1.INTRODUCTION

Agriculture is the backbone of the Indian economy. For data analytics in crop prediction, there are different techniques or algorithms, and with the help of those we can predict crop yield. There are multiple ways to increase the crop yield and the quality of the crops. In India, agricultural yield primarily depends on weather conditions. Rice cultivation mainly depends on rainfall.

In the past farmers used to predict their yield from previous year yield experiences. Mainly we use Random forest classifier algorithm to predict the crop, and Random forest regressor algorithm, Logistic regressor algorithm, Support vector algorithm to predict the crop yield. Data mining is also useful for predicting crop yield production. The main objectives are

a. To use machine learning techniques to predict crop yield.
b. To provide easy to use User Interface.
c. To increase the accuracy of crop yield prediction.
d. To analyse different climatic parameters (cloud cover, rainfall, temperature)

2.SYSTEM ANALYSIS

2.1 EXISTING SYSTEM

For data analytics in crop prediction, there are different techniques or algorithms, and with the help of those algorithms we can predict crop yield.

The accuracy of predictions are 75 percent only in all the crops.

Niketa et al in 2016 have indicated that the yield of the crop depends on the seasonal climate. In India, climate conditions vary unconditionally. In the time of drought, farmers face serious problems. So this taken into consideration they used some machine learning algorithms to help the farmers to suggest the crop for the better yield. They take various data from the previous years to estimate future data. They used SMO classifiers in WEKA to classify the results. The main factors that take into consideration are minimum temperature, maximum t emperature, average temperature, and previous year's crop information and yield information. Using SMO tool they classified the previous data into two classes that are high yield and low yield.

Eswari et al in 2018 have indicated that yield of the crop depends on the perception, average, minimum and maximum temperature. Apart, from that, they have taken one more attribute named crop evapotranspiration. The crop evapotranspiration is a function of both the weather and growth stage of the plant. This attribute is taken into consideration to get a good decision on the yield of the groups. They all collected the dataset with these attributes and send as input to the Bayesian network and classify into the two classes named true and false classes and compared with the observed classifications in the model with a confusion matrix and bring the accuracy. Finally, they concluded that crop yield prediction with Naïve Bayes and Bayesian network give high accuracy when compared to SMO classifier and forecasting the crop yield prediction in different climate and cropping scenarios will be beneficial.

2.2 DISADVANTAGES

- Process is based on image analysis results which are not accurate as in this method soil conditions are not considered.
- Image processing is a time taking process.
- Less accuracy than the proposed algorithm.

2.3 PROPOSED SYSTEM

- In the proposed system, we develop Prediction of the crop yield using the efficient algorithm.
- The challenge in it is to build the efficient model to predict the most efficient model to predict the output of the crop so try with the different algorithms and compare all the algorithms and which one has the less error and loss chose that model and predict the yield of that particular crop.
- We implement the live price prediction system, so the results will be in dynamic and will vary according to the day to day scenarios.
- **Random forest classification:**

  It perform classification to predict the crop and produce a accurate result.

- **Random forest regressor:**

  It perform regression to predict crop yield and produce a accurate profit based on the crop yield.

- **Logistic regressor:**

  It is used to calculate or predict the probability of a binary (yes/no) event occurring.

- **Support vector regressor:**

  Itis a supervised learning algorithm that is used to predict discrete values.

2.4 ADVANTAGES

- It consists of 86%accuracy.

- Crop yield prediction is also used by farmers to make decisions about when to plant and harvest crops based on soil moisture content, pest infestations, and other factors such as weather conditions and fertilizer requirements.

- The prediction tool might be finished with the beneficial resource of the use of facts mining techniques. Previous researches depict the software program software of facts mining techniques within the agricultural area.

3.LITERATURE SURVEY

TITLE:

CROP YIELD PREDICTION USING MACHINE LEARNING

AUTHOR:

Mayank

Champaneri

CONTENT:

In Predicting yield of the crop using machine learning algorithm. International Journal of Engineering Science Research Technology. This paper focuses on predicting the yield of the crop based on the existing data by using Random Forest algorithm. Real data of Tamil Nadu were used for building the models and the models were tested with samples. Random Forest Algorithm can be used for accurate crop yield prediction.

In Random forests for global and regional crop yield prediction, PLoS ONE Journal. Our generated outputs show that RF is an effective and adaptable machine-learning method for crop yield predictions at regional and global scales for its high accuracy and precision, ease of use, and utility in data analysis. Random Forest is the most efficient strategy and it outperforms multiple linear regression (MLR).

In Crop production Ensemble Machine Learning model for prediction. International Journal of Computer Science and Software Engineering (IJCSSE). In this paper, AdaNaive and AdaSVM are the proposed ensemble model used to project the crop production over a time period. Implementation done using AdaSVM and AdaNaive. AdaBoost increases efficiency of SVM and Naive Bayes algorithm.

In Machine learning approach for forecasting crop yield based on parameters of climate. The paper provided in International Conference on Computer Communication and Informatics (ICCCI). In the current research a software tool named Crop Advisor has been developed as a user friendly web page for predicting the influence of climatic parameters on the crop yields.C4.5 algorithm is used to produce the most influencing climatic parameter on the crop yields of selected crops in selected districts of Madhya Pradesh.

In Prediction On Crop Cultivation. International Journal of Advanced Research in Computer Science and Electronics Engineering (IJARCSEE) Volume 5, Issue 10, October 2016. Presently, soil analysis and interpretation of soil test results is paper based. This in one way or another has contributed to poor interpretation of soil test results which has resulted into poor

recommendation of crops, soil amendments and fertilizers to farmers thus leading to poor crop yields, micro-nutrient deficiencies in soil and excessive or less application of fertilizers. Formulae to Match Crops with Soil, Fertilizer Recommendation.

In Analysis of Crop Yield Prediction by making Use Data Mining Methods. IJRET: The paper provided in International Journal of Research in Engineering and Technology. In this paper the main aim is to create a user-friendly interface for farmers, which gives the analysis of rice production based on the available data. For maximizing the crop productivity various Data mining techniques were used to predict the crop yield. Such as K-Means algorithm to forecast the pollution factor in the atmosphere.

In Applications of Machine Learning Techniques in Agricultural Crop Production. Indian Journal of Science and Technology, Vol 9(38), DOI:10.17485/ijst/2016/v9i38/95032, October 2016. From GPS based colour images is provided as an intensified indistinct cluster analysis for classifying plants, soil and residue regions of interest. The paper includes various parameters which can help the crop yield for better enhancement and ratio of the yield can be increased during cultivation.

In this paper, we present a comprehensive review of research dedicated to the application of machine learning in agricultural production systems. Machine learning (ML) has emerged together with big data technologies, techniques, methods and high-performance computing to generate new opportunities to unravel, quantify, and analyse data intensive processes in agricultural operational sectors. By using Support Vector Machines (SVP) the Paper is Implemented.

In a Study to Determine Yield for Crop Insurance using Precision Agriculture on an Aerial Platform. Symbiosis Institute of Geoinformatics Symbiosis International University 5th & 6th Floor, Artur Centre, Gokhale Cross Road, Model Colony, Pune – 411016. Precision agriculture (PA) is the application of geospatial methodologies and remote sensors to identify variations in the field and to deal with them using different strategies. The causes of variability of crop growth in an agricultural field might be due to crop stress, irrigation practices, incidence of pest and disease etc. The Paper is Implemented using Ensemble Learning (EL).

