

CROP YIELD PREDICTION BASED ON INDIAN AGRICULTURE USING STACKING ALGORITHM

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering Degree in Computer Science and Engineering

By

DASARI SRIKANTH
(Reg. No. 38110120)

GANDU PAVAN
(Reg. No. 38110152)



**DEPARTMENT OF COMPUTER SCIENCE AND
ENGINEERING SCHOOL OF COMPUTING
SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY**

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119, TAMILNADU**

MAY-2022



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)



Accredited with Grade “A” by NAAC

(Established under Section 3 of UGC Act, 1956)

JEPPIAAR NAGAR, RAJIV GANDHI SALAI
CHENNAI– 600119
www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **DASARI SRIKANTH (Reg. No. 38110120), GANDU PAVAN (Reg. No. 38110152)** who carried out the project entitled “**CROP YIELD PREDICTION BASED ON INDIAN AGRICULTURE USING STACKING ALGORITHM**” under my supervision from Dec 2021 to March 2022.

Internal Guide

Mr. S. DHAMODARAN., PH.D.,

Head of the Department

Dr. S. VIGNESHWARI, M.E., PH.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

WE, **DASARI SRIKANTH (Reg. No. 38110120), GANDU PAVAN (Reg. No. 38110152)**, hereby declare that the professional Training Report **CROP YIELD PREDICTION BASED ON INDIAN AGRICULTURE USING STACKING ALGORITHM** done under the guidance of **Mr.S.DHAMODARAN., PH.D.**, at Sathyabama Institute of Science and Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering Degree in Computer Science and Engineering.

DATE: 01-03-2022

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D, Dean, School of Computing, Dr.S.Vigneshwari M.E., Ph.D, and Dr.L. Lakshmanan M.E., Ph.D., Head of the Department of Computer Science and Engineering** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Mr.S.DHAMODARAN.,Ph.D.**, for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of COMPUTER SCIENCE AND ENGINEERING** who were helpful in many ways for the completion of the project.

ABSTRACT

FAST Application is Facial Attendance System Technology Application which is an automated attendance system. This system is a prototype model for AI based Attendance System. This system provides a solution using IoT, Microsoft Cognitive Services and JavaScript Frameworks like Angular, Nodejs, Johnny-Five.

Automation is a necessity in the current times as it makes processes more economical and affordable in the long run. It also frees humans from performing banal tasks day in and day out. Once a process is automated the only check that is to be performed is whether it is turned on or not. Automated processes are not prone to errors and even if an error is identified rectification is easy and can be applied system wide without any delay.

Taking attendance is a mundane but necessary task in many academic institutions. In this paper, an alternate and more efficient method of taking attendance is proposed that uses facial recognition and cloud based IoT technology to automate the entire system

TABLE OF CONTENTS

CHAPTER No.	TITLE	PAGE No
	ABSTRACT	6
	LIST OF FIGURES	9
	LIST OF TABLES	10
	LIST OF ABBREVIATIONS	11
1	INTRODUCTION	
	1.1 GENERAL	2
	1.2 OBJECTIVE	3
2	LITERATURE SURVERY	4-9
3	METHODOLOGIES	10
	3.1 EXISTING SYSTEM	10-11
	3.2 PROPOSED SYSTEM	11
	3.3 SYSTEM REQUIREMENT	12
	3.4 HARDWARE	12
	3.5 SOFTWARE REQUIREMENT	13
4	SOFTWARE ENVIROMENT	13
	4.1 PYTHON	13
	4.2 HISTORY OF PYTHON	13
	4.3 PYTHON FEATURES	14
	4.4 GETTING PYTHON	15 -
	SCRIPT MODE PROGRAMMING	17
	4.6 FLASK FRAMEWORK	17-23
	4.7 PYTHON INSTALL	24-26
5	MODULES	26
	5.1 ADMIN LOGIN	27
	5.2 METADATA	27
	5.3 DATA PRE-PROCESSING	27
	5.4 CROP PREDICTION MODULES	28-

6	RESULTS AND DISCUSSION, PERFORMANCE	47
7	SUMMARY AND CONCLUSIONS	68
8	APPENDICES	54-67
9	REFERENCES	69

LIST OF FIGURES

FIGURE NO.	FIGURE NAME	PAGE NO.
3.2	PROPOSED SYSTEM	12
5.7	DATA FLOW DIAGRAM	32
5.1	USECASE DIAGRAM	35
5.2	SEQUENCE DIAAGRAM	36
6.1	HOME SCREEN	47
6.2	UPLOAD DATASET	48
6.3	ENTER DETAILS	50
6.4	PREDICTION	51
6.5	PERFORMANCE ANALYSIS	52
6.6	CROP RECOMMENDATION	53

ABSTRACT

Agriculture and its allied sectors are undoubtedly the largest providers of livelihoods in rural India. The agriculture sector is also a significant contributor factor to the country's Gross Domestic Product (GDP). Blessing to the country is the overwhelming size of the agricultural sector. However, regrettable is the yield per hectare of crops in comparison to international standards. This is one of the possible causes for a higher suicide rate among marginal farmers in India. This paper proposes a viable and user-friendly yield prediction system for the farmers. The proposed system provides connectivity to farmers via a mobile application. The user provides the area & soil type as input. Machine learning algorithms allow choosing the most profitable crop list or predicting the crop yield for a user-selected crop. To predict the crop yield, selected Machine Learning algorithms such as Support Vector Machine (SVM), and K-Nearest Neighbour (KNN) are used. Among them, the Random Forest showed the best results with 95% accuracy. Additionally, the system also suggests the best time to use the fertilizers to boost up the yield.

CHAPTER 1

INTRODUCTION

1.1 GENERAL

Agriculture has an extensive history in India. Recently, India is ranked second in the farm output worldwide. Agriculture-related industries such as forestry and fisheries contributed for 16.6% of 2009 GDP and around 50% of the total workforce. Agriculture's monetary contribution to India's GDP is decreasing. The crop yield is the significant factor contributing in agricultural monetary. The crop yield depends on multiple factors such as climatic, geographic, organic, and financial elements. It is difficult for farmers to decide when and which crops to plant because of fluctuating market prices. Citing to Wikipedia figures India's suicide rate ranges from 1.4-1.8% per 100,000 populations, over the last 10 years. Farmers are unaware of which crop to grow, and what is the right time and place to start due to uncertainty in climatic conditions. The usage of various fertilizers is also uncertain due to changes in seasonal climatic conditions and basic assets such as soil, water, and air. In this scenario, the crop yield rate is steadily declining. The solution to the problem is to provide a smart user-friendly recommender system to the farmers. The crop yield prediction is a significant problem in the agriculture sector. Every farmer tries to know crop yield and whether it meets their expectations, thereby evaluating the previous experience of the farmer on the specific crop predict the yield. Agriculture yields rely primarily on weather conditions, pests, and preparation of harvesting operations. Accurate information on crop history is critical for making decisions on agriculture risk management. In this paper, we have proposed a model that addresses these issues. The novelty of the proposed system is to guide the farmers to maximize the crop yield as well as suggest the most profitable crop for the specific region.

1.2 OBJECTIVE

The proposed model provides crop selection based on economic and environmental conditions, and benefit to maximize the crop yield that will subsequently help to meet the increasing demand for the country's food supplies.

The proposed model predicts the crop yield by studying factors such as rainfall, temperature, area, season, soil type etc.

The system also helps to determine the best time to use fertilizers. The existing system which recommends crop yield is either hardware-based being costly to maintain, or not easily accessible.

temperature, area, season, soil type etc. The system also helps to determine the best time to use fertilizers. The existing system which recommends crop yield is either hardware-based being costly to maintain, or not easily accessible.

CHAPTER 2

LITERATURE SURVEY

A Review on Data Mining Techniques for Fertilizer Recommendation,2018

To keep up nutrition levels in the soil in case of deficiency, fertilizers are added to soil. The standard issue existing among the Indian agriculturists choose approximate amount of fertilizers and add them manually. Excess or deficient extension of fertilizers can harm the plants life and reduce the yield. This paper gives overview of various data mining frameworks used on cultivating soil dataset for fertilizer recommendation.

A Survey on Data Mining Techniques in Agriculture,2017

Agriculture is the most critical application area especially in the developing nations like India .Use of information technology in agriculture can change the situation of decision making and farmers can yield in better way. Data mining plays a important role in decision making on several aspects with agriculture field. It examines about role of data mining in the farming field and their related work by a few authors in related to agriculture domain. It additionally talks about on various data mining applications in taking care of the several agriculture problems. This paper integrates the work of several authors in a single place so it is valuable for specialists to get data of current situation of data mining systems and applications in context to farming field.

AgroNutri Android Application,2016

This paper communicates the idea regarding the making of AgroNutri an android application that helps in conveying the harvest particular fertilizer amount to be applied. The idea is to calculate the measure of NPK composts to be applied depend on the blanked proposal of the crop of interest. This application works

depend on the product chosen by the farmer and that is taken as input, thus providing the farmers. The future scope of the AgroNutri is that GPRS can be included so that according to location nutrients are suggested. Further this application would be incorporated as a piece of the accuracy agriculture wherein sensors can be utilized to discover the measure of NPK present in the dirt and that sum can be deducted from the suggestion and giving us the exact measure of supplements to be added.

Machine Learning: Applications in Indian Agriculture,2016

Agriculture is a field that has been lacking from adaption of technologies and their advancements. Indian agriculturists should be up to the mark with the universal procedures. Machine learning is a native concept that can be applied to every field on all inputs and outputs. It has effectively settled its ability over ordinary calculations of software engineering and measurements. Machine learning calculations have improved the exactness of artificial intelligence machines including sensor based frameworks utilized in accuracy farming. This paper has evaluated the different uses of machine learning in the farming area. It additionally gives a knowledge into the inconveniences looked by Indian farmers and how they can be resolved using these procedures.

