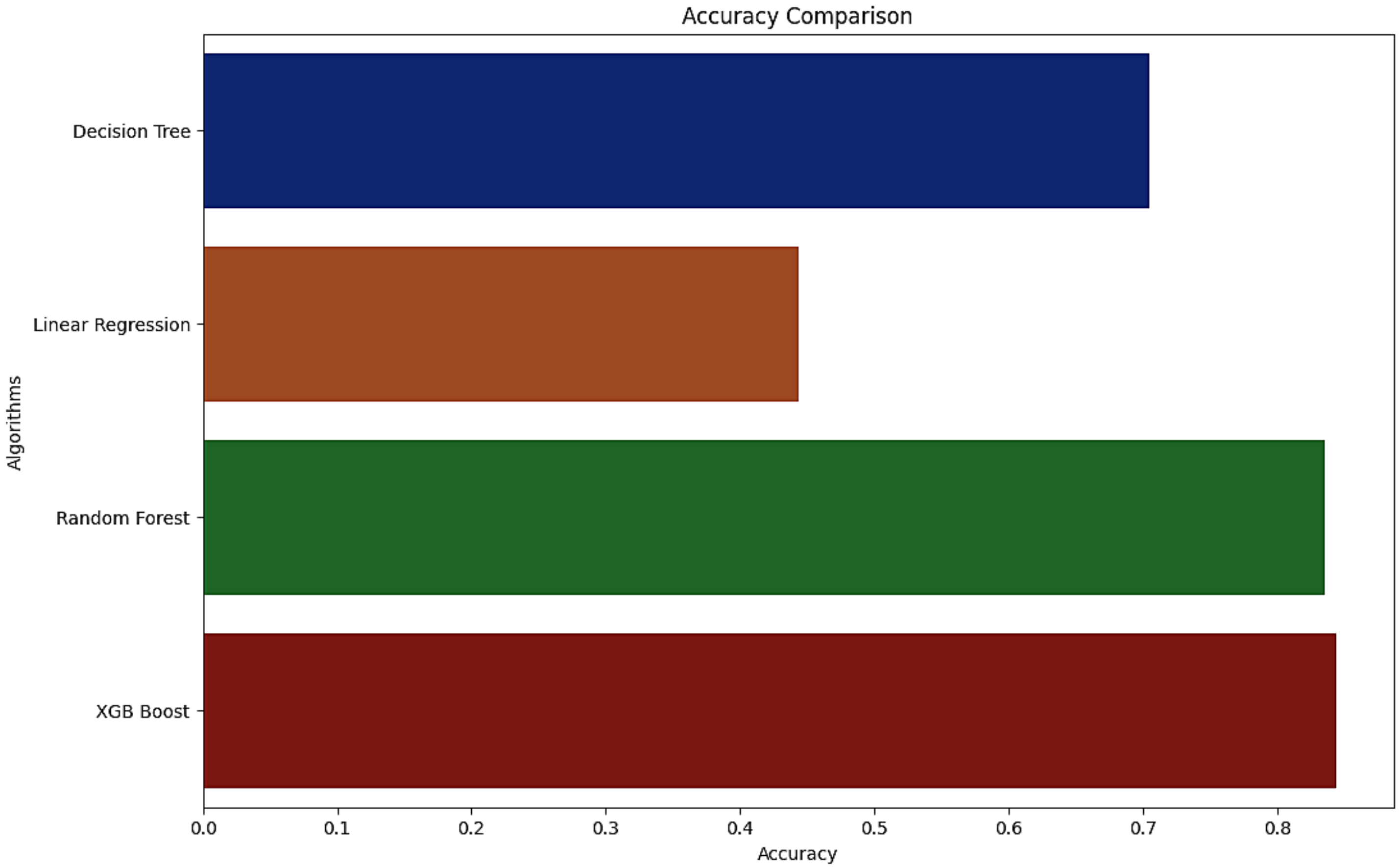
CHAPTER 8

TESTING

8.1 TEST CASES

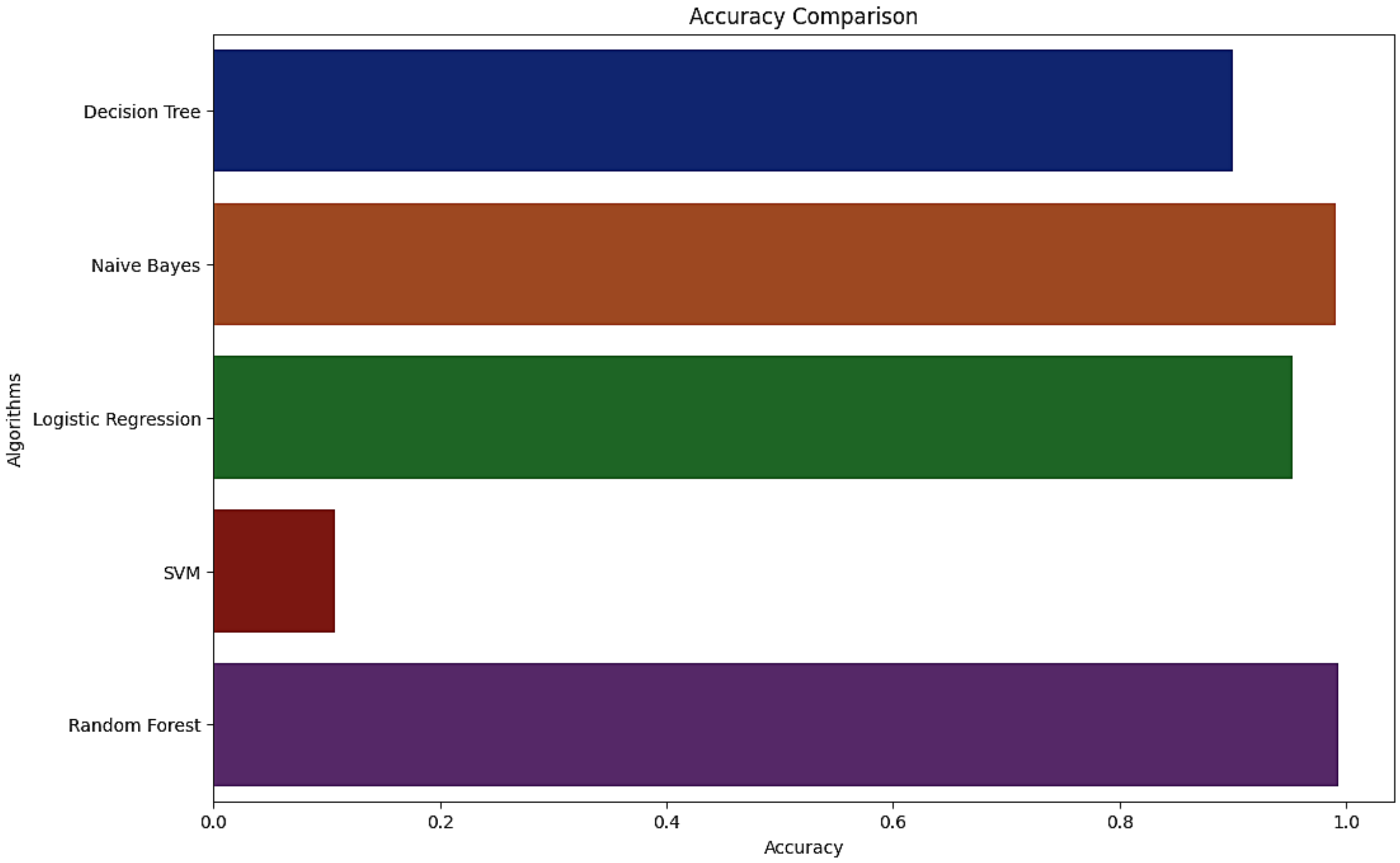
Rainfall Prediction Model

For Analysising for Rainfall Prediction Model, Applying different machine learning algorithm to train and test the model, xgboost shows the higher accuracy score ccompared to other machine learning algorithm