In Random Forests for Global and Regional Crop Yield Predictions. institute on the Environment, University of Minnesota, St. Paul, MN 55108, United States of America. The generated outputs show that RF is an effective and different machine-learning method for crop yield predictions at regional and global scales for its high accuracy. The Paper is Implemented using k-nearest neighbour, Support Vector Regression (SVG).

TITLE:

Rice Crop Yield Forecasting of Tropical Wet and Dry Climatic Zone of India Using Data Mining Techniques

AUTHOR:

Niketa Gandhi

CONTENT:

Data mining is the process of identifying the hidden patterns from large and complex data. It may provide crucial role in decision making for complex agricultural problems. Data visualisation is also equally important to understand the general trends of the effect of various factors influencing the crop yield. The present study examines the application of data visualisation techniques to find correlations between the climatic factors and rice crop yield. The study also applies data mining techniques to extract the knowledge from the historical agriculture data set to predict rice crop yield for Kharif season of Tropical Wet and Dry climatic zone of India. The data set has been visualised in Microsoft Office Excel using scatter plots. The classification algorithms have been executed in the free and open source data mining tool WEKA. The experimental results provided include sensitivity, specificity, accuracy, F1 score, Mathews correlation coefficient, mean absolute error, root mean squared error, relative absolute error and root relative squared error. General trends in the data visualisation show that decrease in precipitation in the selected climatic zone increases the rice crop yield and increase in minimum, average or maximum temperature for the season increases the rice crop yield. For the current data set experimental results show that J48 and LAD Tree achieved the highest accuracy, sensitivity and specificity. Classification performed by LWL classifier displayed the lowest accuracy, sensitivity and specificity results.

TITLE:

Crop Yield Prediction in Tamil Nadu Using Baysian Network

AUTHOR:

K. E. Eswari and L. Vinitha

CONTENT:

Crop yield prediction is an application that helps farmers to improve crop yield. As selection of every crop is very important in agricultural planning, it mainly depends on market price, climate and production rate. The proposed project predicts the crop yield quantity, based on the following factors Temperature, Humidity, Moisture level of soil and area of field. The rate of yield predicted by our proposed project is displayed as an output to the user that aids the farmer to harvest the crop.

TITLE:

Use of Data Mining in Crop Yield Prediction

AUTHOR:

Shruti Mishra, Priyanka Paygude, Snehal Chaudhary and SonaliIdate

CONTENT:

Agriculture is the most important sector that influences the economy of India. It contributes to 18% of India's Gross Domestic Product (GDP) and gives employment to 50% of the population of India. People of India are practicing Agriculture for years but the results are never satisfying due to various factors that affect the crop yield. To fulfill the needs of around 1.2 billion people, it is very important to have a good yield of crops. Due to factors like soil type, precipitation, seed quality, lack of technical facilities etc the crop yield is directly influenced. Hence, new technologies are necessary for satisfying the growing need and farmers must work smartly by opting new technologies rather than going for trivial methods. This paper focuses on implementing crop yield prediction system by using Data Mining techniques by doing analysis on agriculture dataset. Different classifiers are used namely J48, LWL, LAD Tree and IBK for prediction and then the performance of each is compared using WEKA tool. For evaluating performance Accuracy is used as one of the factors. The classifiers are further compared with the values of Root Mean Squared Error (RMSE), Mean Absolute Error (MAE) and Relative Absolute Error (RAE). Lesser the value of error, more accurate the algorithm will work. The result is based on comparison among the classifiers.

TITLE:

Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review" in Computers and Electronics in Agriculture,

AUTHOR:

Anna Chlingaryana, Salah Sukkarieha and Brett Whelanb,

CONTENT:

Agricultural output is the main contributor in the national economy. But the agricultural output completely depends on the climatic and environment conditions like rainfall, temperature, humidity etc. The early prediction of the crop yield can facilitate farmers for better crop management. Here, in this paper the various data sets are simulated using the machine learning technique i.e. Decision Tree algorithm. The algorithm is applied on the datasets and the comparison of the results has been done. This paper shows the influence of rainfall attribute in the crop yield. These findings will supplement to the farmers for timely action towards the crop yield management.

TITLE:

Rice Crop Yield Prediction using Data Mining Techniques: An Overview

AUTHOR:

Dakshayini Patil

CONTENT:

This paper shows the overview of rice crop yield prediction. Examines Different data mining techniques utilized for foreseeing rice crop yield. Rice crop creation assumes an imperative part in sustenance security of India, contributing over 40% to general yield generation. High harvest generation is reliant on appropriate climatic conditions. Inconvenient regular atmosphere conditions, for example low precipitation or temperature extremes can drastically diminish edit yield. Growing better strategies to foresee edit efficiency in various climatic conditions can help rancher and different partners in vital basic leadership as far as agronomy and harvest decision. This paper reports utilization of various information mining methods will anticipate rice trim yield for Maharashtra state, India. To this review, 27 regions of Maharashtra were picked on the establishment of accessible information from openly available Indian Administration records with different atmosphere and harvest parameters. Precipitation, least temperature, normal temperature, most extreme temperature, reference trim evapotranspiration, range, generation and yield for the Kharif season (June to November) were the parameters choosen for the study for the years 1998 to 2002. WEKA tool was used for dataset processing Keywords— WEKA tool; Classifiers; Crop analysis; Data mining; Yield prediction

TITLE:

Applying Data Mining Techniques to Predict Annual Yield of Major Crops and Recommend Planting Different Crops in Different Districts in Bangladesh

AUTHOR:

A. T. M Shakil Ahamed, NavidTanzeem Mahmood, Nazmul Hossain, Mohammad Tanzir Kabir, Kallal Das, Faridur Rahman

CONTENT:

Agricultural crop production depends on various factors such as biology, climate, economy and geography. Several factors have different impacts on agriculture, which can be quantified using appropriate statistical methodologies. Applying such methodologies and techniques on historical yield of crops, it is possible to obtain information or knowledge which can be helpful to farmers and government organizations for making better decisions and policies which lead to increased production. In this paper, our focus is on application of data mining techniques to extract knowledge from the agricultural data to estimate crop yield for major cereal crops in major districts of Bangladesh.

TITLE:

Agricultural Crop Yield Prediction Using Artificial Neural Network Approach

AUTHOR:

Snehal S. Dahikar and Sandeep V. Rode

CONTENT:

By considering various situations of climatologically phenomena affecting local weather conditions in various parts of the world. These weather conditions have a direct effect on crop yield. Various researches have been done exploring the connections between large-scale climatologically phenomena and crop yield. Artificial neural networks have been demonstrated to be powerful tools for modelling and prediction, to increase their effectiveness. Crop prediction methodology is used to predict the suitable crop by sensing various parameter of soil and also parameter related to atmosphere. Parameters like type of soil, PH, nitrogen, phosphate, potassium, organic carbon, calcium, magnesium, sulphur, manganese, copper, iron, depth, temperature, rainfall, humidity. For that purpose we are used artificial neural network (ANN).