Impacts of population growth, economic development, and technical change on global food production and consumption,2015

Throughout the following decades humanity will request more food from less land and water assets. This investigation evaluates the food production effects of four elective advancement situations from the Millennium Ecosystem Assessment and the Special Report on Emission Scenarios. partially and jointly considered are land and water supply impacts from population development, and specialized change, and forests and agriculture demand request shifts from population development

and economic improvement. The income impacts on nourishment request are registered with dynamic flexibilities. Worldwide farming area increments by up to 14% somewhere in the range of 2010 and 2030. Deforestation restrictions strongly impact the price of land and water resources but have little consequences for the global level of food production and food prices. While projected income changes have the highest partial impact on per capita food consumption levels, population growth leads to the highest increase in total food production. The impact of technical change is amplified or mitigated by adaptations of land management intensities

Brief history of agricultural systems modelling, 2016

Rural frameworks science creates information that enables analysts to consider complex issues or take educated farming choices. The rich history of this science represents the decent variety of frameworks and scales over which they work and have been contemplated. Demonstrating, a basic apparatus in agrarian frameworks science, has been expert by researchers from an extensive variety of controls, who have contributed ideas and instruments over six decades. As agrarian researchers currently consider the "people to come" models, information, and learning items expected to meet the inexorably mind boggling frameworks issues looked by society, it is vital to check out this history and its exercises to guarantee that we stay away from re-innovation and endeavor to think about all elements of related difficulties. To this end, we outline here the historical backdrop of rural frameworks demonstrating and distinguish exercises discovered that can help control the structure and advancement of up and coming age of farming framework apparatuses and techniques. Various past occasions joined with generally innovative advancement in different fields have unequivocally added to the development of farming framework demonstrating, including improvement of process-based bio-physical models of yields and domesticated animals, factual models dependent on verifiable perceptions, and financial

streamlining and reproduction models at family unit and local to worldwide scales. Attributes of rural frameworks models have changed broadly relying upon the frameworks included, their scales, and the extensive variety of purposes that spurred their advancement and use by specialists in various controls. Late patterns in more extensive joint effort crosswise over establishments, crosswise over orders, and between people in general and private segments recommend that the stage is set for the significant advances in rural frameworks science that are required for the up and coming age of models, databases, learning items and choice emotionally supportive networks. The exercises from history ought to be considered to help stay away from barricades and entanglements as the network builds up this up and coming age of horticultural frameworks models.

A Smart Agricultural Model by Integrating Iot, Mobile and Cloud-based Big Data Analytics,2017

In the cultivating field, the system models play a significant role to the enhancement of the agro-normal and money related conditions. In the proportions of benefits of the field and farm examinations to give the information and to recognize fitting and fruitful organization practices. It can recognize the organization to arrive managers and transversely over reality as long as the required soil, the board, environment, and money related information. Decision Support Systems (DSSs) use to make the information for the vermin the board, develop the officials. These systems are not using the impelled strategies to process the data. Thusly, use the adroit system thoughts to take the decisions for the issue. It expect a crucial activity in the comprehension of agronomic results, and their use as decision sincerely steady systems for farmers is extending.

An Overview of Internet of Things and Data Analytics in Agriculture: Benefits and Challenges,2018

A blueprint of Iot and DA in agriculture has been shown in this paper. A couple of zones related to the association of Iot in agribusiness have been discussed in detail. The investigation of composing exhibits that there are clusters of work advancing being produced of Iot development that can be used to increase operational efficiency and gainfulness of plant and creatures. The benefits of Iot and DA, and open troubles have been identified and inspected in this paper. Iot is depended upon to offer a couple of benefits to the agribusiness division. Regardless, there are up 'til now different issues to be steered to make it moderate for close to nothing and medium-scale farmers. The key issues are security and cost. It is typical that as contention increases in the cultivating part

Circulation Mode Selection Based on Cost Analysis,2017

If every farmer and each average production base will join their optimal conditions in making cooperatives, it will accomplish economies of scale. Furthermore, producers will have an all the more favourable position in the plans with downstream firms (shipper or retailer).Second, the main customers of wholesale market are not inhabitants nearby who buy small quantities products but lower distributors or retailers. More redesigned transportation mode respects intensive attempt of new agrarian things, which prompts bolster the movement of new chain joint logistics and strengthen resource utilize and made logistics advantage quality. Refresh everything considered agrarian things spread. By then, regard the examination of gigantic worth control of standard things and achieve the mind blowing control to stream process.

Support Vector Machine-based Fuzzy Self-learning Control for Induction Machines,2010

Using support vector machine (SVM) is to realize the self learning of fuzzy inference system (FIS), based on a fast modified varying metric method (MDFP) and a support vector machine identifier (SVMI), a SVM-FIS self-learning

controller for the threephase induction machine adjustable speed system has been designed. The proposed controller is not only of the advantages that FIS does not depend on the plant model, strong robustness, and adaptive self-learning ability, but also learning ability and generalization performance of SVM. The designed processes of SVM-FIS, MDFFP, and SVMFI algorithms have been described in details. Simulation results show the feasibility, correctness and effectiveness of the proposed control strategy, such as the excellent static and dynamic performances, and strong anti-interference ability.

CHAPTER 3

METHODOLOGIES

Extensive work has been done, and many ML algorithms have been applied in the agriculture sector. The biggest challenge in agriculture is to increase farm production and offer it to the end-user with the best possible price and quality. It is also observed that at least 50% of the farm produce gets wasted, and it never reaches the end-user. The proposed model suggests the methods for minimizing farm produce wastage.

3.1 EXISTING SYSTEM

One of the recent works, S. Pavani et.al. presented a model where the crop yield is predicted using KNN algorithms by making the clusters. It has been shown that KNN clustering proved much better than SVM or regression. Nishant et. al. predicted the crop yield for the specific year with the help of advanced regression techniques like Enet, Lasso and Kernel Ridge algorithms. The Stacking regression helped to enhance the accuracy of the algorithms.

DISADVANTAGES OF EXISTING SYSTEM

- The main challenge faced in agriculture sector is the lack of knowledge about the changing variations in climate. Each crop has its own suitable climatic features. This can be handled with the help of precise farming techniques. The precision farming not only maintains the productivity of crops but also increases the yield rate of production.
- The existing system which recommends crop yield is either hardware-based being costly to maintain, or not easily accessible.
- Despite many solutions that have been recently proposed, there are still open challenges in creating a user-friendly application with respect to crop recommendation.

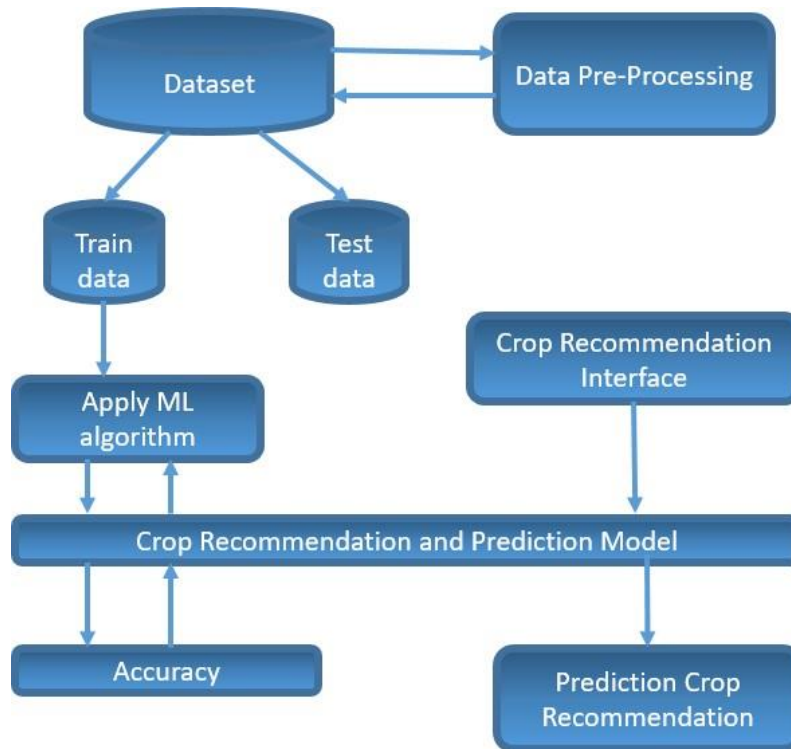
3.2 PROPOSED SYSTEM

- In this project, we have proposed a model that addresses the existing issues. The novelty of the proposed system is to guide the farmers to maximize the crop yield as well as suggest the most profitable crop for the specific region.
- The proposed model provides crop selection based on economic and environmental conditions, and benefit to maximize the crop yield that will subsequently help to meet the increasing demand for the country's food supplies. The proposed model predicts the crop yield by studying factors such as rainfall, temperature, area, season, soil type etc. The system also helps to determine the best time to use fertilizers.
- The user provides an area under cultivation and soil type as inputs. According to the requirement, the model predicts the crop yield for a specific crop. The model also recommends the most profitable crop and suggests the right time to use the fertilizers.
- The main objective is to obtain a better variety of crops that can be grown over the season. The proposed system would help to minimize the difficulties faced by farmers in choosing a crop and maximize the yield.

ADVANTAGES OF PROPOSED SYSTEM

- The proposed model predicts the crop yield for the data sets of the given region. Integrating agriculture and ML will contribute to more enhancements in the agriculture sector by increasing the yields and optimizing the resources involved. The data from previous years are the key elements in forecasting current performance.
- The proposed system uses recommender system to suggest the right time for using fertilizers.
- The methods in the proposed system includes increasing the yield of crops, real-time analysis of crops, selecting efficient parameters, making smarter decisions and getting better yield.

