Project Final Deliverable
Report Document

Date	19 November 2022
Team ID	PNT2022TMID48071
Project Name	Project – Fertilizers Recommendation System For Disease Prediction

FERTILIZERS RECOMMENDATION SYSTEM FOR DISEASE PREDICTION

Team ID: PNT2022TMID48071

TEAM LEADER:

Name: S.NAGU

Register Number: 912419104020

TEAM MEMBERS:

Name: S.KARTHIKA

Register Number: 912419104014

Name: P.HARINI BRINDHA

Register Number: 912419104009

Name: M.SNEHA

Register Number: 912419104030

Name: T.VALARMATHI

Register Number:912419104035

FERTILIZERS RECOMMENDATION SYSTEM FOR DISEASE PREDICTION

A PROJECT REPORT

Submitted by,

S.NAGU	912419104020
S. KARTHIKA	912419104014
P.HARINI BRINDHA	912419104009
M.SNEHA	912419104030
T.VALARMATHI	912419104035

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SHANMUGANATHAN ENGINEERING COLLEGE

ARASAMPATTI – 622507

ANNA UNIVERSITY:: CHENNAI 600 025

NOVEMBER 2022

BONAFIDE CERTIFICATE

Certified that this project report "**FERTILIZERS RECOMMENDATION SYSTEM FOR DISEASE PREDICTION**" is the bonafide works of **"S.NAGU(912419104020), S. KARTHIKA (912419104014), P.HARINI BRINDHA (912419104009) M.SNEHA(912419104030) AND T.VALARMATHI (912419104035)"** who carried out the project workunder my supervision.

SIGNATURE

HEAD OF THE DEPARTMENT

Mrs. R. KAVITHA, M.Tech.,
Department of Computer Science and Engineering,
Shanmuganathan EngineeringCollege,
Arasampatti-622507.

SIGNATURE

SUPERVISOR

Mrs. R. KAVITHA, M.Tech.,
Assistant Professor, Department of Computer Science and Engineering,Shanmuganathan Engineering College,
Arasampatti-622507

Submitted for the viva voce to be held on

INTERNAL EXAMINER

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

"HARD WORK NEVER FAILS" So we thank god for having gracefully blessed usto come up till now and thereby giving strength and courage to complete the project successfully. We sincerely submit this project to the almighty lotus feet.

We wish to acknowledge with thanks to the significant contribution given by the management of our college chairman"KALVI VALLAL Mrs.PichappaValliammai", Correspondent Dr.P.Manikandan, and Secretary Mr.Vishvanathan, Shanmuganathan Engineering College,Arasampatti, for their extensive support.

We convey our indebted humble thanks to our energetic Principal Dr. KL. Muthuramu M.E., (W.R), M.E.,(S.E),M.I.S.T.E., F.I.E., Ph.D., for his moral
support to complete this project.

We are grateful to our Head of the Department Mrs. R. Kavitha, M.Tech., for her valuable guidelines, constructive instruction, constant encouragement, unending helps that have been provided to us during the courses of the project.

Gratitude will never fail towards our dynamic and effective internal guide Mrs. R. Kavitha, M.Tech., for his valuable, innovative suggestion and efforts thathe hastaken for the idea of the mini project.

We also convey our humble thanks to all the staff members of the CSE Departmentwho had provide their contribution to steer our project towards the successfulcompletion.

Finally, we thank our family members who have spent their wealth for our studies and have motivated us to complete this project successfully.

FERTILIZERS RECOMMENDATION SYSTEM FOR DISEASE PREDICTION

1.INTRODUCTION

1.1 PROJECT OVERVIEW:

Agriculture is the practice of cultivating plants and livestock. Agriculture was the key development in the rise of sedentary human civilization, whereby farming of domesticated species created food surpluses that enabled people to live in cities. The history of agriculture began thousands of years ago. After gathering wild grains beginning at least 105,000 years ago, nascent farmers began to plant them around 11,500 years ago.Plants were independently cultivated in at least 11 regions of the world. Over one-third of the world's workers are employed in agriculture, second only to the service sector, although in recent decades, the global trend of a decreasing number of agricultural workers continues, especially in developing countries, where smallholding is being overtaken by industrial agriculture and mechanization that brings an enormous crop yield increase.