4.SYSTEM REQUIREMENTS

4.1 HARDWARE REQUIREMENTS:

- System : Intel/Ryzen Processor.
- Hard Disk : 500 GB.
- Monitor : 15'' LED
- Input Devices : Keyboard, Mouse
- Ram : 4 GB

4.2 SOFTWARE REQUIREMENTS:

- Operating system(OS)-Windows 7/10/11
- Frontend- Tkinter
- Backend-Python version of 3.7.0
- Coding Language : Python
- Database : MYSQL

5.SYSTEM ARCHITECTURE

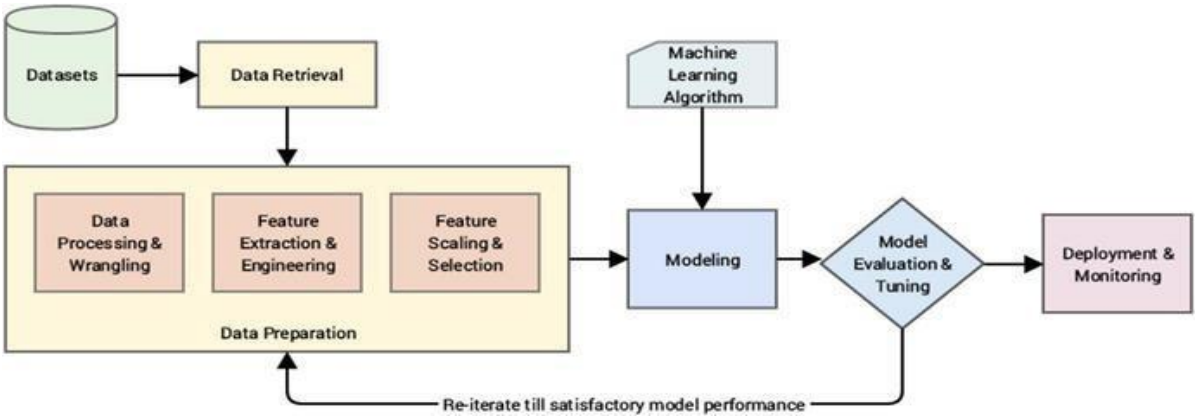


Figure 5 System Architecture for Crop Yield Prediction

6.UML DIAGRAMS

6.1 USECASE DIAGRAM:

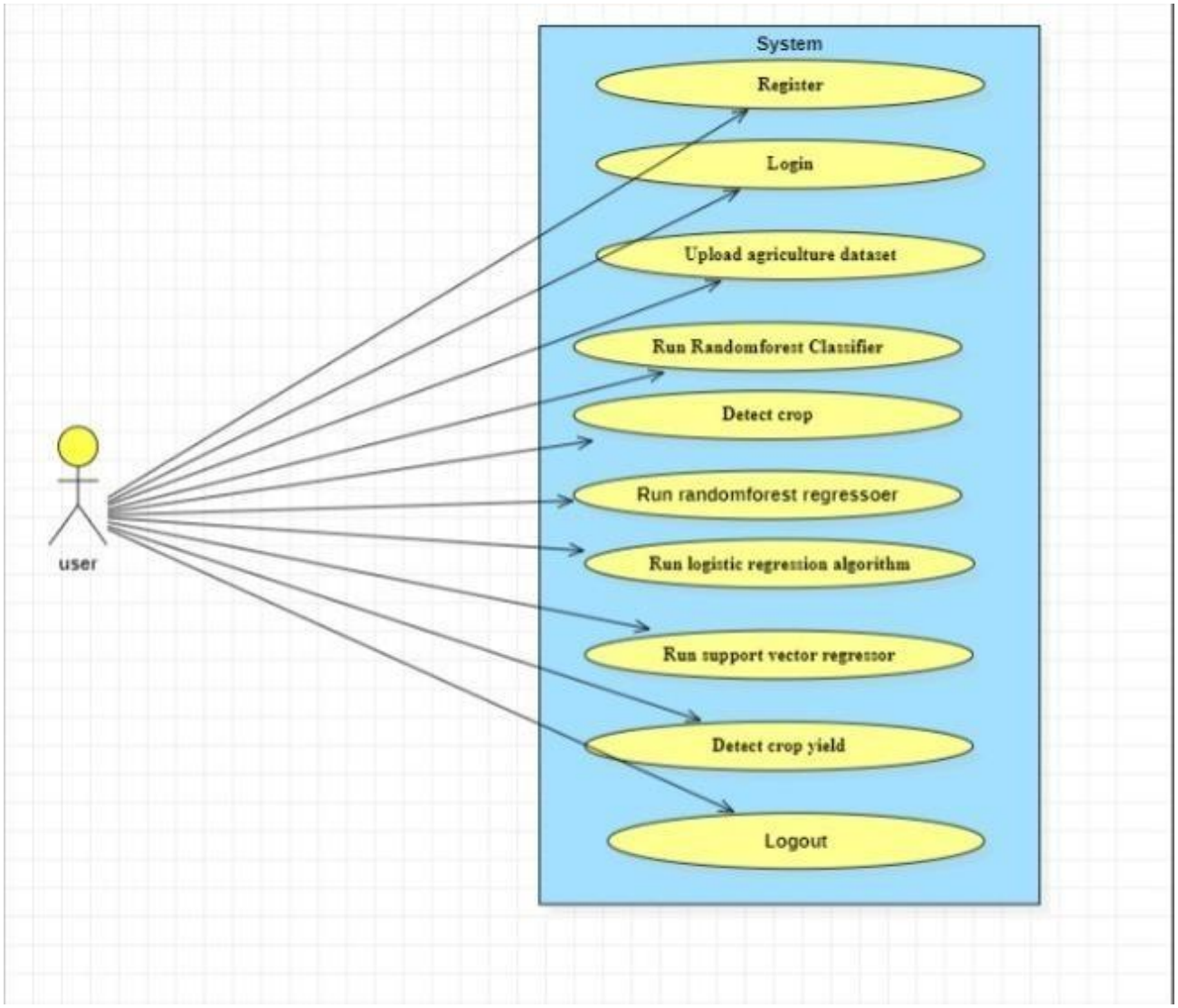


Figure 6.1 Use Case Diagram for Crop Yield Prediction

6.2 CLASS DIAGRAM:

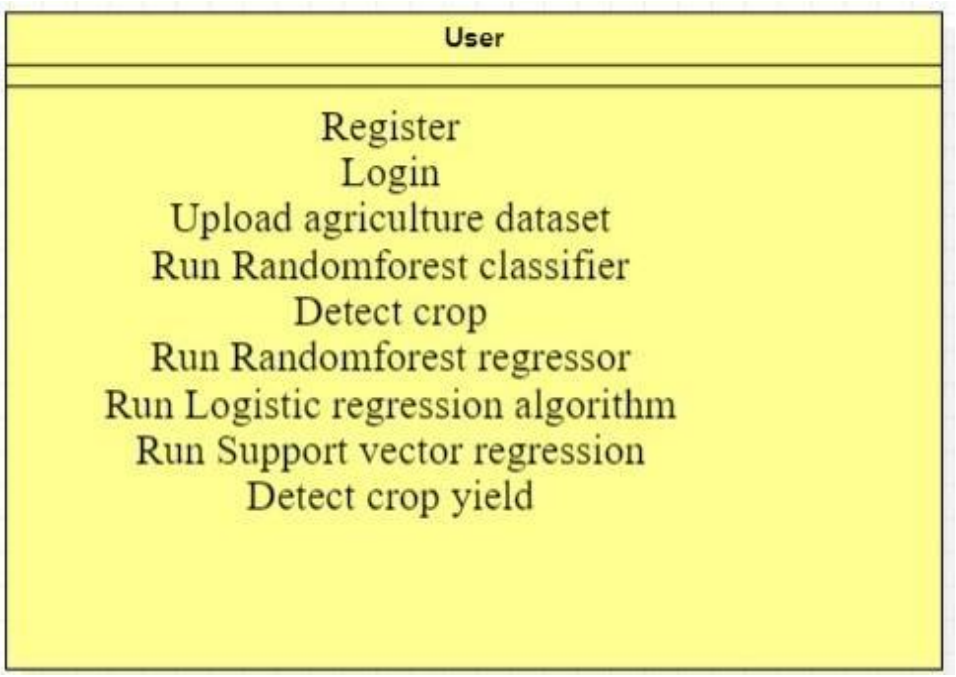


Figure 6.2 Class Diagram for Crop Yield Prediction