SYSTEM ARCHITECTURE



3.3 SYSTEM REQUIREMENTS

3.4 HARDWARE

REQUIREMENTS:

- System - Pentium-IV
- Speed - 2.4GHZ
- Hard disk - 40GB
- Monitor - 15VGA color
- RAM - 512MB

3.5 SOFTWARE REQUIREMENTS:

- Operating System - Windows XP
- Coding language - Python

4 .SOFTWARE ENVIRONMENT

4.1 Python:

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted** – Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive** – You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented** – Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- **Python is a Beginner's Language** – Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

4.2 History of Python

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

4.3 Python Features

Python's features include –

- **Easy-to-learn** – Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- **Easy-to-read** – Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** – Python's source code is fairly easy-to-maintain.
- **A broad standard library** – Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** – Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable** – Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- **Extendable** – You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- **Databases** – Python provides interfaces to all major commercial databases.

- **GUI Programming** – Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- **Scalable** – Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below –

- It supports functional and structured programming methods as well as OOP.
- It can be used as a scripting language or can be compiled to byte-code for building large applications.
- It provides very high-level dynamic data types and supports dynamic type checking.
- It supports automatic garbage collection.
- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

Python is available on a wide variety of platforms including Linux and Mac OS X. Let's understand how to set up our Python environment.

4.4 Getting Python

The most up-to-date and current source code, binaries, documentation, news, etc., is available on the official website of Python <https://www.python.org>.

Windows Installation

Here are the steps to install Python on Windows machine.

- Open a Web browser and go to <https://www.python.org/downloads/>.

- Follow the link for the Windows installer python-XYZ.msifile where XYZ is the version you need to install.
- To use this installer python-XYZ.msi, the Windows system must support Microsoft Installer 2.0. Save the installer file to your local machine and then run it to find out if your machine supports MSI.
- Run the downloaded file. This brings up the Python install wizard, which is really easy to use. Just accept the default settings, wait until the install is finished, and you are done.

The Python language has many similarities to Perl, C, and Java. However, there are some definite differences between the languages.

First Python Program

Let us execute programs in different modes of programming.

Interactive Mode Programming

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

```
$ python

Python2.4.3(#1,Nov112010,13:34:43)

[GCC 4.1.220080704(RedHat4.1.2-48)] 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!
```

4.5 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 –

```
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!
```

4.6 Flask Framework:

Flask is a web application framework written in Python. Armin Ronacher, who leads an international group of Python enthusiasts named Pocco, develops it.

Flask is based on Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects.

Http protocol is the foundation of data communication in world wide web. Different methods of data retrieval from specified URL are defined in this protocol.

The following table summarizes different http methods –

Sr.No	Methods & Description
1	GET Sends data in unencrypted form to the server. Most common method.
2	HEAD Same as GET, but without response body
3	POST Used to send HTML form data to server. Data received by POST method is not cached by server.
4	PUT Replaces all current representations of the target resource with the uploaded content.

5	DELETE Removes all current representations of the target resource given by a URL
---	--

By default, the Flask route responds to the **GET** requests. However, this preference can be altered by providing methods argument to **route()** decorator.

In order to demonstrate the use of **POST** method in URL routing, first let us create an HTML form and use the **POST** method to send form data to a URL.

Save the following script as login.html

```
<html>

<body>

<form action="http://localhost:5000/login" method="post">

<p>Enter Name:</p>

<p><input type="text" name="nm"/></p>

<p><input type="submit" value="submit"/></p>

</form>

</body>

</html>
```

Now enter the following script in Python shell.

```

from flask import Flask, redirect, url_for, request

app=Flask(__name__)

@app.route('/success/<name>')

def success(name):

return'welcome %s'% name

@app.route('/login',methods=['POST','GET'])

def login():

if request.method=='POST':

user=request.form['nm']

return redirect(url_for('success',name= user))

else:

user=request.args.get('nm')

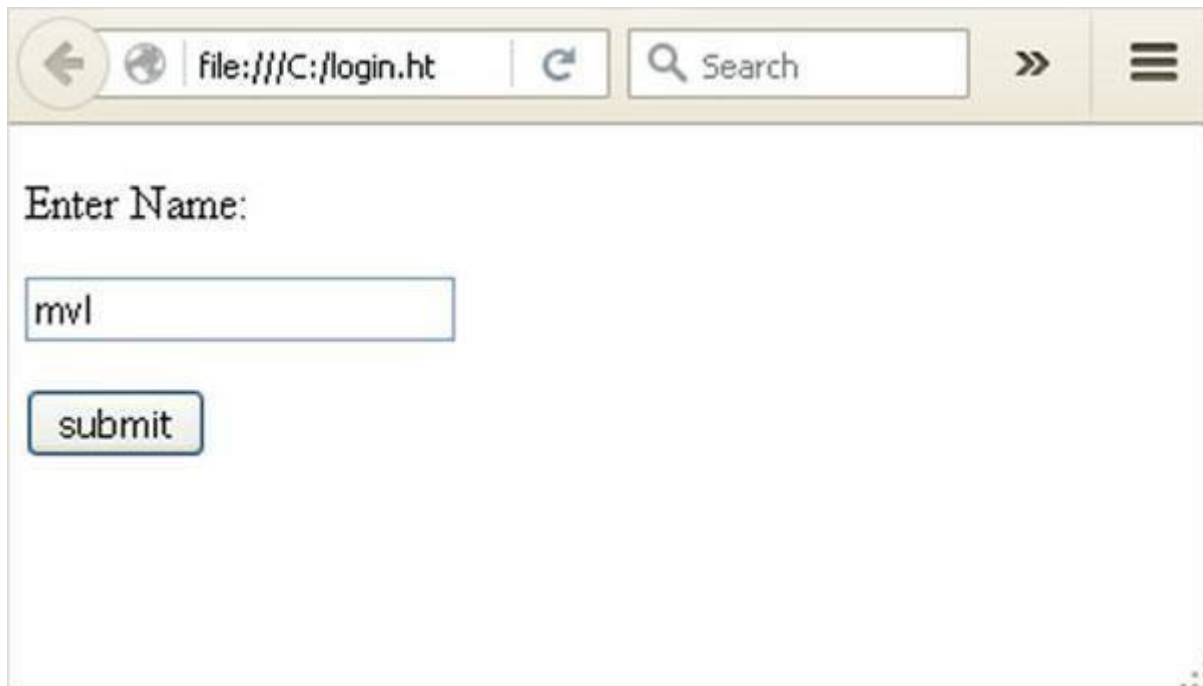
return redirect(url_for('success',name= user))

if __name__ == '__main__':

app.run(debug =True)

```

After the development server starts running, open **login.html** in the browser, enter name in the text field and click **Submit**.

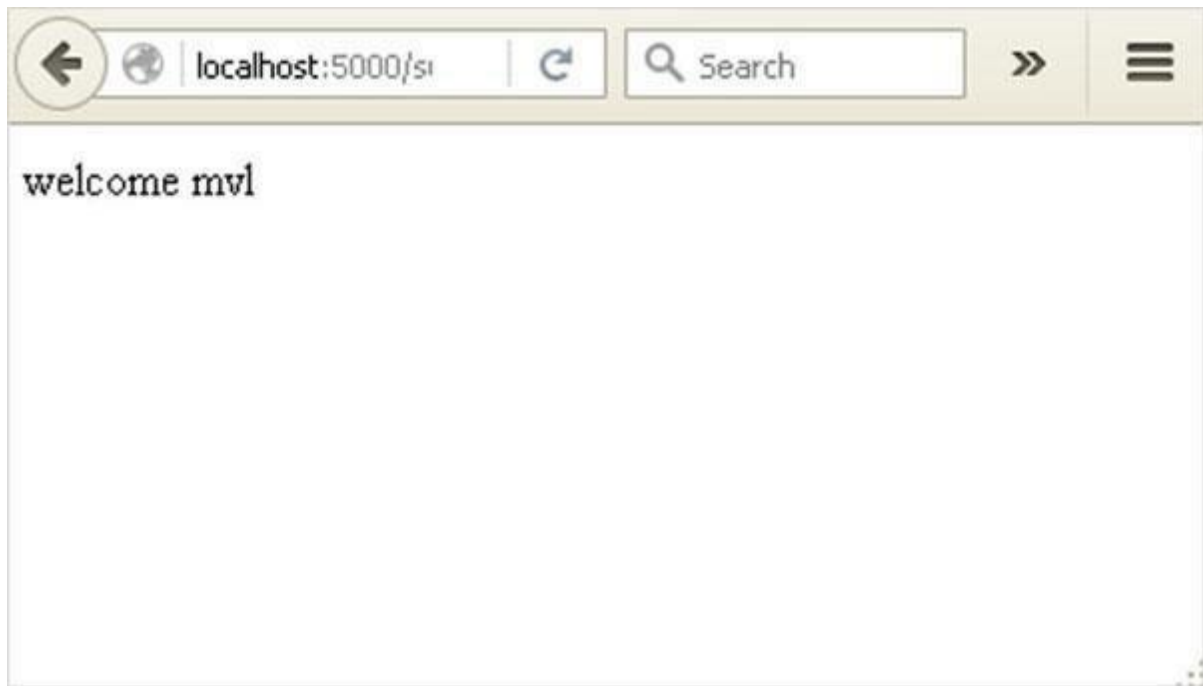
A screenshot of a web browser window. The address bar shows the file path 'file:///C:/login.ht'. The page content includes the text 'Enter Name:', a text input field containing 'mvl', and a 'submit' button.

Form data is POSTed to the URL in action clause of form tag.

http://localhost/login is mapped to the **login()** function. Since the server has received data by **POST** method, value of 'nm' parameter obtained from the form data is obtained by –

```
user = request.form['nm']
```

It is passed to '**success**' URL as variable part. The browser displays a **welcome** message in the window.



Change the method parameter to '**GET**' in **login.html** and open it again in the browser. The data received on server is by the **GET** method. The value of 'nm' parameter is now obtained by –

```
User = request.args.get('nm')
```

Here, **args** is dictionary object containing a list of pairs of form parameter and its corresponding value. The value corresponding to 'nm' parameter is passed on to '/success' URL as before.

What is Python?

Python is a popular programming language. It was created in 1991 by Guido van Rossum.

It is used for:

- web development (server-side),
- software development,
- mathematics,
- system scripting.