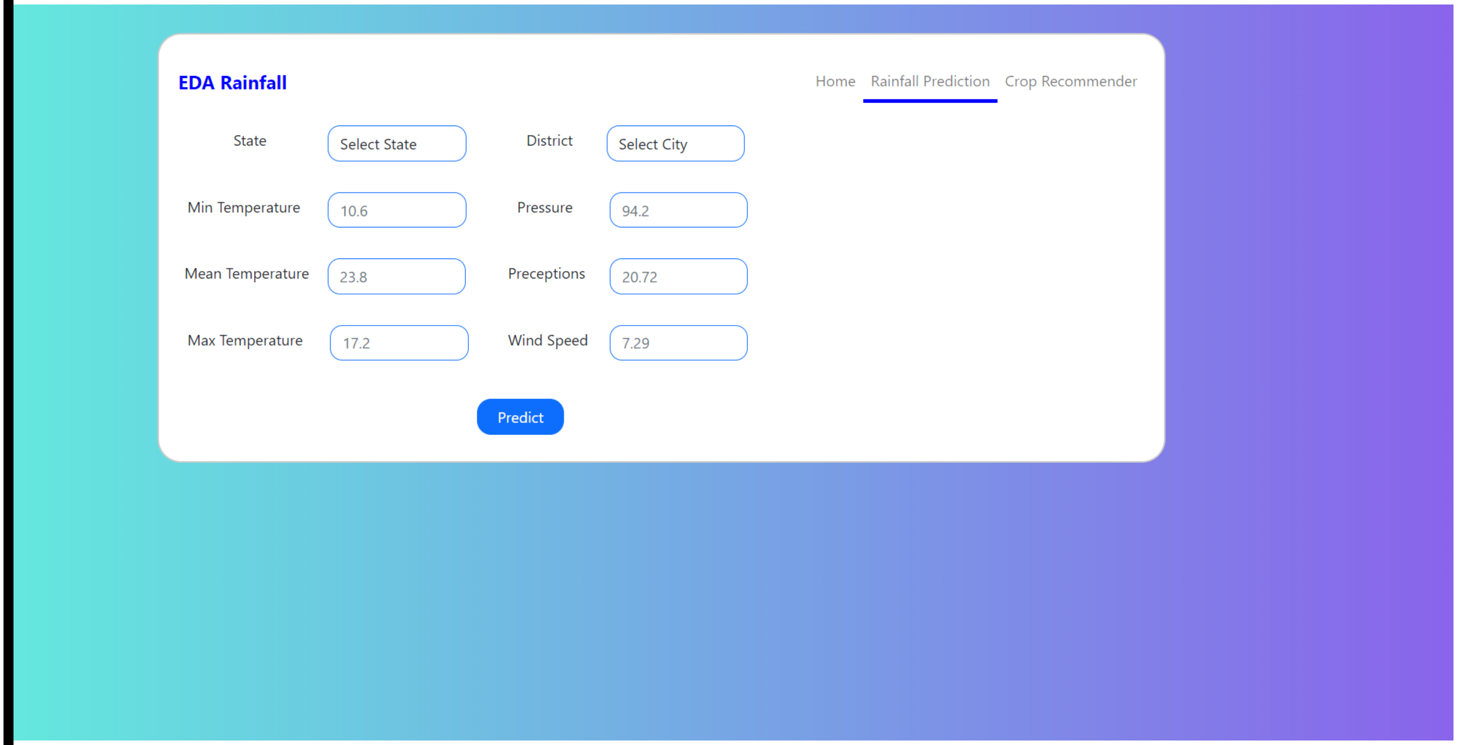
Crop Recomendation Model

For Analysising for Crop Recommendation Model, applying different machine learning algorithm to train and test the model, random forest classification algorithm shows the higher accuracy score ccompared to other machine learning algorithm