6.3 SEQUENCE DIAGRAM:

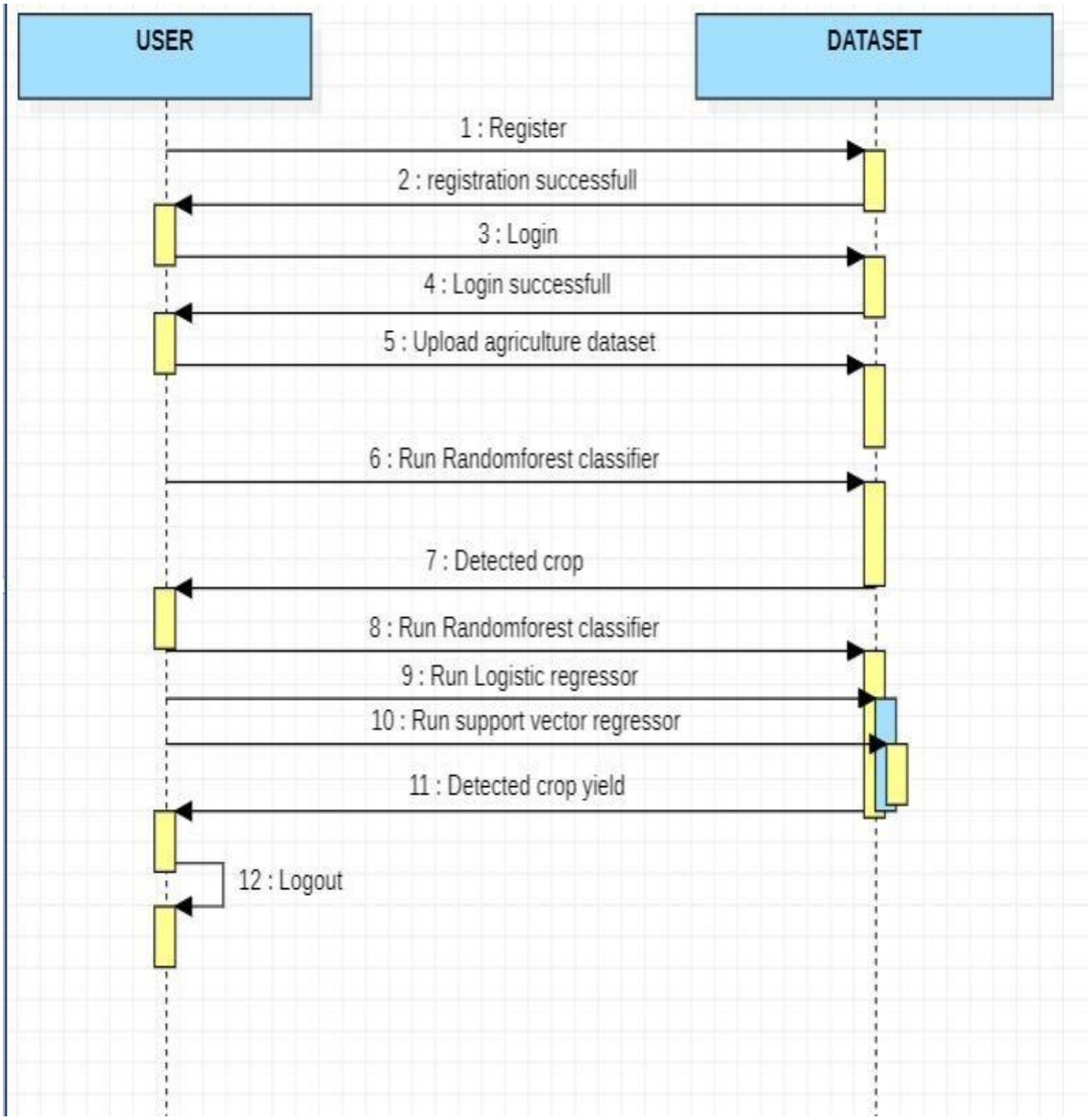


Figure 6.3 Sequence Diagram for Crop Yield Prediction

6.4 ACTIVITY DIAGRAM:

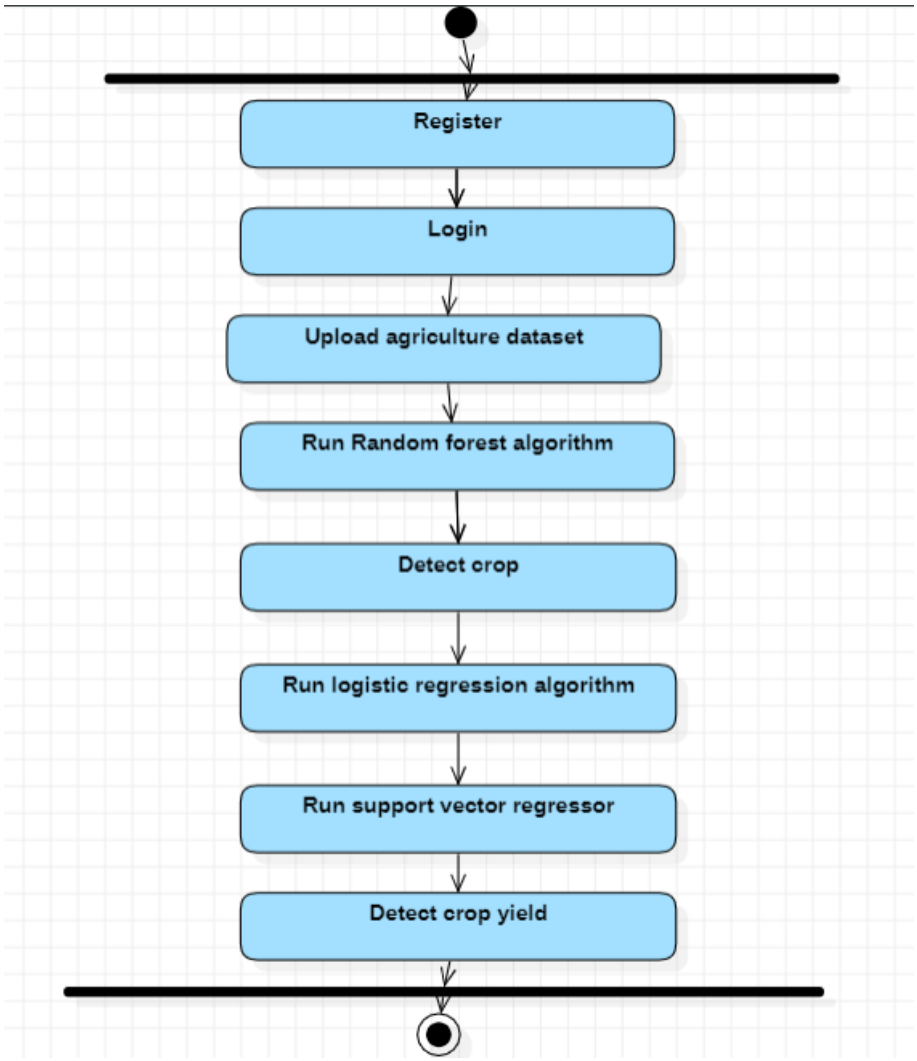


Figure 6.4 Activity Diagram for Crop Yield Prediction

7.MODULES

- **User:**

  Takes dataset and predict the crop yield. User has to register than only he can login to the system. He will train different machine learning algorithms with the crop yield dataset. He will get prediction results of each algorithm. Than he will calculate accuracy for each algorithm and show the accuracy results. He can test individual record prediction with the random forest regressor algorithm and he will get results of crop yield.