What can Python do?

- Python can be used on a server to create web applications.

- Python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.
- Python can be used to handle big data and perform complex mathematics.
- Python can be used for rapid prototyping, or for production-ready software development.

Why Python?

- Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
- Python has a simple syntax similar to the English language.
- Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
- Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
- Python can be treated in a procedural way, an object-orientated way or a functional way. Good to know
- The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.
- In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse which are particularly useful when managing larger collections of Python files.

Python Syntax compared to other programming languages

- Python was designed to for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

4.7 Python Install

Many PCs and Macs will have python already installed.

To check if you have python installed on a Windows PC, search in the start bar for Python or run the following on the Command Line (cmd.exe):

```
C:\Users\Your Name>python --version
```

To check if you have python installed on a Linux or Mac, then on linux open the command line or on Mac open the Terminal and type:

```
python --version
```

If you find that you do not have python installed on your computer, then you can download it for free from the following website: <https://www.python.org/>

Python Quickstart

Python is an interpreted programming language, this means that as a developer you write Python (.py) files in a text editor and then put those files into the python interpreter to be executed.

The way to run a python file is like this on the command line:

```
C:\Users\Your Name>python helloworld.py
```

Where "helloworld.py" is the name of your python file.

Let's write our first Python file, called helloworld.py, which can be done in any text editor.

```
helloworld.py
```

```
print("Hello, World!")
```

Simple as that. Save your file. Open your command line, navigate to the directory where you saved your file, and run:

```
C:\Users\Your Name>python helloworld.py
```

The output should read:

```
Hello, World!
```

Congratulations, you have written and executed your first Python program.

The Python Command Line

To test a short amount of code in python sometimes it is quickest and easiest not to write the code in a file. This is made possible because Python can be run as a command line itself.

Type the following on the Windows, Mac or Linux command line:

```
C:\Users\Your Name>python
```

From there you can write any python, including our hello world example from earlier in the tutorial:

```
C:\Users\Your Name>python
Python 3.6.4 (v3.6.4:d48eceb, Dec 19 2017, 06:04:45) [MSC v.1900 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello, World!")
```

Which will write "Hello, World!" in the command line:

```
C:\Users\Your Name>python
Python 3.6.4 (v3.6.4:d48eceb, Dec 19 2017, 06:04:45) [MSC v.1900 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello, World!") Hello,
World!
```

Whenever you are done in the python command line, you can simply type the following to quit the python command line interface:

```
exit()
```

Execute Python Syntax

As we learned in the previous page, Python syntax can be executed by writing directly in the Command Line:

```
>>>                                     print("Hello,                               World!")
```

Hello, World!

Or by creating a python file on the server, using the .py file extension, and running it in the Command Line:

```
C:\Users\Your Name>python myfile.py
```

Python Indentations

Where in other programming languages the indentation in code is for readability only, in Python the indentation is very important. Python uses indentation to indicate a block of code. Example if $5 > 2$:

```
print("Five is greater than two!")
```

Python will give you an error if you skip the indentation: Example if $5 > 2$:

```
print("Five is greater than two!")
```

Comments

Python has commenting capability for the purpose of in-code documentation.

Comments start with a #, and Python will render the rest of the line as a comment:

Example

Comments in Python:

```
#This is a comment. print("Hello, World!")
```

Docstrings

Python also has extended documentation capability, called docstrings.

Docstrings can be one line, or multiline.

Python uses triple quotes at the beginning and end of the docstring:

Example

Docstrings are also comments:

```
"""This is a multiline docstring.""" print("Hello, World!")
```

5. MODULES

- Admin Login
- Metadata

- Data Pre-processing
- Crop Prediction Module
- Crop Recommendation Module

MODULE DESCRIPTION

5.1 Admin login

This is the first activity, Admin needs to provide a correct contact number and a password, which user enters while registering, in order to login into the webpage. If information provided by the admin matches with the data in the database table then user successfully login into the webpage else message of login failed is displayed and user need to re-enter correct information.

5.2 Metadata

All the main data used in the data set are initialized with the number to use in the algorithm it is like initializing all the details. In this metadata, we are going to initialize all the crop names with the numbers. This data makes us use the data easily in the algorithm. Hear the metadata of all the crops is given with a particular number. This number is not duplicated that is one number is given to one crop, the same number is not given to the other crop. This metadata consists of more than a hundred crops that grown all over India.

5.3 Data Pre-processing

Hear the raw data in the crop data is cleaned and the metadata is appending to it by removing the things which are converted to the integer. So, the data is easy to train. Hear all the data. In this pre-processing, we first load the metadata into this and then this metadata will be attached to the data and replace the converted data with metadata. Then this data will be moved further and remove the unwanted data in the list and it will divide the data into the train and the test data.

5.4 Crop Prediction Module

The obtained result will be helpful for the farmers to know the Yield of the crop so, he can go for the better crop which gives high yield and also say them the efficient use of agriculture field. This way we can help the farmers to grow the crop which gives them better yield.

5.5 Crop Recommendation Module

In this module, we have proposed a model that addresses these issues. The novelty of the proposed system is to guide the farmers to maximize the crop yield as well as suggest the most profitable crop for the specific region.

5.6 REQUIREMENT ANALYSIS

Requirement analysis, also called requirement engineering, is the process of determining user expectations for a new modified product. It encompasses the tasks that determine the need for analysing, documenting, validating and managing software or system requirements. The requirements should be documentable, actionable, measurable, testable and traceable related to identified business needs or opportunities and define to a level of detail, sufficient for system design.

FUNCTIONAL REQUIREMENTS

It is a technical specification requirement for the software products. It is the first step in the requirement analysis process which lists the requirements of particular software systems including functional, performance and security requirements. The function of the system depends mainly on the quality hardware used to run the software with given functionality.

Usability

It specifies how easy the system must be use. It is easy to ask queries in any format which is short or long, porter stemming algorithm stimulates the desired response for user.

Robustness

It refers to a program that performs well not only under ordinary conditions but also under unusual conditions. It is the ability of the user to cope with errors for irrelevant queries during execution.

Security

The state of providing protected access to resource is security. The system provides good security and unauthorized users cannot access the system there by providing high security.

Reliability

It is the probability of how often the software fails. The measurement is often expressed in MTBF (Mean Time Between Failures). The requirement is needed in order to ensure that the processes work correctly and completely without being aborted. It can handle any load and survive and survive and even capable of working around any failure.

Compatibility

It is supported by version above all web browsers. Using any web servers like localhost makes the system real-time experience.

Flexibility

The flexibility of the project is provided in such a way that it has the ability to run on different environments being executed by different users.

Safety

Safety is a measure taken to prevent trouble. Every query is processed in a secured manner without letting others to know one's personal information.

NON- FUNCTIONAL REQUIREMENTS

Portability

It is the usability of the same software in different environments. The project can be run in any operating system.

Performance

These requirements determine the resources required, time interval, throughput and everything that deals with the performance of the system.

Accuracy

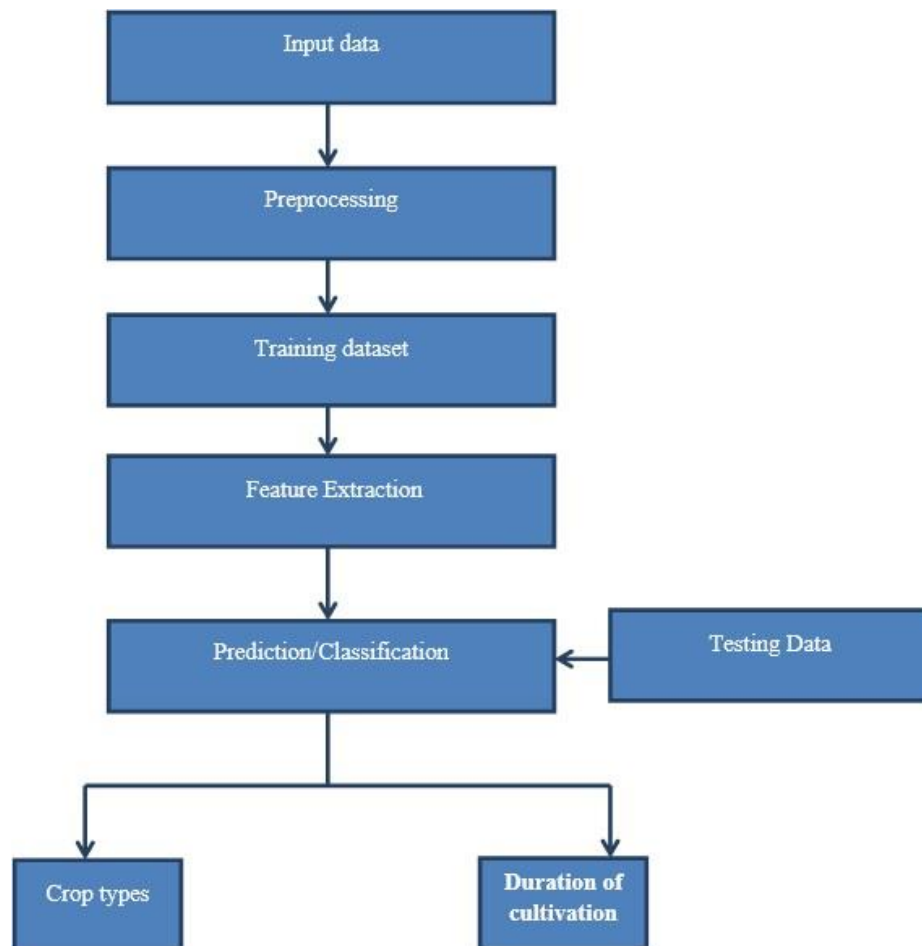
The result of the requesting query is very accurate and high speed of retrieving information. The degree of security provided by the system is high and effective.

Maintainability

Project is simple as further updates can be easily done without affecting its stability. Maintainability basically defines that how easy it is to maintain the system. It means that how easy it is to maintain the system, analyse, change and test the application. Maintainability of this project is simple as further updates can be easily done without affecting its stability.