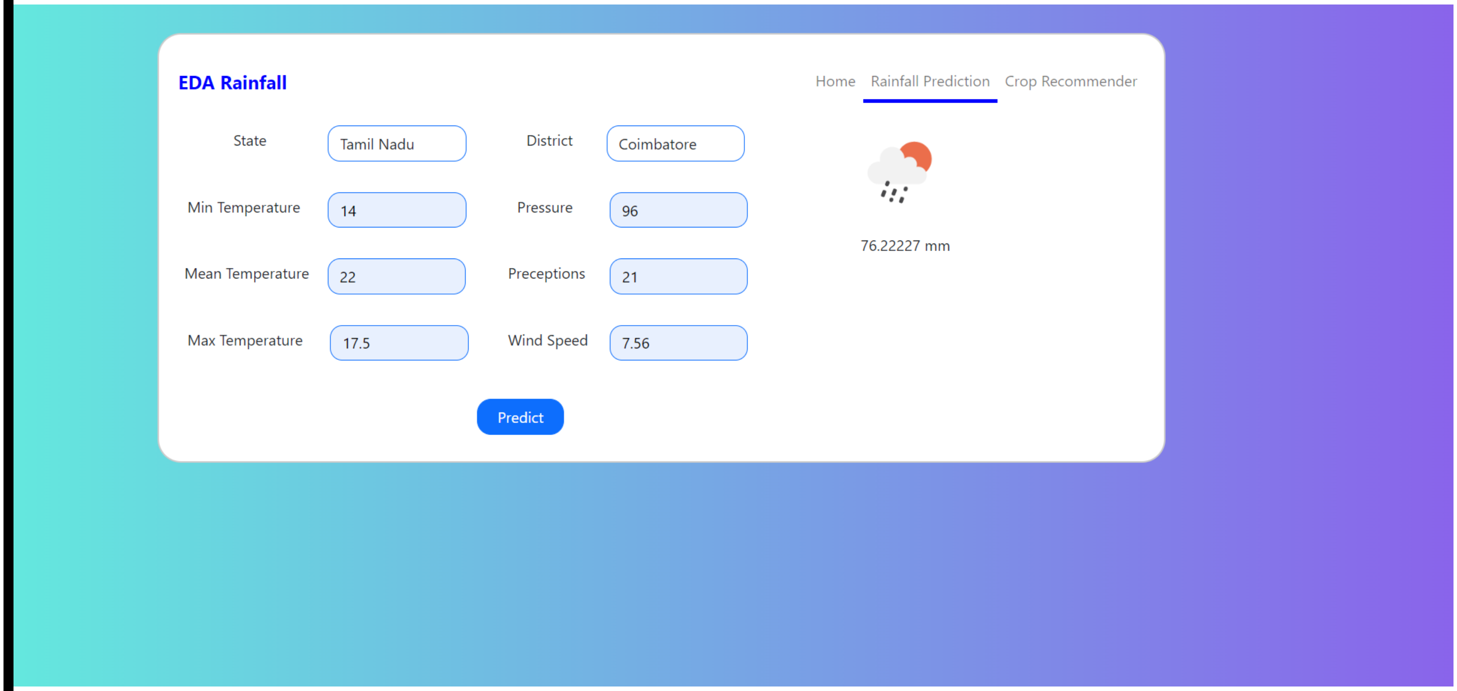
8.2 USER ACCEPTANCE TESTING

Rainfall Prediction Page



State	Select State	District	Select City
Min Temperature	10.6	Pressure	94.2
Mean Temperature	23.8	Preceptions	20.72
Max Temperature	17.2	Wind Speed	7.29

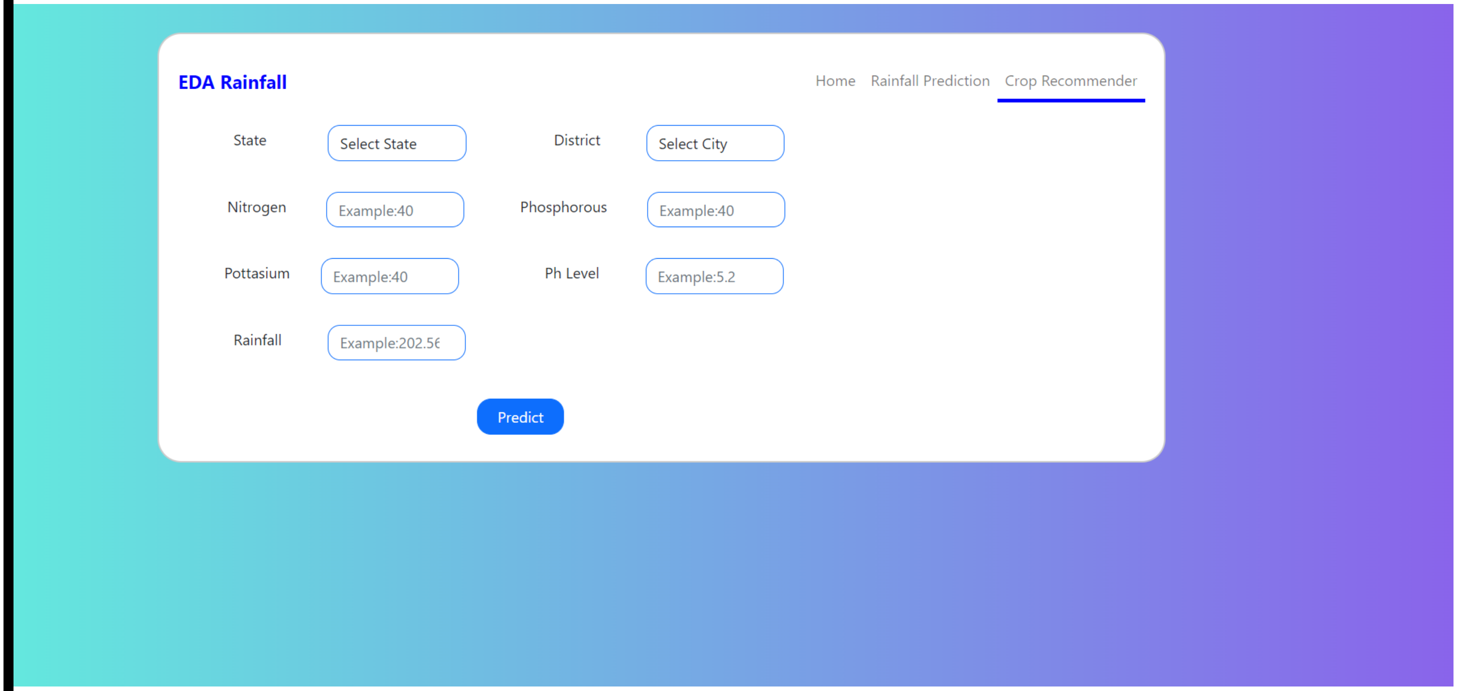
For a given value it predicts as Rainy



State	Tamil Nadu	District	Coimbatore
Min Temperature	14	Pressure	96
Mean Temperature	22	Preceptions	21
Max Temperature	17.5	Wind Speed	7.56