- **Upload agriculture dataset:**

  The dataset relating to crop yield is essentially nonlinear.

- **Pre-process Dataset:**

  It is a technique that is used to convert the raw data set into a clean data set.

- **Run Random forest algorithm-**Our generated outputs show that RF is an effective and adaptable machine-learning method for crop yield predictions at regional and global scales for its high accuracy and precision, ease of use, and utility in data analysis.

- **Detect crop-**It performs classification and try to predict the status of the crop as damaged or not damaged.

- **Run random forest regressor-**It is used to solve a variety of business problems where the crop needs to predict a continuous value.

8.DATABASE DESIGN

```
create table user
( username varchar(30) NOT NULL,
password varchar(32) NOT NULL,
email varchar(32),
phone varchar(32),
address varchar(32)
);
```

9.IMPLEMENTATION:

```
# Libraries
from django.shortcuts import render
from django.http import HttpResponse
import pandas as pd
import emoji
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
import itertools
from sklearn.naive_bayes import MultinomialNB
from sklearn import metrics
from sklearn.linear_model import PassiveAggressiveClassifier
import os
import seaborn as sns
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix
# Input data files are available in the "../input/" directory.
```

```
# For example, running this (by clicking run or pressing Shift+Enter) will list the files in the input directory

import pymysql

def reg(request):

    if request.method=='POST':

        if request.POST.get('username') and request.POST.get('password') and request.POST.get('email') and request.POST.get('phone') and request.POST.get('address'):

            db_connection = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'crop',charset='utf8')

            db_cursor = db_connection.cursor()

            student_sql_query="INSERT INTO user (username, password,email, phone ,address) VALUES('" +request.POST.get('username')+"' ,'" + request.POST.get('password')+"' , '"+request.POST.get('email')+"','"+ request.POST.get('phone')+"','"+request.POST.get('address')+"')"


            db_cursor.execute(student_sql_query)

            db_connection.commit()

            print('hellllllllllllllllllllllllllllllllllllllllllllllllloo')

            return render(request,'loginpage.html')

        return render(request,'loginpage.html')

def loginuser(request):

    if request.method=='POST':

        if request.POST.get('username') and request.POST.get('password'):

            con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'crop',charset='utf8')

            with con:
```

```
                cur = con.cursor()
                cur.execute("select * FROM user")
                rows = cur.fetchall()
                for row in rows:
                    if row[0] == request.POST.get('username') and row[1]
                    == request.POST.get('password'):
                        status = 'success'
                        #status_data = row[5]
                        return render(request,'index.html')
                return render(request,'index.html')
################ Home #################
def home(request):
    return render(request,'cropyield.html')
######## SVM ######
def nvb(request):
    data = pd.read_csv('D:/crop yield plant disease/crop.csv')
    from sklearn import preprocessing
    labelencoder_X = preprocessing.LabelEncoder()
    X = data.iloc[:, 1:8].values
    y = data.iloc[:, 9].values
    X.shape
    y.shape

    X_train,X_test,y_train,y_test = train_test_split(X,y,test_size =0.2)
```

```
        A_test=[[1997,598400,24.243,42.3484,84,217000,1]]
        #testing
        from sklearn import linear_model
        reg = linear_model.LinearRegression()
        reg.fit(X_train,y_train)
        pred = reg.predict(X_test)
        pred1 = reg.predict(A_test)
        print(pred1)
        score = reg.score(X_train,y_train)
        print("R-squared:", score)
        d = {'a': score}
        #print(reg.score(X_test,y_test))
        #acclogistic=reg.score(X_test,y_test)
        return render(request,'NaiveBayes.html',d)
def rf(request):
        data = pd.read_csv('D:/crop yield plant disease/crop.csv')
        X = data.iloc[:, 1:8].values
        y = data.iloc[:, 9].values
        X.shape
         y.shape

        X_train,X_test,y_train,y_test = train_test_split(X,y,test_size =0.2)

        A_test=[[1997,598400,24.243,42.3484,84,217000,1]]
```

```
        #testing
        from sklearn.datasets import make_regression
        from sklearn.ensemble import RandomForestClassifier
        regr = RandomForestClassifier()
        regr.fit(X_train,y_train)
        pred = regr.predict(X_test)
        pred1 = regr.predict(A_test)
        print(pred1)
        score = metrics.accuracy_score(y_test, pred)
        print("accuracy:  %0.3f" % score)
        d = {'a': score}
        #print(reg.score(X_test,y_test))
        #acclogistic=reg.score(X_test,y_test)
        return render(request,'NaiveBayes.html',d)

def svr(request):
        data = pd.read_csv('D:/crop yield plant disease/crop.csv')
        X = data.iloc[:, 1:8].values
        y = data.iloc[:, 9].values

        #testing
        X_train,X_test,y_train,y_test = train_test_split(X,y,test_size =0.2)
        A_test=[[1997,598400,24.243,42.3484,84,217000,1]]
```

```
        from sklearn.svm import SVR
        regressor = SVR(kernel = 'rbf')

        regressor.fit(X_train,y_train)
        pred = regressor.predict(X_test)
        pred1 = regressor.predict(A_test)
        print(pred1)
        score = regressor.score(X_train,y_train)
        print("R-squared:", score)
        '''score = metrics.accuracy_score(y_test, pred)
        print("accuracy:  %0.3f" % score)'''
        d = {'a': score}
        #print(reg.score(X_test,y_test))
        #acclogistic=reg.score(X_test,y_test)
        return render(request,'NaiveBayes.html',d)

def pac(request):
        return render(request,'NaiveBayes.html')
def svm(request):
        return render(request,'NaiveBayes.html')

def accuracy(request):
        return render(request,'index.html')
```

```
def test(request):
    if request.method=='POST':
        headline1= request.POST.get('headline1')
        headline2= request.POST.get('headline2')
        headline3= request.POST.get('headline3')
        headline4= request.POST.get('headline4')
        headline5= request.POST.get('headline5')
        headline6= request.POST.get('headline6')

        from sklearn import preprocessing
        labelencoder_X = preprocessing.LabelEncoder()
        headline6 = labelencoder_X.fit_transform([[headline6]])

        headline7= request.POST.get('headline7')

        print(headline1)



        headline1= int(headline1)
        headline2 = int(headline2)
        headline3 = float(headline3)
        headline4 = float(headline4)
        headline5 = int(headline5)
        headline6 = int(headline6)
```

```
headline7  = int(headline7)


data = pd.read_csv('D:/crop  yield  plant  disease/crop.csv')


X = data.iloc[:,  1:8].values

y = data.iloc[:,  9].values

X.shape

y.shape


X_train,X_test,y_train,y_test  = train_test_split(X,y,test_size  =0.2)


A_test=[[headline1,headline2,headline3,headline4,headline5,headline6,headline7]]

#testing

from  sklearn.datasets  import  make_regression

from  sklearn.ensemble  import  RandomForestClassifier

reg  = RandomForestClassifier()

reg.fit(X_train,y_train)

pred = reg.predict(X_test)

pred1 = reg.predict(A_test)

print(pred1)

print('---- ---------- ---- ----------')

print(pred)


fakefalse=''

if pred1==0:
```

```
                fakefalse='less  crop yield'
            else:
                fakefalse='high  crop yield'
            score = metrics.accuracy_score(y_test,  pred)
            print("accuracy:  %0.3f" % score)
            d = {'a':pred1,'crop':request.POST.get('headline6')}
            print('hellllllllllllllllllllllllllllllo')
            return  render(request,'NaiveBayes.html',d)
        return render(request,'fres.html',d)
def simple_uoload(request):
        return  render(request,'indexfile1.html')
def sample(request):
        return  render(request,'indexfile1.html')
def fileshow1(request):
        return  render(request,'indexfile2.html')
def loginpage(request):
        return  render(request,'loginpage.html')
def register(request):
        return  render(request,'register.html')
def input(request):
        return  render(request,'input.html')
```