5.7 DATA FLOW DIAGRAM:

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.



UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

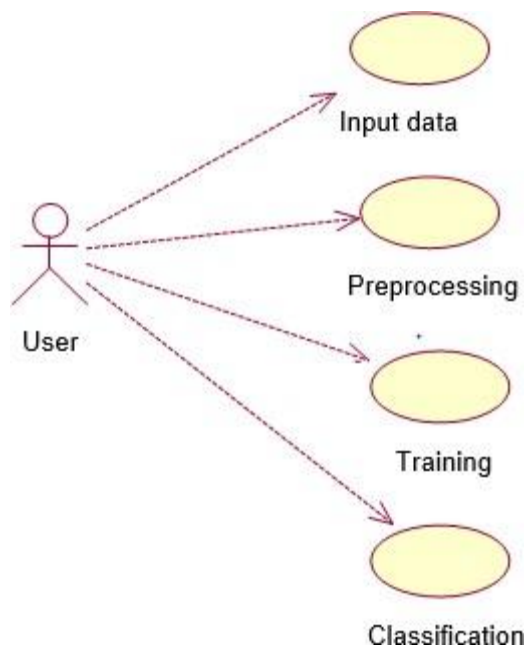
The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

USE CASE DIAGRAM:

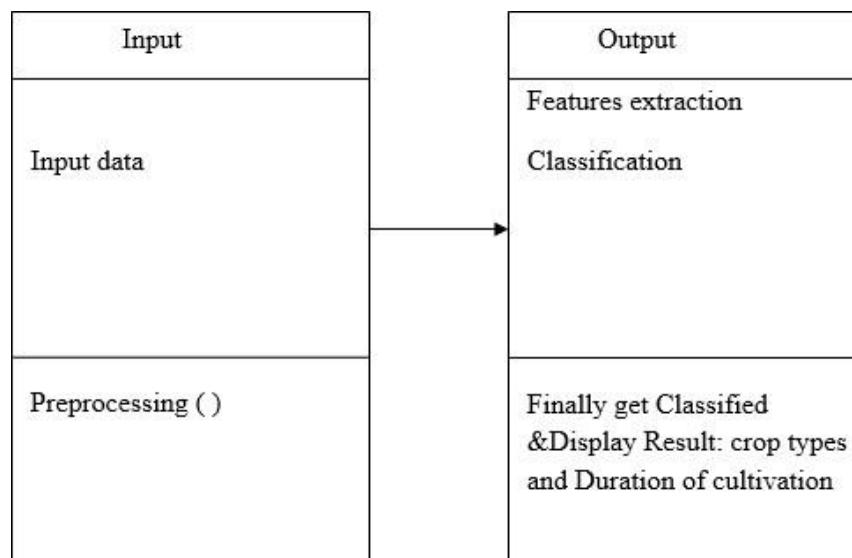
A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose

is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



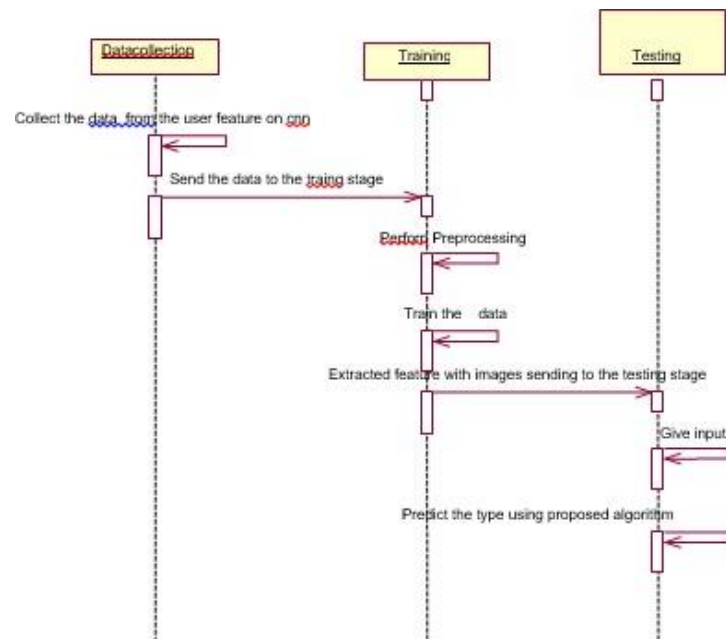
CLASS DIAGRAM:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



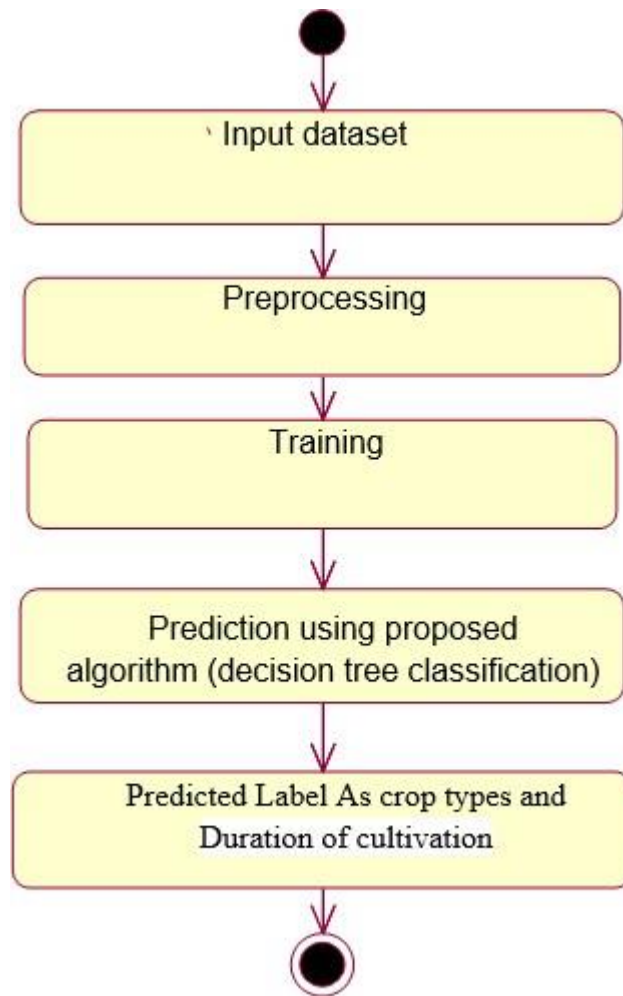
SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagram.



ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



INPUT DESIGN AND OUTPUT 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

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 the 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.

SYSTEM STUDY

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

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.

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.

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.

SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

TYPES OF TESTS

Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and

consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

- Valid Input : identified classes of valid input must be accepted.
- Invalid Input : identified classes of invalid input must be rejected.
- Functions : identified functions must be exercised.
- Output : identified classes of application outputs must be exercised.
- Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

6.1 Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

6.2 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

6.3 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

CHAPTER 6 RESULT AND DISCUSSION

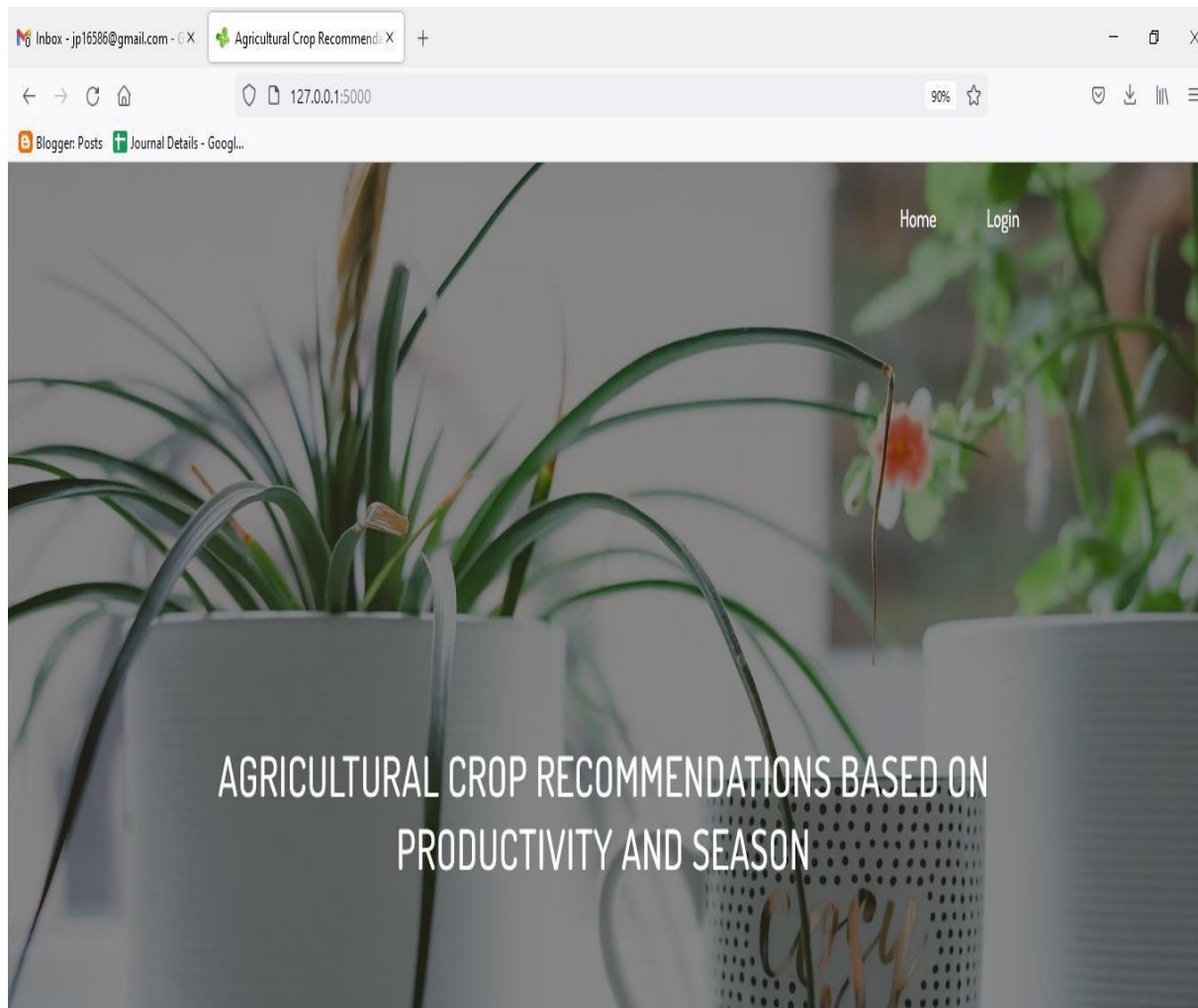
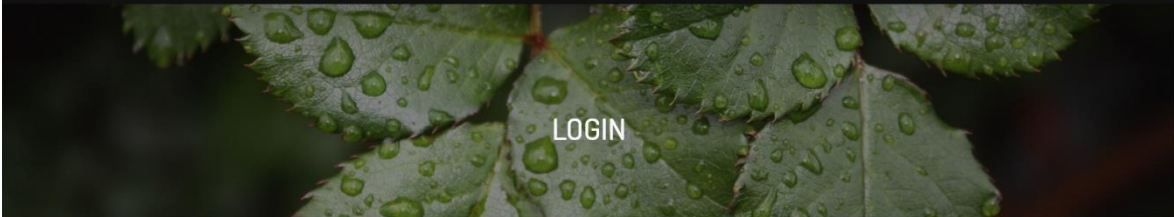