76.22227 mm

Crop Recommendation Page



State: Select State

District: Select City

Nitrogen: Example:40

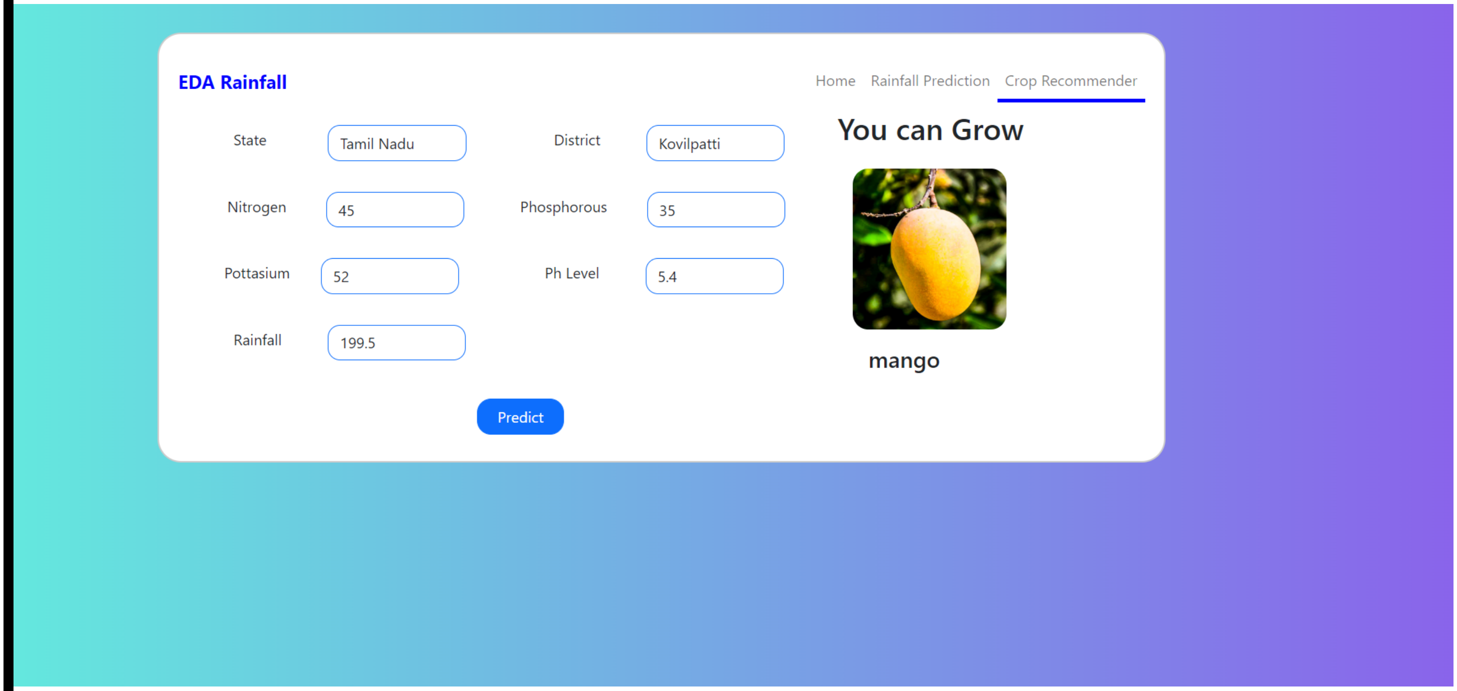
Phosphorous: Example:40

Pottasium: Example:40

Ph Level: Example:5.2

Rainfall: Example:202.5€

It Predict as Mango



State: Tamil Nadu

District: Kovilpatti

Nitrogen: 45

Phosphorous: 35

Pottasium: 52

Ph Level: 5.4

Rainfall: 199.5

You can Grow

mango

CHAPTER 9

RESULTS

9.1 PERFORMANCE METRICS

For rainfall prediction model, applying different machine learning algorithm to train and test the model, the xgboost machine learning algorithm shows the higher accurracy score compare to other machine learning algorithm like decison tree, Naive Bayes, Radom Forest,etc and saving the xgboost model as pickle for future use

For sowing crop recommendation model, applying different machine learning algorithm to train and test the model, the Random Forest Classification machine learning algorithm shows the higher accurracy score compare to other machine learning algorithm like decison tree, Naive Bayes, xgboost,etc and saving the random forest model for future use

Then use this saved model, using flask create a responsive web page to access the prediction and test the flask with postman appication with multiple user, it show good testing results

CHAPTER 10

ADVANTAGES & DISADVANTAGES

10.1 ADVANTAGES

- As Weather conditions have been changing for the time being this helps people to know about the rainfall prediction.
- To avoid unnecessary floods by opening dams with the help of rainfall prediction.
- Farmers and fisherman will get the most advantage of these rainfall details so that we they can plan accordingly
- During the monsoon days it helps the government to find the evacuation areas to avoid loss of human life and costly things

10.2 DISADVANTAGES

- As the data was collected from limited places so it helps only for the people who located in those areas.
- In case the data was collected being wrong the algorithm will produce the wrong prediction.
- As of now have collecting only a limited number of data set, In feature, we will make the algorithm to work worldwide

CHAPTER 11

CONCLUSION

Rainfall prediction and Crop recommending is successfully predicted and also found the efficient algorithm from to train the model and obtained the most efficient output by best model From Analysing different algorithm, the Random Forest Algorithm show the best accuracy score which will outcome with percentage 99% for the Model Crop Recommending Which Identify the suitable crop based on weather and soil parameters and For Rainfall prediction XG Boost algorithm show almost 88% . Deployment the Model in IBM Cloud to easily accessed by user. Farmer use this application to get the crop sowing based on weather and find the weather changes

CHAPTER 12

FUTURE SCOPE

In future developing the android application based on this method can make the user to use this easily in smart phone and help the user to understand the weather condition, he can find the correct sowing crop based on weather and soil parameters. And also, there are many modules can be included like Species management (Species Breeding and Species Recognition), Filed condition management (soil management and water management), Crop Management (Yield Prediction, Crop Quality, Disease Detection and weed Detection).

CHAPTER 13

APPENDIX

SOURCE CODE

SAMPLE DATA PREPROCESSING OF RAINFALL DATA

```
 # Importing the required libraries
 import pandas as pd
 import numpy as np
 import warnings warnings.filterwarnings("ignore")

 rain_df= pd.read_csv("Collected Dataset/Rainfall.csv")
 rain_df.head()

 rain_df=rain_df[["SUBDIVISION","YEAR","JAN","FEB","MAR","APR","MAY","JUN","JUL","AUG","SEP","OCT","NOV","
DEC"]]
 rain_df.head()

 rain_df.count()
 rain_df.info()
 rain_df.isnull().sum()
 rain_df=rain_df.fillna(rain_df.mean())
 ain_df.isnull().sum()

rain_df.rename(columns={"JAN":1,"FEB":2,"MAR":3,"APR":4,"MAY":5,"JUN":6,"JUL":7,"AUG":8,"SEP":9,"OCT":10,"NO
V" :11,"DEC":12}, inplace = True)
 rain_df.head()

 maxtemp_df=pd.read_csv("Collected Dataset/Max_temp.csv")
 maxtemp_df.head()

 maxtemp_df.count()
 maxtemp_df.isnull().sum()

maxtemp_df.rename(columns={"Year":"YEAR","Jan":1,"Feb":2,"Mar":3,"Apr":4,"May":5,"Jun":6,"Jul":7,"Aug":8,"Sep":9,
"Oct" :10,"Nov":11,"Dec":12}, inplace = True)
 maxtemp_df.head()

 mintemp_df=pd.read_csv("Collected Dataset/Min_temp.csv")
```

```
mintemp_df.head()
mintemp_df.count()
mintemp_df.isnull().sum()

mintemp_df.rename(columns={"Year":"YEAR","Jan":1,"Feb":2,"Mar":3,"Apr":4,"May":5,"Jun":6,"Jul":7,"Aug":8,"Sep":9,"
Oct" :10,"Nov":11,"Dec":12}, inplace = True)
```