Performance metrics:

Linear regression got 83% accuracy.

10.INPUT DESIGN

INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

OBJECTIVES

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow.

11.OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by system.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.

12.SYSTEM STUDY

12.1 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are,

- **ECONOMICAL FEASIBILITY**
- **TECHNICAL FEASIBILITY**
- **SOCIAL FEASIBILITY**

12.1.1 ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

12.1.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

12.1.3 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

13.PYTHON

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++or Java. It provides constructs that enable clear programming on both small and large scales. Python interpreters are available for many operating systems. CPython, the reference implementation of Python is open source software and has a community-based development model, as do nearly all of its variant implementations. CPython is managed by the non-profit Python Software Foundation. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

13.1 Interactive Mode Programming

Invoking the interpreter without passing a script file as a parameter brings up the following prompt −

$ python

Python 2.4.3 (#1, Nov 11 2010, 13:34:43)

[GCC 4.1.2 20080704 (Red Hat 4.1.2-48)] on linux2

Type "help", "copyright", "credits" or "license" for more information.

>>>

Type the following text at the Python prompt and press the Enter −

>>> print "Hello, Python!"

If you are running new version of Python, then you would need to use print statement with parenthesis as in print ("Hello, Python!");.However in Python version 2.4.3, this produces the following result −

Hello, Python!

13.2 Script Mode Programming

Invoking the interpreter with a script parameter begins execution of the script and continues until the script is finished. When the script is finished, the interpreter is no longer active.

Let us write a simple Python program in a script. Python files have extension (.py). Type the following source code in a test.py file −

Live Demo

```
print "Hello, Python!"
```

We assume that you have Python interpreter set in PATH variable. Now, try to run this program as follows −

```
$ python test.py
```

This produces the following result –

```
Hello, Python!
```

Let us try another way to execute a Python script. Here is the modified test.py file −

Live Demo

```
#!/usr/bin/python
```

14.TESTING

14.1 INTRODUCTION TO TESTING

Testing is a procedure, which uncovers blunders in the program. Programming testing is a basic component of programming quality affirmation and speaks to a definitive audit of determination, outline and coding. The expanding perceivability of programming as a framework component and chaperon costs related with a product disappointment are propelling variables for we arranged, through testing. Testing is the way toward executing a program with the plan of finding a mistake. The plan of tests for programming and other built items can be as trying as the underlying outline of the item itself It is the significant quality measure utilized amid programming improvement. Amid testing, the program is executed with an arrangement of experiments and the yield of the program for the experiments is assessed to decide whether the program is executing as it is relied upon to perform.

14.2 TESTING STRATEGIES

A technique for programming testing coordinates the outline of programming experiments into an all around arranged arrangement of steps that outcome in fruitful improvement of the product. The procedure gives a guide that portrays the means to be taken, when, and how much exertion, time, and assets will be required. The procedure joins test arranging, experiment configuration, test execution, and test outcome gathering and assessment. The procedure gives direction to the specialist and an arrangement of points of reference for the chief. Due to time weights, advance must be quantifiable and issues must surface as ahead of schedule as would be prudent

Keeping in mind the end goal to ensure that the framework does not have blunders, the distinctive levels of testing techniques that are connected at varying periods of programming improvement are:

4.2.1Unit Testing

Unit Testing is done on singular modules as they are finished and turned out to be executable. It is restricted just to the planner's prerequisites. It centers testing around the capacity or programming module. It Concentrates on the interior preparing rationale and information structures. It is rearranged when a module is composed with high union

- Reduces the quantity of experiments
- Allows mistakes to be all the more effectively anticipated and revealed

14.2.2 Black Box Testing

It is otherwise called Functional testing. A product testing strategy whereby the inward workings of the thing being tried are not known by the analyzer. For instance, in a discovery test on a product outline the analyzer just knows the information sources and what the normal results ought to be and not how the program touches base at those yields. The analyzer does not ever inspect the programming code and does not require any further learning of the program other than its determinations. In this system some experiments are produced as information conditions that completely execute every single practical prerequisite for the program. This testing has been utilizations to discover mistakes in the accompanying classifications:

- Incorrect or missing capacities
- Interface blunders
- Errors in information structure or outside database get to
- Performance blunders
- Initialization and end blunders.

In this testing just the yield is checked for rightness.

14.2.3 White Box testing

It is otherwise called Glass box, Structural, Clear box and Open box testing . A product testing procedure whereby express learning of the inner workings of the thing being tried are utilized to choose the test information. Not at all like discovery testing, white box testing utilizes particular learning of programming code to inspect yields. The test is precise just if the analyzer comprehends what the program should do. He or she would then be able to check whether the program veers from its expected objective. White box testing does not represent blunders caused by oversight, and all obvious code should likewise be discernable. For an entire programming examination, both white box and discovery tests are required. In this the

experiments are produced on the rationale of every module by drawing stream diagrams of that module and sensible choices are tried on every one of the cases. It has been utilizations to produce the experiments in the accompanying cases:

- Guarantee that every single free way have been Executed.
- Execute every single intelligent choice on their actual and false Sides.

14.2.4 Integration Testing

Coordination testing guarantees that product and subsystems cooperate an entirety. It tests the interface of the considerable number of modules to ensure that the modules carry on legitimately when coordinated together. It is characterized as a deliberate procedure for developing the product engineering. In the meantime reconciliation is happening, lead tests to reveal blunders related with interfaces. Its Objective is to take unit tried modules and assemble a program structure in view of the recommended outline

Two Approaches of Integration Testing

- Non-incremental Integration Testing
- Incremental Integration Testing

14.2.5 System Testing

Framework testing includes in-house testing of the whole framework before conveyance to the client. Its point is to fulfill the client the framework meets all necessities of the customer's determinations. This testing assesses working of framework from client perspective, with the assistance of particular report. It doesn't require any inward learning of framework like plan or structure of code.

It contains utilitarian and non-useful zones of utilization/item. Framework Testing is known as a super arrangement of a wide range of testing as all the significant sorts of testing are shrouded in it. In spite of the fact that attention on sorts of testing may differ on the premise of item, association procedures, course of events and necessities. Framework Testing is the start of genuine testing where you test an item all in all and not a module/highlight.