FIGURE OF 6.1 : HOME SCREEN

[Home](#) [Login](#) [Upload](#) [Preview](#)



LOGIN

LOGIN

Username

admin

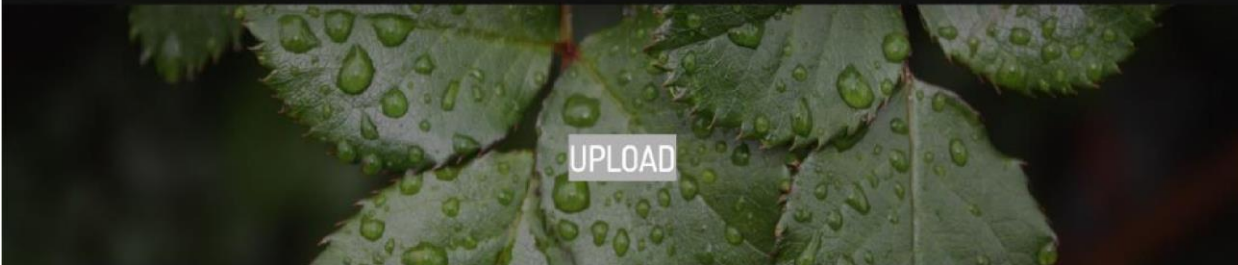
Password

•••••

Login



[Home](#) | [Login](#) | [Upload](#) | [Preview](#)



UPLOAD

UPLOAD

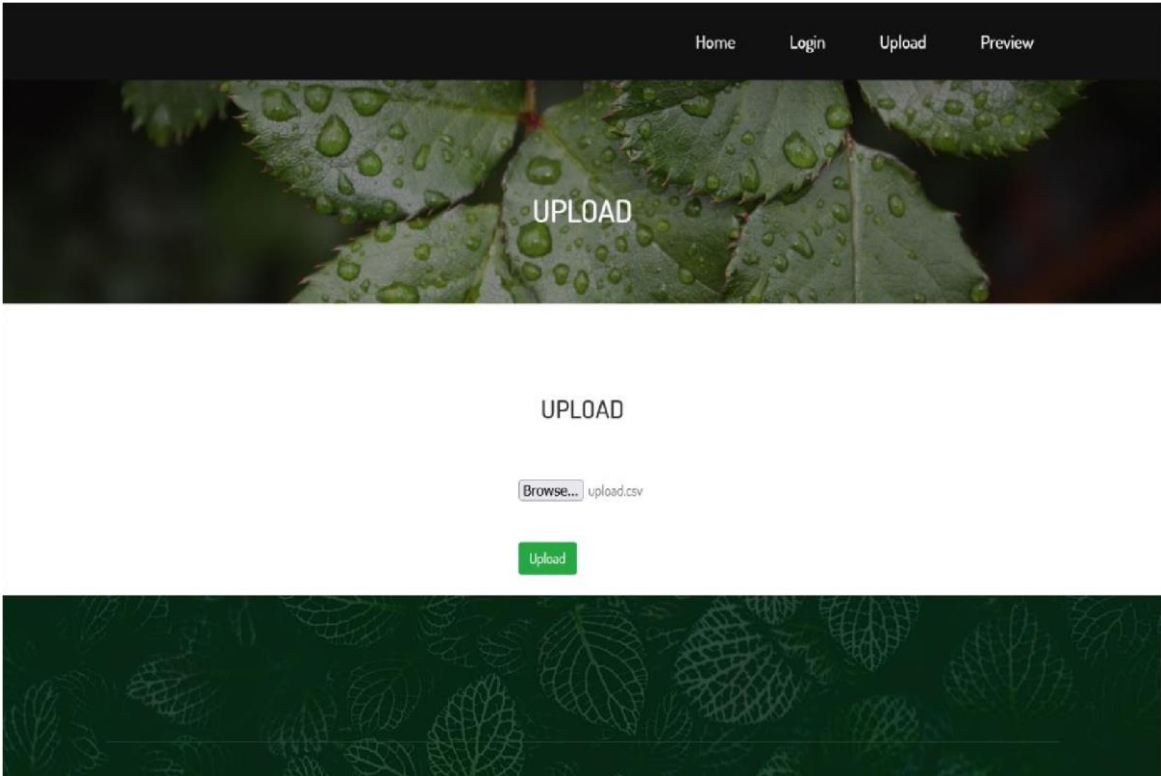
Browse...

No file selected.

Upload



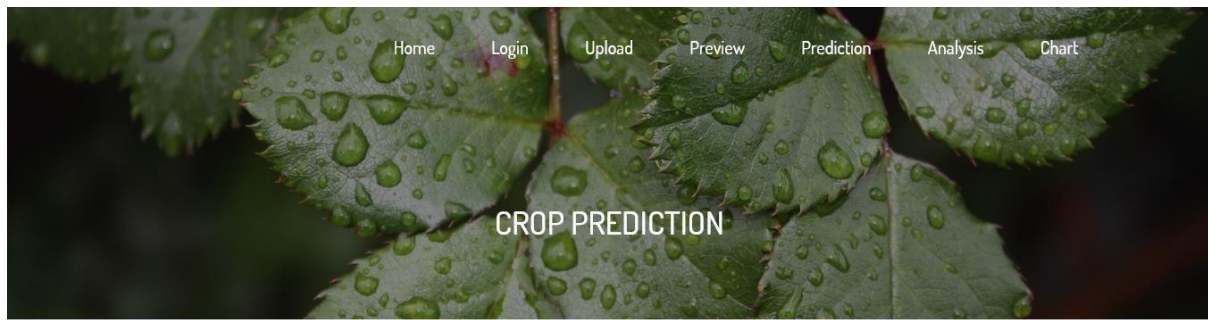
Fig 6.2: upload data set
48



FIGHRE OF 6.2: Login page

Manipur	148.80	4.00	30.5	Black	Kharif	Bajra	Nitrogen,Phosphorus,Potash	8000	70414.47817	4.5kg
Odisha	215.00	5.00	30.0	Latterite	Kharif	Bajra	Nitrogen,Phosphorus,Potash	8000	70414.47817	4.5kg
Odisha	215.00	5.00	30.5	Black	Kharif	Bajra	Nitrogen,Phosphorus,Potash	8000	70414.47817	4.5kg
Punjab	60.00	8.90	30.0	Latterite	Kharif	Bajra	Nitrogen,Phosphorus,Potash	8000	70414.47817	4.5kg
Punjab	60.00	8.90	30.5	Black	Kharif	Bajra	Nitrogen,Phosphorus,Potash	8000	70414.47817	4.5kg

	States	Rainfall	Ground Water	Temperature	Soil type	Season	Crops	Fertilisers re
Id								
1	Andhra Pradesh	110.75	9.74	31.5	Clayey	Kharif	Paddy IGVVR-2 (IET 19795)	Potassium,Urea
2	Andhra Pradesh	110.75	9.74	31.5	Alluvial	Kharif	Paddy IGVVR-2 (IET 19795)	Potassium,Urea



CROP PREDICTION

Enter the details

State:

Rainfall:

Ground Water:

Temperature:

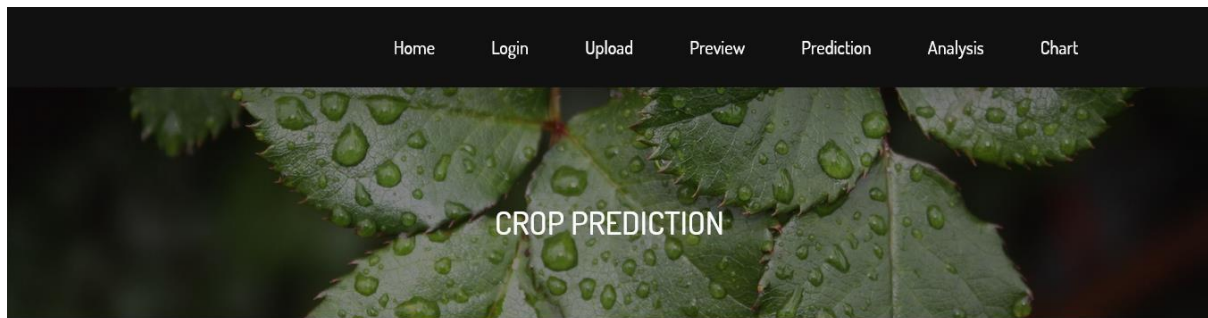
Soil type:

Season:

Prediction is

A decorative horizontal bar with a dark green background and a repeating pattern of light green leaf outlines.