SAMPLE CODE FOR RAINFALL PREDICTION MODEL

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.metrics
import classification_report
from sklearn
import metrics
from sklearn
import tree
import warnings
warnings.filterwarnings('ignore')

rain_df=pd.read_csv('Final Dataset/rainfall_prediction.csv')
rain_df.head()
rain_df.tail()
rain_df.info()

rain_df.SUBDIVISION.unique()

rain_df.describe() rain_df.corr()
rain_df.isnull().sum()rain_df.info()
sns.heatmap(s,annot=True)
rain_df.head()
from sklearn.preprocessing import LabelEncoder

lab = LabelEncoder()

rain_df.SUBDIVISION = lab.fit_transform(rain_df.SUBDIVISION)
rain_df.head()

rain_df.SUBDIVISION.unique()
feature=rain_df[["SUBDIVISION","MONTH","MAX_TEMP","MIN_TEMP","MEAN_TEMP","PRECEPTIONS","PRESSUR
E" ,"WIND_SPEED"]]
target=rain_df["RAINFALL"]
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(feature,target,test_size=0.2,random_state=2)
X_train

from sklearn.tree import DecisionTreeRegressor
from sklearn.metrics import roc_auc_score,classification_report,mean_squared_error,r2_score
# create a regressor object dtregressor = DecisionTreeRegressor(random_state = 0)
# fit the regressor with X and Y data
dtregressor.fit(X_train, y_train)
```

SAMPLE CODE FOR CROP RECOMMENDATION MODEL

```
 import pandas as pd
 import numpy as np
 import seaborn as sns
 import matplotlib.pyplot as plt
 from sklearn.metrics import classification_report
 from sklearn import metrics
 from sklearn import tree
 import warnings
 warnings.filterwarnings('ignore')

 crop_df=pd.read_csv('Final Dataset/crop_recommendation.csv')
 crop_df.head()
 crop_df.tail()
 crop_df.info()
 crop_df.describe()
 crop_df["label"].value_counts()
 crop_df.dtypes
 crop_df.corr()
 crop_df.isnull().sum()
 s=crop_df.corr()
 s
 sns.heatmap(s,annot=True)
 feature=crop_df[['N','P','K','temperature','humidity','ph','rainfall']]
 target=crop_df['label']
 from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(feature,target,test_size=0.2,random_state=2)

 from sklearn.tree import DecisionTreeClassifier
```

```
DecisionTree=DecisionTreeClassifier(criterion='entropy',max_depth=5,random_state=2)
DecisionTree.fit(X_train,y_train)
 predicted=DecisionTree.predict(X_test)
 x=metrics.accuracy_score(y_test,predicted)
 acc.append(x)
 model.append('Decision Tree')

 print("Accuracy[Decision Tree]:",x*100)
 print(classification_report(y_test,predicted))

 from sklearn.model_selection import cross_val_score
score=cross_val_score(DecisionTree,feature,target,cv=5)
 score
```

SAMPLE CODE FOR LOGIN PAGE

```
 <!DOCTYPE html>
<html>
    <head>
        <title>Welcome Page</title>
        <!--link href="../Static/css/bootstrap.min.css" rel="Stylesheet" type="text/css">
        <link href="../Static/css/Style.css" rel="Stylesheet" type="text/css"-->
        <link href="{{ url_for('static', filename='css/bootstrap.min.css') }}"  rel= "stylesheet"text='text/css'/>
        <link href="{{ url_for('static', filename='css/style.css') }}" rel='stylesheet' type='text/css' />
        <style>
            .wr{
                background-color: rgb(250, 175, 175); margin:0% 0% 3% -7%; padding: 1.5%;text-align: center;
            }
        </style>
    </head>
    <body class="bg-info">
        <div class="card" style="max-width:50%; margin:2% 2% 2% 20%;border-radius: 25px; ">
            <div class="row">
                <div class="col-6"><img class="card-img-top" src="{{ url_for('static', filename='image/login.jpg')
}}" alt="Card image cap"style="border-radius: 25px 0px 0px 25px;margin:-0.3%;"></div>
```

```
                <div class="col-6">
                    <h1 style="padding-left:20%;font-size:400%;">LOGIN</h1>
                    <br>
                    <div class="form-group">
                        <p class={{wrong[1]}}>{{wrong[0]}}</p>
                    </div>
                    <form method="post" action="{{ url_for('welcome') }}">
                        <div class="form-group w-75">
                            <label for="email" style="color:blue;"><b>Email</b></label>
                            <input type="email" style="margin:5%; border-radius:13px;" class="form-control border-primary" id="email" placeholder="Enter email" name="email">
                        </div>
                        <div class="form-group w-75">
                            <label for="password" style="color:blue;"><b>Password</b></label>
                            <input type="password" style="margin:5%; border-radius: 13px;" class="form-control border-primary" id="pass" placeholder="Enter password" name="password"><br>
                        </div>
                        <button type="submit" class="btn btn-primary" style="margin-left:30%; padding:3% 10% 3% 10%; border-radius: 15px; ">Login</button>
                    </form>
                    <br>
                    <div>
                        <p style="color:blue; margin-left: 40%;"><b>OR</b></p>
                        <p style="color:grey; margin-left: 20%;">Don't have an account ?</p>
                        <a href="signin" style="text-decoration:none;color:blue; margin-left: 40%;"><b>SignUp</b></a>
                    </div>
                </div>

    </body>
</html>
```

SAMPLE CODE FOR REGISTRATION PAGE

```
 <!DOCTYPE html>
<html>
    <head>
        <title>SignIn</title>
        <!--link href="../Static/css/bootstrap.min.css" rel="Stylesheet" type="text/css">
        <link href="../Static/css/Style.css" rel="Stylesheet" type="text/css"-->
        <link href="{{ url_for('static', filename='css/bootstrap.min.css') }}"  rel= "stylesheet"text='text/css'/>
        <link href="{{ url_for('static', filename='css/style.css') }}" rel='stylesheet' type='text/css' />
    </head>
    <body class="bg-info">
        <div class="card" style="max-width:58%; margin:2% 2% 2% 20%;border-radius: 25px;">
            <div class="row">
                <div class="col-6"><img class="card-img-top" src="{{ url_for('static', filename='image/login.jpg') }}" alt="Card image cap"style="border-radius: 25px 0px 0px 25px;margin:-0.3%;"></div>
                <div class="col-6">
                    <h1 style="padding-left:20%; padding-top:10px;font-size:350%;">SIGNUP</h1>
                    <br>
                    <form method="post" action="{{ url_for('signins') }}">
                        <div class="form-group w-75">
                            <label for="name" style="color:blue;"><b>Name</b></label>
                            <input type="text" style="margin:5%; border-radius:10px;" class="form-control border-primary" id="name" placeholder="Enter a Name" name="name">
                        </div>
                        <div class="form-group w-75">
                            <label for="email" style="color:blue;"><b>Email</b></label>
                            <input type="email" style="margin:5%; border-radius:13px;" class="form-control border-primary" id="email" placeholder="Enter a Email" name="email">
                        </div>
                        <div class="form-group w-75">
                    <label for="mobile" style="color:blue;"><b>Mobile Number</b></label>
                    <input type="text" style="margin:5%; border-radius:10px;" class="form-control border-
primary" id="mobile" placeholder="Enter a Mobile Number" name="mobile">
                  </div>
                  <div class="form-group w-75">
                    <label for="password" style="color:blue;"><b>Password</b></label>
                    <input type="password" style="margin:5%; border-radius: 13px;" class="form-control border-
primary" id="pass" placeholder="Enter password" name="password">
                  </div>
                  <button type="submit" class="btn btn-primary" style="margin-left:30%; padding:3% 10% 3%
10%; border-radius: 15px; " onclick="myFunction()">SignUp</button>
                </form>
                <br>
                <div>
                  <p style="color:blue;margin-left:40%;"><b>OR</b></p>