14.2.6 Acceptance Testing

Acknowledgment testing, a testing method performed to decide if the product framework has met the prerequisite particulars. The principle motivation behind this test is to assess the framework's consistence with the business necessities and check in the event that it is has met the required criteria for conveyance to end clients. It is a pre-conveyance testing in which whole framework is tried at customer's site on genuine information to discover blunders. The acknowledgment test bodies of evidence are executed against the test information or utilizing an acknowledgment test content and afterward the outcomes are contrasted and the normal ones.

The acknowledgment test exercises are completed in stages. Right off the bat, the essential tests are executed, and if the test outcomes are palatable then the execution of more intricate situations are done.

14.3 TEST APPROACH

A Test approach is the test system usage of a venture, characterizes how testing would be done. The decision of test methodologies or test technique is a standout amongst the most intense factor in the achievement of the test exertion and the precision of the test designs and gauges.

Testing should be possible in two ways

- Bottom up approach
- Top down approach

14.3.1 Bottom up Approach

Testing can be performed beginning from littlest and most reduced level modules and continuing each one in turn. In this approach testing is directed from sub module to primary module, if the fundamental module is not built up a transitory program called DRIVERS is utilized to recreate the principle module. At the point when base level modules are tried consideration swings to those on the following level that utilization the lower level ones they are tried exclusively and afterward connected with the already inspected bring down level modules.

14.3.2 Top down Approach

In this approach testing is directed from fundamental module to sub module. in the event that the sub module is not built up an impermanent program called STUB is utilized for mimic the sub module. This sort of testing begins from upper level modules. Since the nitty gritty exercises more often than not performed in the lower level schedules are not given stubs are composed. A stub is a module shell called by upper level module and that when achieved legitimately will restore a message to the calling module demonstrating that appropriate association happened.

14.4 VALIDATION

The way toward assessing programming amid the improvement procedure or toward the finish of the advancement procedure to decide if it fulfills determined business prerequisites. Approval Testing guarantees that the item really addresses the customer's issues. It can likewise be characterized as to exhibit that the item satisfies its proposed utilize when sent on proper condition.

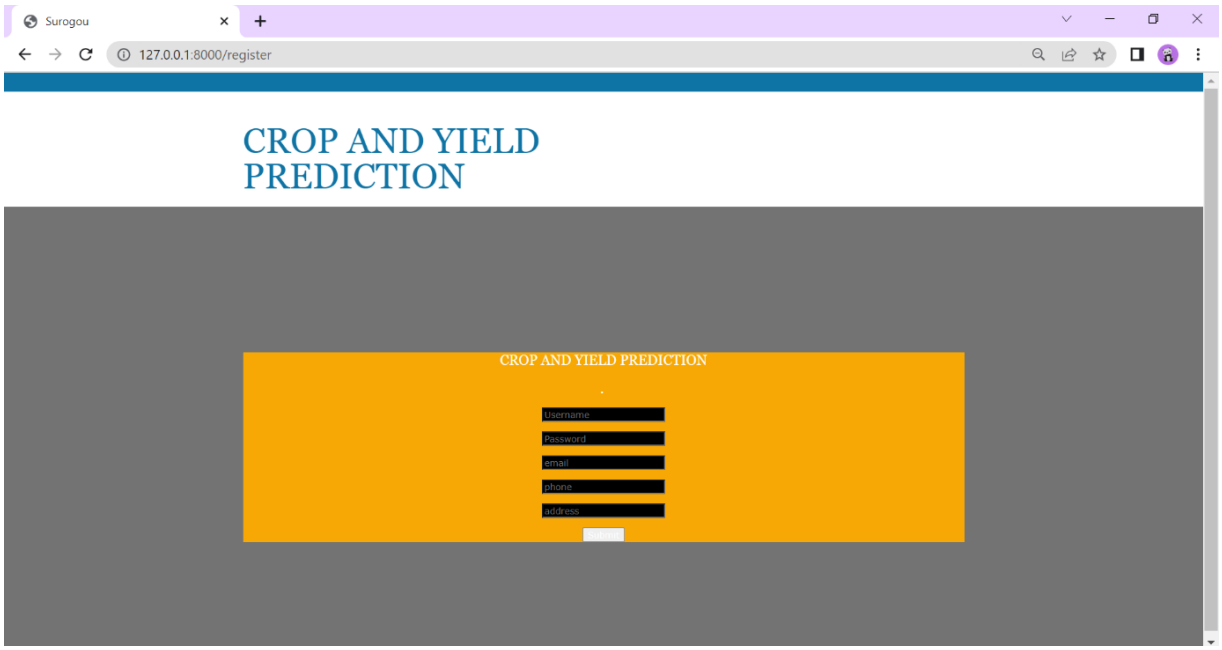
The framework has been tried and actualized effectively and along these lines guaranteed that every one of the prerequisites as recorded in the product necessities determination are totally satisfied.

14.5 Test Cases

Experiments include an arrangement of steps, conditions and sources of info that can be utilized while performing testing undertakings. The principle expectation of this action is to guarantee whether a product passes or bombs as far as usefulness and different perspectives. The way toward creating experiments can likewise help discover issues in the prerequisites or plan of an application. Experiment goes about as the beginning stage for the test execution, and in the wake of applying an arrangement of information esteems, the application has a conclusive result and leaves the framework at some end point or otherwise called execution post condition.