Fig 6.3: Enter the deatails



CROP PREDICTION

Enter the details

State:

Rainfall:

Ground Water:

Temperature:

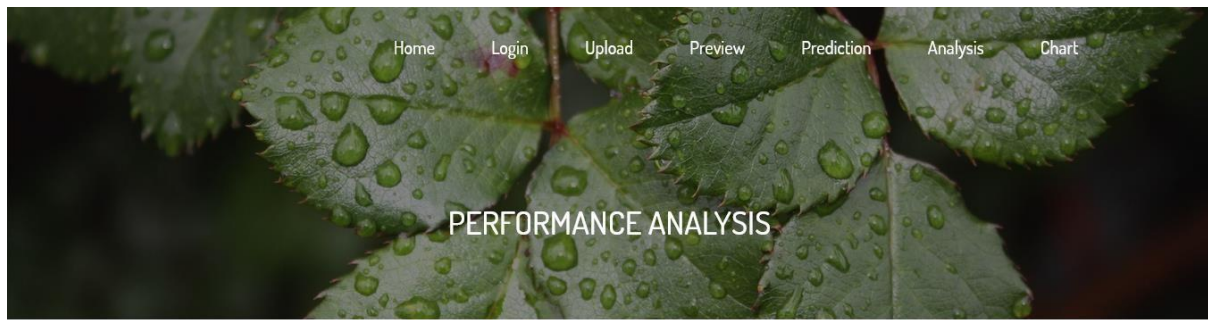
Soil type:

Season:

Prediction is

Crop: Sweet potato Duration of cultivation: 120-125

Fig 6.4: Prediction



performance analysis

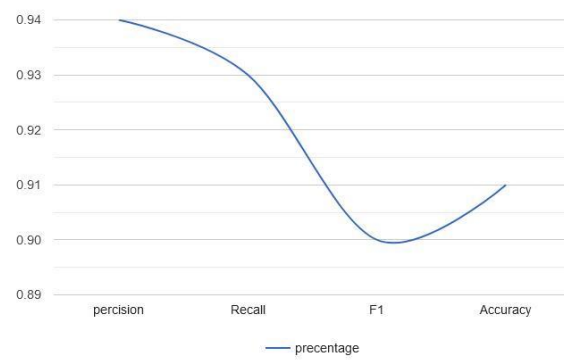
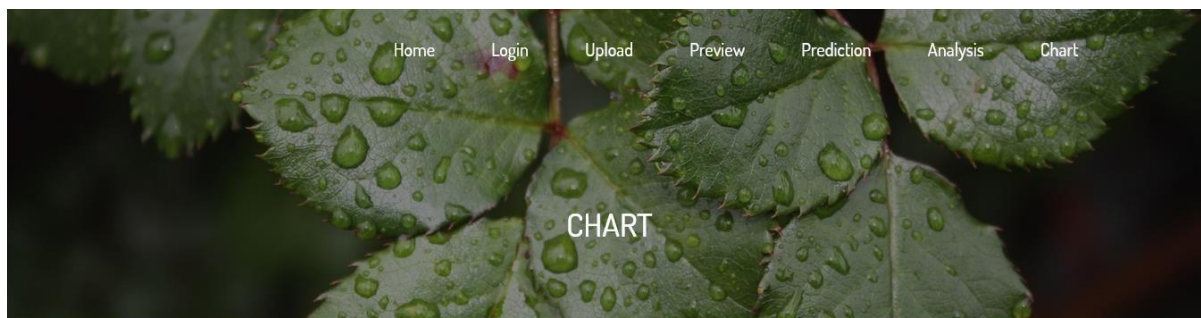
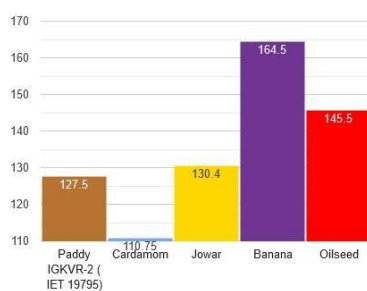


Fig 6.5: Performance analysis



Crop recommendation for Rainfall (Top 5 crop)



Crop recommendation for Temperature (Top 5 crop)

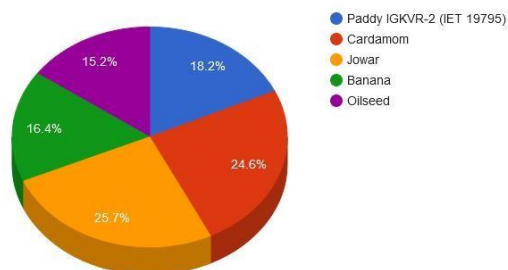


Figure 6.6: Crop recommendation

APPENDICES

SOURCE CODE

```
import numpy as np import pandas as pd from flask import Flask, request,
jsonify,    render_template,    redirect,    flash,    send_file    from
sklearn.preprocessing import MinMaxScaler from werkzeug.utils import
secure_filename import pickle
```

```
import numpy as np import pandas as pd from flask import Flask, request,
jsonify,    render_template,    redirect,    flash,    send_file    from
sklearn.preprocessing import MinMaxScaler from werkzeug.utils import
secure_filename import pickle import numpy as np import pandas as pd
from sklearn.tree import DecisionTreeClassifier
```

```
from sklearn.ensemble import RandomForestClassifier, BaggingClassifier,
AdaBoostClassifier, VotingClassifier
app = Flask(__name__) #Initialize the flask App
```

```
#forest = pickle.load(open('boosting.pkl','rb')) crop
```

```
= pickle.load(open('crop.pkl','rb'))
```

```
@app.route('/')
```

```
@app.route('/index') def
```

```
index():
```

```
    return render_template('index.html')
```

```
@app.route('/analysis') def
```

```
analysis():
```

```
    return render_template('analysis.html')
```

```
@app.route('/chart') def
```

```
chart():
```

```
    return render_template('chart.html')
```

```
#@app.route('/future')
```

```
#def future():
```

```
#    return render_template('future.html') @app.route('/login') def login():
```

```
    return render_template('login.html')
```

```
@app.route('/upload')
```

```
def upload():
```

```
    return render_template('upload.html')
```

```
@app.route('/preview',methods=["POST"])
```

```
def preview():
```

```
if request.method == 'POST':  
  
    dataset = request.files['datasetfile'] df =  
  
    pd.read_csv(dataset,encoding = 'unicode_escape')  
  
    df.set_index('Id', inplace=True)  
  
    return render_template("preview.html",df_view = df)
```

```
#@app.route('/home')
```

```
#def home():
```

```
# return render_template('home.html')
```

```
@app.route('/prediction', methods = ['GET', 'POST']) def
```

```
prediction():
```

```
    return render_template('prediction.html') #@app.route('/upload')
```

```
#def upload_file():
```

```
# return render_template('BatchPredict.html')
```

```
@app.route('/predict',methods=['POST']) def
```

```
predict():
```

```
int_feature = [x for x in request.form.values()]
```

```
final_features = [np.array(int_feature)]
```

```
y_pred=crop.predict(final_features)
```

```
if y_pred[0] == 'Paddy IGKVR-2 (IET 19795)':
```

```
    label="Crop: Paddy IGKVR-2 (IET 19795) Duration of  
    cultivation: 105-123"
```

```
elif y_pred[0] == 'Paddy CR Dhan 501 (IET 19189)':
```

```
    label="Crop: Paddy CR Dhan 501 (IET 19189) Duration of  
    cultivation: 105-126"
```

```
elif y_pred[0] == 'Wheat VL Gehun 907 (VL 907)':
```

```
    label="Crop: Wheat VL Gehun 907 (VL 907)) Duration of  
    cultivation: 60-154"
```

```
elif y_pred[0] == 'Wheat WHD 943':
```

```
    label="Crop: Wheat WHD 943 Duration of cultivation: 60-  
    157" elif y_pred[0] == 'Millet Nandi-65 (MH-1549)':
```

```
    label="Crop: Millet Nandi-65 (MH-1549) Duration of  
    cultivation: 65-70"
```

```
elif y_pred[0] == 'Lentil Pant Lentil-8(Pant L-063)':
```

```
    label="Crop: Lentil Pant Lentil-8(Pant L-063) Duration of
```

cultivation: 45-60"

elif y_pred[0] == 'Bajra':

label="Crop: Bajra Duration of cultivation: 45-50" elif

y_pred[0] == 'Cardamom': label="Crop: Cardamom Duration of

cultivation: 730-735" elif y_pred[0] == 'Urad': label="Crop:

Urad Duration of cultivation: 70-85" elif y_pred[0] == 'Jowar':

label="Crop: Jowar Duration of cultivation: 65-75" elif

y_pred[0] == 'Paddy CR Dhan 401 (REETA)':

label="Crop: Paddy CR Dhan 401 (REETA) Duration of

cultivation: 105-124 " elif y_pred[0] == 'Millet MH 1540

(86M64) (Hybrid)': label="Crop: Millet MH 1540

(86M64) (Hybrid) Duration of

cultivation: 65-73"

elif y_pred[0] == 'Sugarcane Karan 5 (Co 0124)':

label="Crop: Sugarcane Karan 5 (Co 0124) Duration of

cultivation: 300-451"

elif y_pred[0] == 'Banana':

label="Crop: Banana Duration of cultivation: 365-370" elif

y_pred[0] == 'Arhar': label="Crop: Arhar Duration of cultivation:

120-200" elif y_pred[0] == 'Clove': label="Crop: Clove Duration of

cultivation: 120-180 " elif y_pred[0] == 'Oilseed': label="Crop:

Oilseed Duration of cultivation: 110-115" elif y_pred[0] == 'Tea':