                  <p style="margin-left:20%;">Already have an account? <b><a href="login" style="text-
decoration:none;color:blue;"> SignIn</a></b></p>
                </div>
              </div>
            </div>
          </div>
          <script>
            function myFunction() {
                alert('Created Successfull');
            }
          </script>

        </body>
      </html>
```

SAMPLE CODE FOR HOME PAGE

```
 <!DOCTYPE html>
<html>
    <head>
        <title>Home Page</title>
        <!--link href="../Static/css/bootstrap.min.css" rel="Stylesheet" type="text/css">
        <link href="../Static/css/Style.css" rel="Stylesheet" type="text/css"-->
        <link href="{{ url_for('static', filename='css/bootstrap.min.css') }}"  rel= "stylesheet"text='text/css'/>
        <link href="{{ url_for('static', filename='css/style.css') }}" rel='stylesheet' type='text/css'>
    </head>
    <body class="bg-info">
        <div class="container">
            <nav class="navbar navbar-expand-lg navbar-light">
                <div class="collapse navbar-collapse" id="navbarSupportedContent">
                    <ul class="navbar-nav mr-auto">
                        <li class="nav-item active">
                            <a class="navbar-brand" href="#" style="color:blue ;"><b>EDA Rainfall</b></a>
                        </li>
                    </ul>
                    <ul class="nav navbar-nav ml-auto w-100 justify-content-end">
                        <li class="nav-item active" Style="border-bottom: 4px solid blue;" >
                            <a class="nav-link" href="{{ url_for('home') }}">Home</a>
                        </li>
                        <li class="nav-item active">
                            <a class="nav-link" href="rainfall">Rainfall Prediction</a>
                        </li>
                        <li class="nav-item active">
                            <a class="nav-link" href="crop">Crop Recommender</a>
                        </li>
                    </ul>
                </div>
            </nav>
            <div class="row">
                <div class="col-10"><p id="time" style="font-size: 50px;">12.00<span id="am-pm">PM</span></p></div>
                <div class="col-2"><h3 class="time-zone" id="time-zone">Asia/Kolkata</h3>{{city}}<div id="country" class="country">IN</div></div>
                <div class="col-4">
                    <div class="card" Style="border-radius: 25px; padding:2%;">
                        <div class="row">
                            <div class="col-12" id="date" style="font-size:25px;">Monday, 25 May</div>
                            <div class="col-8"><div id="current-weather-icon"></div></div>
                            <div class="col-4"><div id="current-weather-items"></div></div>
                    </div>
                    </div>
                </div>
                <div class="col-8"><br><br><br></div>
                <div class="col-12"><br><br><br></div>
                <div class="col-3">
                    <div class="row">
                        <div class="col-12" id="date0" style="font-size:25px;">Monday, 25 May</div>
                        <div class="col-6"><div id="current-weather-icon0"></div></div>
                        <div class="col-6"><div id="current-weather-items0"></div></div>
                    </div>
            </div>
                <div class="col-3">
                        <div class="row">
                            <div class="col-12" id="date1" style="font-size:25px;">Monday, 25 May</div>
                            <div class="col-6"><div id="current-weather-icon1"></div></div>
                            <div class="col-6"><div id="current-weather-items1"></div></div>
                        </div>
                </div>
                <div class="col-3">
                    <div class="row">
                        <div class="col-12" id="date2" style="font-size:25px;">Monday, 25 May</div>
                        <div class="col-6"><div id="current-weather-icon2"></div></div>
                        <div class="col-6"><div id="current-weather-items2"></div></div>
                    </div>
                </div>
                <div class="col-3">
                    <div class="row">
                        <div class="col-12" id="date3" style="font-size:25px;">Monday, 25 May</div>
                        <div class="col-6"><div id="current-weather-icon3"></div></div>
                        <div class="col-6"><div id="current-weather-items3"></div></div>
                    </div>
                </div>
                <div class="col-12"><br><br></div>
                <div class="col-3">
                    <div class="row">
                        <div class="col-12" id="date4" style="font-size:25px;">Monday, 25 May</div>
                        <div class="col-6"><div id="current-weather-icon4"></div></div>
                        <div class="col-6"><div id="current-weather-items4"></div></div>
                    </div>
                </div>
                <div class="col-3">
                    <div class="row">
                        <div class="col-12" id="date5" style="font-size:25px;">Monday, 25 May</div>
                        <div class="col-6"><div id="current-weather-icon5"></div></div>
                        <div class="col-6"><div id="current-weather-items5"></div></div>
                    </div>
                </div>
                <div class="col-3">
                    <div class="row">
                        <div class="col-12" id="date6" style="font-size:25px;">Monday, 25 May</div>
                        <div class="col-6"><div id="current-weather-icon6"></div></div>
```

```
                        <div class="col-6"><div id="current-weather-items6"></div></div>
                    </div>
                </div>
                <div class="col-3">
                    <div class="row">
                        <div class="col-12" id="date7" style="font-size:25px;">Monday, 25 May</div>
                        <div class="col-6"><div id="current-weather-icon7"></div></div>
                        <div class="col-6"><div id="current-weather-items7"></div></div>
                    </div>
                </div>



                </div>
                </div>
                </div>
            </div>
        </div>
        <script src="https://cdnjs.cloudflare.com/ajax/libs/moment.js/2.29.1/moment.min.js" integrity="sha512-qTXRIMyZIFb8iQcfjXWCO8+M5Tbc38Qi5WzdPOYZHIlZpzBHG3L3by84BBBOiRGiEb7KKtAOAs5qYdUiZiQNNQ==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>
        <script href="../Static/css/bootstrap.bundle.min.js"></script>
        <script src="{{ url_for('static', filename='script/forecast.js') }}"></script>