15.RESULTS

REGISTER PAGE

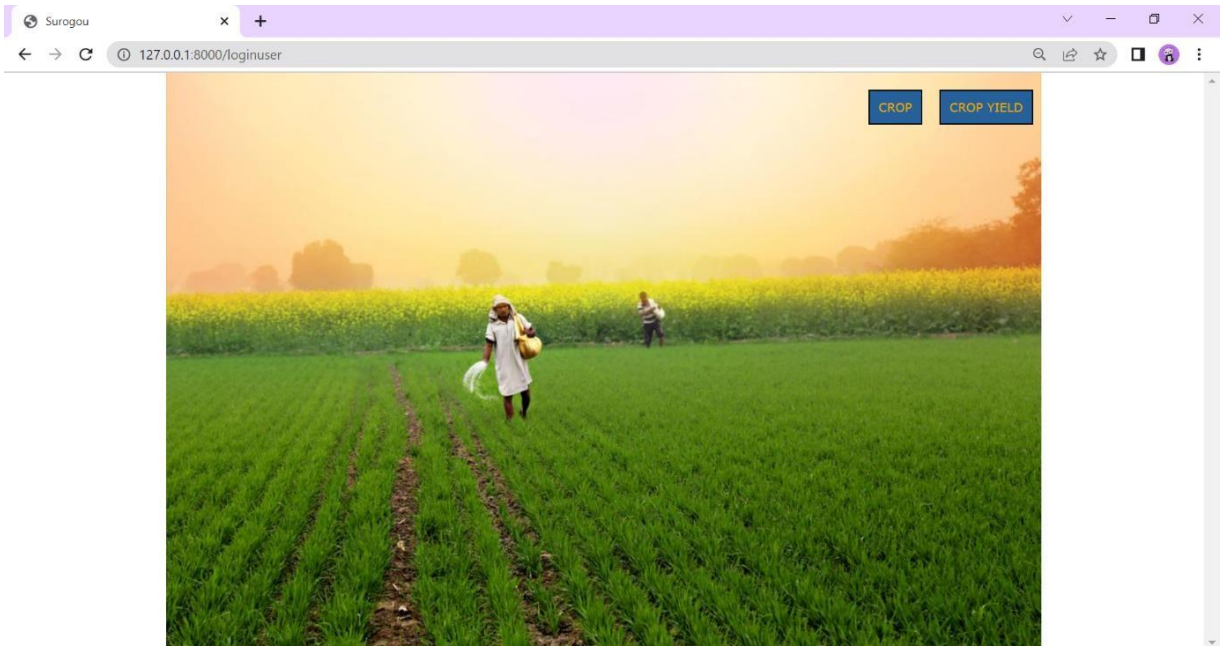


Screenshot 15.1 User Registration Page

LOGIN PAGE

Screenshot 15.2 Login Page

HOME PAGE

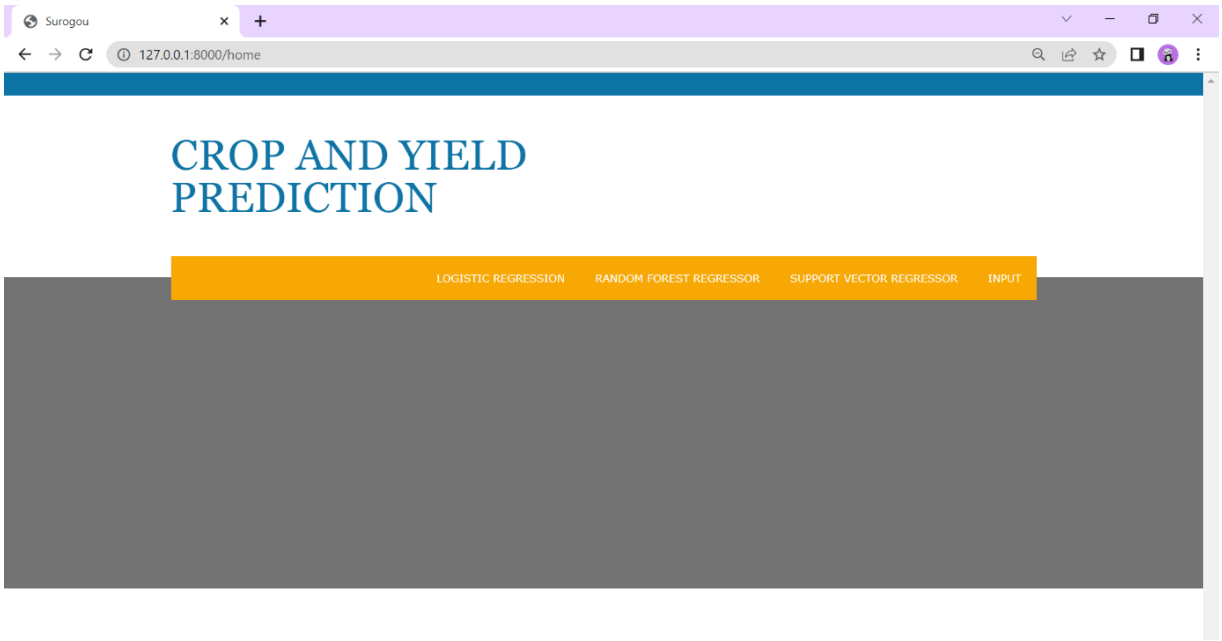


Screenshot 15.3 Home Page

CROP DETAILS PAGE

Screenshot 15.4 Entering the Crop Details

CROP YIELD PAGE



Screenshot 15.5 Crop Yield Page

CROP YIELD DETAILS PAGE

Screenshot 15.6 Crop Yield Details Page

CROP YIELD PREDICTION RESULT

Screenshot 15.7 Crop Yield Prediction

16.CONCLUSION

It produce accuracy of 86% in all crops. Based on the climatic input parameters the present study provided the demonstration of the potential use of techniques in predicting the crop yield based. The developed webpage is user friendly and districts selected in the study indicating higher accuracy of prediction. By providing climatic data of places the user-friendly web page developed for predicting crop and crop yield can be used by any user their choice of crop. Accurate crop yield prediction across different districts helps the farmers in getting the better profit in India. Random Forest algorithm achieves a largest number of crop yield models with a lowest models. It is suitable for massive crop yield prediction in agricultural planning. This makes the farmers to take the right decision for right crop such that the agricultural sector will be developed by innovative ideas.

17.REFERENCE

Niketa Gandhi et al., "Rice Crop Yield Forecasting of Tropical Wet and Dry Climatic Zone of India Using Data Mining Techniques", *IEEE International Conference on Advances in Computer Applications (ICACA)*, 2016.

2.K. E. Eswari and L. Vinitha, "Crop Yield Prediction in Tamil Nadu Using Baysian Network", *International Journal of Intellectual Advancements and Research in Engineering Computations*, vol. 6, no. 2, ISSN 2348-2079.

3.Shruti Mishra, Priyanka Paygude, Snehal Chaudhary and SonaliIdate, "Use of Data Mining in Crop Yield Prediction", *IEEE Xplore Compliant - Part Number:CFP18J06-ART*, ISBN 978-1-5386-0807-4.

4.Anna Chlingaryana, Salah Sukkarieha and Brett Whelanb, "Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review" in Computers and Electronics in Agriculture, Elisver, vol. 151, no. 2018, pp. 61-69, 2018.

5.Dakshayini Patil et al., "Rice Crop Yield Prediction using Data Mining Techniques:An Overview", *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 7, no. 5, May 2017.

6.A. T. M Shakil Ahamed, NavidTanzeem Mahmood, Nazmul Hossain, Mohammad Tanzir Kabir, Kallal Das, Faridur Rahman, et al., "Applying Data Mining Techniques to Predict Annual Yield of Major Crops and Recommend Planting Different Crops in Different Districts in Bangladesh", pp. 1-3, June 2015.

7.K. Menaka and N. Yuvaraj, *Indian Journal of Innovations and Developments*, vol. 5, no. 12, October 2016, ISSN 2277-5390.

8.*Machine Learning by Tom M. Mitchell*, ISBN 0070428077.

9.Snehal S. Dahikar and Sandeep V. Rode, "Agricultural Crop Yield Prediction Using Artificial Neural Network Approach", *International journal of innovative and research in electrical instrumentation and control engineering*, vol. 2, no. 2, 2014.

10.D Ramesh and B Vishnu Vardhan, "Analysis of crop yield prediction using data mining techniques", *International Journal of research in engineering and technology*, ISSN 2319-1163.

11. Ramesh A. Medar, "A survey on data mining techniques for crop yield prediction", *International Journal of advance in computer science and management studies*, vol. 2, no. 9, 2014, ISSN 2231-7782.

12. M C Pooja et al., "Implementation of Crop Yield Forecasting Using Data Mining", *International Research Journal of Engineering and Technology (IRJET)*, 2018, ISSN 2395-0072.

13. Jig Han Jeong et al., "Random Forests for Global and Regional Crop Yield Predictions", *PLOS-ONE*, June 2016.

14. Jharna Majumdar, Sneha Naraseeyappa and Shilpa Ankalaki, "Analysis of agriculture data using data mining techniques: application of big data", *Springer journal*, 2017.

15. S. kanagaSubba Raju et al., "Demand based crop recommender system for farmers International", *Conference on Technological Innovations in ICT For Agriculture and Rural Development*, 2017.

16. Shriya Sahu et al., "An Efficient Analysis of Crop Yield Prediction Using Hadoop framework based on random Forest approach", *International Conference on Computing Communication and Automation*, 2017.

17. E Manjula and S. Djodkitachoumy, *International journal of computation Intelligence and informatics*, vol. 6, no. 4, March 2017.