**label="Crop: Tea Duration of cultivation: 60-65" elif y_pred[0] ==
'Coffee': label="Crop: Coffee Duration of cultivation: 240-270" elif
y_pred[0] == 'Turmeric': label="Crop: Turmeric Duration of
cultivation: 210-270" elif y_pred[0] == 'Cashewnut': label="Crop:
Cashewnut Duration of cultivation: 1030-1035" elif y_pred[0] ==
'Ragi':**

elif

label="Crop: Ragi Duration of cultivation: 5.0-7.0"

**y_pred[0] == 'Soyabean': label="Crop: Soyabean Duration of
cultivation: 45-65" elif y_pred[0] == 'Black Gram':**

**label="Crop: Black Gram Duration of cultivation: 70-85" elif
y_pred[0] == 'Khesari':**

**label="Crop: Khesari Duration of cultivation: 125-130" elif
y_pred[0] == 'Wheat MACS 6222':**

**label="Crop: Wheat MACS 6222 Duration of cultivation: 60-
151"**

elif y_pred[0] == 'Black Pepper':

label="Crop: Black Pepper Duration of cultivation: 120-150"

**elif y_pred[0] == 'Chillies': label="Crop: Chillies Duration of
cultivation: 40-45" elif y_pred[0] == 'Garlic':**

**label="Crop: Garlic Duration of cultivation: 120-150" elif
y_pred[0] == 'Wheat MPO(JW) 1215 (MPO 1215)':**

**label="Crop: Wheat MPO(JW) 1215 (MPO 1215) Duration of
cultivation: 60-150"**

elif y_pred[0] == 'Maize PMH 5 (JH 3110)':

**label="Crop: Maize PMH 5 (JH 3110) Duration of cultivation:
95-105"**

elif y_pred[0] == 'Groundnut Kadiri Harithandhra (K 1319)':

label="Crop: Groundnut Kadiri Harithandhra (K 1319)

elif

Duration of cultivation: 110-121"

elif y_pred[0] == 'Groundnut GPBD 5':

label="Crop: Groundnut GPBD 5 Duration of cultivation: 110-122"

elif y_pred[0] == 'Lentil Pant Lentil-7(Pant L-024)':

label="Crop: Lentil Pant Lentil-7(Pant L-024) Duration of cultivation: 45-61"

elif y_pred[0] == 'Tobacco':

label="Crop: Tobacco Duration of cultivation: 90-120"

elif y_pred[0] == 'Tomato': label="Crop: Tomato Duration

of cultivation: 50-60" elif y_pred[0] == 'Cocoa': label="Crop:

Cocoa Duration of cultivation: 150-180" elif y_pred[0] ==

'Rubber': label="Crop: Rubber Duration of cultivation: 120-

130" elif y_pred[0] == 'Masoor': label="Crop: Masoor

Duration of cultivation: 120-130" elif y_pred[0] ==

'Sunhemp': label="Crop: Sunhemp Duration of cultivation:

60-90" elif y_pred[0] == 'Varagu':

label="Crop: Varagu Duration of cultivation: 160-165"

y_pred[0] == 'Paddy CR Dhan 601 (IET 18558)':

label="Crop: Paddy CR Dhan 601 (IET 18558) Duration of cultivation: 105-125"

elif y_pred[0] == 'Wheat Netravati (NIAW 1415)':

elif

**label="Crop: Wheat Netravati (NIAW 1415) Duration of
cultivation: 95-100"**

elif y_pred[0] == 'Maize HSC1':

label="Crop: Maize HSC1 Duration of cultivation: 95-100" elif

y_pred[0] == 'Millet Nandi-61 (MH-1548)':

**label="Crop: Millet Nandi-61 (MH-1548) Duration of
cultivation: 65-71"**

elif y_pred[0] == 'Millet 86M64 (MSH 203) (Hybrid)':

**label="Crop: Millet 86M64 (MSH 203) (Hybrid) Duration of
cultivation: 65-72"**

elif y_pred[0] == 'Barley Pusa Losar (BH- 380)':

**label="Crop: Barley Pusa Losar (BH- 380) Duration of
cultivation: 60-72"**

elif y_pred[0] == 'Maize DHM 119 (BH 4062)':

**label="Crop: Maize DHM 119 (BH 4062) Duration of
cultivation: 95-103" elif y_pred[0] == 'Millet HHB**

226 (MH 1479)': label="Crop: Millet HHB

226 (MH 1479) Duration of

cultivation: 65-76"

elif y_pred[0] == 'Jute':

label="Crop: Jute Duration of cultivation: 120-150" elif

y_pred[0] == 'Paddy Chinsurah Rice (IET 19140)':

elif

label="Crop: Paddy Chinsurah Rice (IET 19140) Duration of cultivation: 105-120"

elif y_pred[0] == 'Paddy (CNI 383-5-11)':

label="Crop: Paddy (CNI 383-5-11) Duration of cultivation: 105-121"

elif y_pred[0] == 'Paddy IGKVR-1 (IET 19569)':

label="Crop: Paddy IGKVR-1 (IET 19569) Duration of cultivation: 105-122"

elif y_pred[0] == 'Onion':

**label="Crop: Onion Duration of cultivation: 80-150" elif
y_pred[0] == 'Cotton CNH012':**

label="Crop: Cotton CNH012 Duration of cultivation: 80-150"

elif y_pred[0] == 'Cotton CICR-3 (CISA 614)':

label="Crop: Cotton CICR-3 (CISA 614) Duration of cultivation: 105-120"

elif y_pred[0] == 'Rice':

**label="Crop: Rice Duration of cultivation: 105-120" elif
y_pred[0] == 'Flax':**

**label="Crop: Flax Duration of cultivation: 120-140" y_pred[0]
== 'Barley BH-902':**

elif

label="Crop: Barley BH-902 Duration of cultivation: 60-70"

**elif y_pred[0] == 'Sunflower': label="Crop: Sunflower Duration
of cultivation: 90-100" elif y_pred[0] == 'Maize HQPM-4':**

label="Crop: Maize HQPM-4 Duration of cultivation: 95-101"

elif y_pred[0] == 'Groundnut Girnar - 3 (PBS 12160)':

**label="Crop: Groundnut Girnar - 3 (PBS 12160) Duration of
cultivation: 175-180"**

elif y_pred[0] == 'Dry Ginger':

label="Crop: Dry Ginger Duration of cultivation: 175-180"

**elif y_pred[0] == 'Horse Gram': label="Crop: Horse Gram
Duration of cultivation: 120-180" elif y_pred[0] == 'Castor Seed':**

**label="Crop: Castor Seed Duration of cultivation: 7.0-10" elif
y_pred[0] == 'Sesame':**

**label="Crop: Sesame Duration of cultivation: 40-45" elif
y_pred[0] == 'Sugarcane Karan 6 (Co 0239)':**

**label="Crop: Sugarcane Karan 6 (Co 0239) Duration of
cultivation: 300-450" elif**

y_pred[0] == 'Peas':

label="Crop: Peas

Duration of cultivation:

50-100" elif y_pred[0]

== 'Paddy RC

elif

Maniphou 11 (IET

20193)':

**label="Crop: Paddy RC Maniphou 11 (IET 20193) Duration of
cultivation: 105-127"**

elif y_pred[0] == 'Cowpea':

label="Crop: Cowpea Duration of cultivation: 45-90" elif

y_pred[0] == 'Maize MCH 36 (Hybrid) (DKC 9099)':

**label="Crop: Maize MCH 36 (Hybrid) (DKC 9099) Duration of
cultivation: 95-102"**

elif y_pred[0] == 'Pulses':

label="Crop: Pulses Duration of cultivation: 95-102" elif

y_pred[0] == 'Sugarcane Co-0218':

**label="Crop: Sugarcane Co-0218 Duration of cultivation: 300-
452"**

elif y_pred[0] == 'Wheat PDW 314':

label="Crop: Wheat PDW 314 Duration of cultivation: 60-

152" elif y_pred[0] == 'Sweet potato': label="Crop: Sweet potato

Duration of cultivation: 120-125" elif y_pred[0] == 'Cotton VBCH

2231':

**label="Crop: Cotton VBCH 2231 Duration of cultivation: 150-
180"**

elif

return render_template('prediction.html', prediction_text=label)

```
#@app.route('/performance')
```

```
#def performance():
```

```
    return render_template('performance.html')
```

```
if __name__ == "__main__":
```

```
    app.run(debug=True)
```

CONCLUSION

In this paper, significance of management of crops was studied vastly. Farmers need assistance with recent technology to grow their crops. Proper prediction of crops can be informed to agriculturists in time basis. Many Machine Learning techniques have been used to analyze the agriculture parameters. Some of the techniques in different aspects of agriculture are studied by a literature study. Blooming Neural networks, Soft computing techniques plays significant part in providing recommendations. Considering the parameter like production and season, more personalized and relevant recommendations can be given to farmers which makes them to yield good volume of production.

REFERENCES

- [1] Manpreet Kaur, Heena Gulati, Harish Kundra, “Data Mining in Agriculture on Crop Price Prediction: Techniques and Applications”, *International Journal of Computer Applications*, Volume 99– No.12, August 2014.
- [2] J. Meng, “Research on the cost of agricultural products circulation and its control under the new normal economic development,” *Commercial Times*, no. 23, pp. 145147, 2016.
- [3] A. Kaloxylou et al., “Farm management systems and the future Internet era,” *Comput. Electron. Agricult.*, vol. 89, pp. 130–144, Nov. 2012.
- [4] N. N. Li, T. S. Li, Z. S. Yu, Y. Rui, Y. Y. Miao, and Y. S. Li, “Factors influencing farmers’ adoption of new technology based on Logistic-ISM model- a case study of potato planting technology in Dingxi City, Gansu Province,” *Progress in Geography*, vol. 33, no. 4, pp. 542-551, 2014.
- [5] Y. Wang, "A neural network adaptive control based on rapid learning method and its application," *Advances In Modeling and Analysis*, Vol. 46(3), pp. 27-34,1994.