    </body>
</html>
```

SAMPLE CODE FOR RAINFALL PREDICTION PAGE

```
 <!DOCTYPE html>
<html>
    <head>
    <title>Home Page</title>
    <!--link href="../Static/css/bootstrap.min.css" rel="Stylesheet" type="text/css">
    <link href="../Static/css/Style.css" rel="Stylesheet" type="text/css"-->
    <link href="{{ url_for('static', filename='css/bootstrap.min.css') }}"  rel= "stylesheet"text='text/css'/>
    <link href="{{ url_for('static', filename='css/style.css') }}" rel='stylesheet' type='text/css'>
    <script type="text/JavaScript" src="{{ url_for('static', filename='script/cities.js') }}"></script>
    <script>
        function pre(){
            document.getElementById('icons').innerHTML=`
            <image style="width:80%;" src="http://openweathermap.org/img/wn/10d@4x.png"></image>
                <p style="margin-left:15%;">{{rainfall[0]}} mm</p>`;}
    </script>
</head>
<body class="bg-info" onload={{rainfall[1]}}>
    <div class="container">
        <nav class="navbar navbar-expand-lg navbar-light">
            <div class="collapse navbar-collapse" id="navbarSupportedContent">
                <ul class="navbar-nav mr-auto">
                    <li class="nav-item active">
                        <a class="navbar-brand" href="#" style="color:blue ;"><b>EDA Rainfall</b></a>
                    </li>
                </ul>
                <ul class="nav navbar-nav ml-auto w-100 justify-content-end">
                    <li class="nav-item active" >
                        <a class="nav-link" href="{{ url_for('home') }}">Home</a>
                    </li>
                    <li class="nav-item active" Style="border-bottom: 4px solid blue;">
                        <a class="nav-link" href="rainfall">Rainfall Prediction</a>
                    </li>
                    <li class="nav-item active">
                        <a class="nav-link" href="crop">Crop Recommender</a>
                    </li>
```

```
          </ul>
        </div>
      </nav>
      <div class="row">
        <div class="col-8">
          <div class="row">
            <div class="col-6">
              <form method="POST" action="{{url_for('rainfall_prediction')}}">
              <div class="form-group d-flex">
                <label for="State" style="padding:6% 15% 6% 18%;">State</label>
                <select style="margin:5%; border-radius: 13px; width:45%;" class="form-control border-primary" onchange="print_city('state', this.selectedIndex);" id="state" class="form-control" name="state" required></select>
                <script language="Javascript">print_state("state");</script>                </div>
              <div class="form-group d-flex">
                <label for="Min temperature" style="padding:6% 4% 6% 3%;">Min Temperature</label>
                <input type="text" style="margin:5%; border-radius: 13px; width:45%;" class="form-control border-primary" id="mintemp" name="mintemp" placeholder="10.6"><br>
              </div>
              <div class="form-group d-flex">
                <label for="Mean Temperature" style="padding:6% 1% 6% 2%;">Mean Temperature</label>
                <input type="text" style="margin:5%; border-radius: 13px; width:45%;" class="form-control border-primary" id="meantemp" name="meantemp" placeholder="23.8"><br>
              </div>
              <div class="form-group d-flex">
                <label for="Max Temperature" style="padding:6% 4% 6% 3%;">Max Temperature</label>
                <input type="text" style="margin:5%; border-radius: 13px; width:45%;" class="form-control border-primary" id="maxtemp" name="maxtemp" placeholder="17.2"><br>
              </div>
            </div>
            <div class="col-6">
```

```
                <div class="form-group d-flex">
                    <label for="District" style="padding:6% 6% 6% 6%;">District</label>
                    <select id="city" style="margin:5%; border-radius: 13px; width:45%;" class="form-control border-primary" name="city" required></select>
                    <script language="Javascript">print_city("city");</script>                    </div>

                <div class="form-group d-flex">
                    <label for="pressure"style="padding:6% 7% 6% 3%;">Pressure</label>
                    <input type="text" style="margin:5%; border-radius: 13px;width:45%;" class="form-control border-primary" id="pressure" name="pressure" placeholder="94.2"><br>
                </div>

                <div class="form-group d-flex">
                    <label for="Preceptions" style="padding:6% 3% 6% 0%;;">Preceptions</label>
                    <input type="text" style="margin:5%; border-radius: 13px;width:45%;" class="form-control border-primary" id="preceptions" name="preceptions" placeholder="20.72"><br>
                </div>

                <div class="form-group d-flex">
                    <label for="Wind Speed" style="padding:6% 2% 6% 0%;;">Wind Speed</label>
                    <input type="text" style="margin:5%; border-radius: 13px;width:45%;" class="form-control border-primary" id="windspeed" name="windspeed" placeholder="7.29"><br>
                </div>
            </div>
            <div class="col-12"><br></div>
            <div class="col-4"></div>
            <div class="col-4">
                <button type="submit" class="btn btn-primary" style="margin-left:40%; padding:3% 10% 3% 10%; border-radius: 15px;">Predict</button>
            </div>
        </form>
        </div>
```

```
                </div>
                <div class="col-4">
                    <div class="col-6"><div id="icons"></div></div>

                </div>

                <div class="col-4">
                    <h3>{{error}}</h3>
                </div>
            </div>
        </div>
        <script src="https://cdnjs.cloudflare.com/ajax/libs/moment.js/2.29.1/moment.min.js" integrity="sha512-qTXRIMyZIFb8iQcfjXWCO8+M5Tbc38Qi5WzdPOYZHIlZpzBHG3L3by84BBBOiRGiEb7KKtAOAs5qYdUiZiQNNQ==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>
        <script href="../Static/css/bootstrap.bundle.min.js"></script>

    </body>
</html>
```

SAMPLE CODE FOR CROP RECOMMENDATION PAGE

```
 <!DOCTYPE html>
<html>
    <head>
        <title>Home Page</title>
        <!--link href="../Static/css/bootstrap.min.css" rel="Stylesheet" type="text/css">
        <link href="../Static/css/Style.css" rel="Stylesheet" type="text/css"-->
        <link href="{{ url_for('static', filename='css/bootstrap.min.css') }}"  rel= "stylesheet"text='text/css'/>
        <link href="{{ url_for('static', filename='css/style.css') }}" rel='stylesheet' type='text/css'>
        <!--City-->
        <script type="text/JavaScript" src="{{ url_for('static', filename='script/cities.js') }}"></script>
        <script>
          function myfunc(){
          var el=`<h2> You can Grow</h2><image  style="width:50%; margin:5%; border-radius:10%;"
src="{{crops[0]}}"></image><h4 style="margin-left:10%;">{{crops[1]}}</h4>`;
          document.getElementById("predict").innerHTML=el;
        };
        function error(){
          document.getElementById('err').innerHTML="Can't predicted Some Error. Try Again"
        }
        </script>
    </head>
    <body class="bg-info" onload={{crops[2]}}>
        <div class="container">
          <nav class="navbar navbar-expand-lg navbar-light">
            <div class="collapse navbar-collapse" id="navbarSupportedContent">
              <ul class="navbar-nav mr-auto">
                <li class="nav-item active">
                  <a class="navbar-brand" href="#" style="color:blue ;"><b>EDA Rainfall</b></a>
                </li>
              </ul>
              <ul class="nav navbar-nav ml-auto w-100 justify-content-end">
                <li class="nav-item active" >
                  <a class="nav-link" href="{{ url_for('home') }}">Home</a>
                </li>
                <li class="nav-item active">
                  <a class="nav-link" href="rainfall">Rainfall Prediction</a>
                </li>
                <li class="nav-item active"Style="border-bottom: 4px solid blue;">
                  <a class="nav-link" href="crop">Crop Recommender</a>
                </li>
              </ul>
            </div>
```

```
</nav>
<div class="row">
    <div class="col-8">
        <div class="row">
            <div class="col-6">
                <form  method="POST" action="{{url_for('crop_prediction')}}">
                <div class="form-group d-flex">
                    <label for="State" style="padding:6% 15% 6% 18%;">State</label>
                    <select style="margin:5%; border-radius: 13px; width:45%;" class="form-control border-primary" onchange="print_city('state', this.selectedIndex);" id="state" class="form-control" name="state" required></select>
                    <script language="Javascript">print_state("state");</script>
                </div>
                <div class="form-group d-flex">
                    <label for="nitrogen" style="padding:6% 8% 6% 16%;">Nitrogen</label>
                    <input type="number" style="margin:5%; border-radius: 13px; width:45%;" class="form-control border-primary" id="nitrogen" name="nitrogen" placeholder="Example:40" required><br>
                </div>
                <div class="form-group d-flex">
                    <label for="pottasium" style="padding:6% 5% 6% 15%;">Pottasium</label>
                    <input type="number" style="margin:5%; border-radius: 13px; width:45%;" class="form-control border-primary" id="pottasium" name="pottasium" placeholder="Example:40" required><br>
                </div>
                <div class="form-group d-flex">
                    <label for="rainfall" style="padding:6% 10% 6% 18%;">Rainfall</label>
                    <input type="number" style="margin:5%; border-radius: 13px; width:45%;" class="form-control border-primary" id="rainfall" name="rainfall" placeholder="Example:202.56" step="0.01" required><br>
                </div>
            </div>
            <div class="col-6">
                <div class="form-group d-flex">
                    <label for="city" style="padding:6% 10% 6% 15%;">District</label>
                    <select id="city" style="margin:5%; border-radius: 13px; width:45%;" class="form-control border-primary" name="city" required></select>
                    <script language="Javascript">print_city("city");</script>
                </div>
                <div class="form-group d-flex">
                    <label for="phosphorous"style="padding:6% 8% 6% 4%;">Phosphorous</label>
                    <input type="number" style="margin:5%; border-radius: 13px;width:45%;" class="form-control border-primary" id="phosphorous" name="phosphorous" placeholder="Example:40" required><br>
                </div>

                <div class="form-group d-flex">
                    <label for="ph" style="padding:6% 10% 6% 12%;;">Ph Level</label>
                    <input type="number" style="margin:5%; border-radius: 13px;width:45%;" class="form-control border-primary" id="ph" name="ph" placeholder="Example:5.2" step="0.01"><br>
                </div>
            </div>
            <div class="col-12"><br></div>
            <div class="col-4"></div>
            <div class="col-4">
                <button type="submit" class="btn btn-primary" style="margin-left:40%; padding:3% 10% 3% 10%; border-radius: 15px; ">Predict</button>
            </div>
        </form>
        <div class="form-group" style="justify-content: center;">
            <h1 class="text-center"><b id="err"></b></h1>
       </div>
        </div>
    </div>
    <div class="col-4">
        <p id="predict"></p>
    </div>
```

```
          </div>
        </div>
        <script href="../Static/css/bootstrap.bundle.min.js"></script>

    </body>
</html>
```

GITHUB & PROJECT DEMO LINK

GitHub Link

Demonstration Link

REFERENCE

[1] National disaster management guidelines – Management of landslides and snow avalanches, 2009. A publication of the national disaster management authority, Government of India, June 2009, New Delhi.

[2] D. T. Anh, M. D. Bui and P. Rutschmann, “Long Short Term Memory for monthly rainfall prediction in Camau, Vietnam,” 2018.

[3] S. Khanduri, “Landslides over two places of Uttarakhand; an observation,” International Journal of Engineering Applied Sciences and Technology, vol. 3, no. 2, pp. 52-56, June 2018.

[4] I. Mahmud, S. H. Bari, and M. T. U. Rahman, “Monthly rainfall forecast of Bangladesh using autoregressive integrated moving average method,” Environmental Engineering Research, vol. 0, p. 0, 2016, doi: 10.4491/eer.2016.075.

[5] M. T. Abraham, D. Pothuraju, N. Satyam, “Rainfall Thresholds for Prediction of Landslides in Idukki, India: An Empirical Approach,” Water, 2019. doi: 10.3390/w11102113.

[6] J. Joseph and T. K. Rathees, “Rainfall Prediction using Data Mining Techniques,” International Journal of Computer Applications, vol. 83, pp. 11–15, 2013, doi: 10.5120/14467-2750.

[7] M. A. I. Navid and N. H. Niloy, “Multiple Linear Regressions for Predicting Rainfall for Bangladesh,” Communications, vol. 6, no. 1, Art. no. 1, Feb. 2018, doi: 10.11648/j.com.20180601.11.

[8] A. J. Smola and B. Schölkopf, “A tutorial on Support Vector Regression,” Statistics and Computing, vol. 14, no. 3, pp. 199-222, 2004. doi: 10.1023/B:STCO.0000035301.49549.88

[9] M. Welling, “Support Vector Regression,” Department of Computer Science, University of Toronto, 2004. Available: https://www.ics.uci.edu/~welling/teaching/KernelsICS273B/SVregres sion.pdf

[10] C. Hu, Q. Wu, H. Li, S. Jian, N. Li and Z. Lou, “Deep learning with a Long Short-Term Memory networks approach for Rainfall-Runoff Simulation,” Water, vol. 10, 2018. doi: 10.3390/w10111543

[11] X. H. Le, H. V. Ho, G. Lee and S. Jung, “Application of Long ShortTerm Memory (LSTM) Neural Network for Flood Forecasting,” Water, 2019. doi: 10.3390/w11071387.

[12] M. Swapna, N.Sudhakar, “A hybrid model of rainfall prediction using both parameterized and time series models,” International Journel of Pure and Applied Mathematics, vol. 119, no.14, pp. 1549-1556, 2018.

[13] T. Mohamed and A. Ibrahim, “Time Series Analysis of Nyala Rainfall Using ARIMA Method,” vol. 17, 2016.

[14] D. Eni and F. Adeyeye, “Seasonal ARIMA Modeling and Forecasting of Rainfall in Warri Town, Nigeria,” Journal of Geoscience and Environment Protection, vol. 03, pp. 91–98, 2015, doi: 10.4236/gep.2015.360