India largely depends on the agriculture sector. Besides, agriculture is not just a mean of livelihood but a way of living life in India. Moreover, the government is continuously making efforts to develop this sector as the whole nation depends on it for food. For thousands of years, we are practicing agriculture but still, it remained underdeveloped for a long time. Moreover, after independence, we use to import food grains from other countries to fulfill our demand. But, after the green revolution, we become self-sufficient and started exporting our surplus to other countries.Besides, these earlier we use to depend completely on monsoon for the cultivation of food grains but now we have constructed dams, canals, tube-wells, and pump-sets. Also, we now have a better variety of fertilizers, pesticides, and seeds, which help us to grow more food in comparison to what we produce during old times.With the advancement of technology, advanced equipment, better irrigation facility and the specialized knowledge of agriculture started improving.

Agriculture is the important aspect of country development. Many people lead their life from agriculture field, which gives fully related to agricultural products. Plant disease, especially

on leaves, is one of the major factors of reductions in both quality and quantity of the food crops. In agricultural aspects, if the plant is affected by leaf disease then it reduces the growth of the agricultural level. Finding the leaf disease is an important role of agriculture preservation. After pre-processing using a median filter, segmentation is done by Guided Active Contour method and finally, the leaf disease is identified by using Support Vector Machine. The disease-based similarity measure is used for fertilizer recommendation.

Artificial Intelligence is used in many fields to increase the productivity and also performance. It plays a major role in agriculture to increase yield. We may provide large dataset to predict the disease with accuracy and give exact fertilizer for that disease. Most of the farmers are uneducated so we develop the system which is easily accessible by anyone. Detection and recognition of plant diseases using machine learning are very efficient in providing symptoms of identifying diseases at its earliest. Plant pathologists can analyze the digital images using digital image processing for diagnosis of plant diseases. Application of computer vision and image processing strategies simply assist farmers in all of the regions of agriculture. Generally, the plant diseases are caused by the abnormal physiological functionalities of plants. Therefore, the characteristic symptoms are generated based on the differentiation between normal physiological functionalities and abnormal physiological functionalities of the plants. Mostly, the plant leaf diseases are caused by Pathogens which are positioned on the stems of the plants. These different symptoms and diseases of leaves are predicted by different methods in image processing. These different methods include different fundamental processes like segmentation, feature extraction and classification and so on. Mostly, the prediction and diagnosis of leaf diseases are depending on the segmentation such as segmenting the healthy tissues from diseased tissues of leaves.

1.2 PURPOSE:

Fertilizers are generally defined as "any material, organic or inorganic, natural or synthetic, which supplies one or more of the chemical elements required for the plant growth." Most fertilizers that are commonly used in agriculture contain the three basic plant nutrients: nitrogen, phosphorus, and potassium. Some fertilizers also contain certain "micronutrients," such as zinc and other metals, that are necessary for plant growth. Fertilizers are applied to replace the essential

nutrients for plant growth to the soil after they have been depleted.Excess amounts of fertilizers may enter streams creating sources of nonpoint pollution. Fertilizers most commonly enter water sources by surface runoff and leaching from agricultural lands. Large amounts of nitrogen and phosphorous are present in the runoff. Increased amounts of nitrogen, phosphorous, and other micronutrients can have negative impacts on public health and aquatic ecosystems.

To grow healthy crops full of nutrients, farmers need to ensure they have healthy plant. Without fertilizers, nature struggles to replenish the nutrients in the plant.When crops are harvested, important nutrients are removed from the soil, because they follow the crop and end up at the dinner table.Often, the plants have few possibilities to avoid nutrient deficiencies without the help of fertilizers. In addition they are additional substances supplied to the crops to increase their productivity. These are used by the farmers daily to increase the crop yield. These fertilisers contain essential nutrients required by the plants, including nitrogen, potassium, and phosphorus. They also enhance the water retention capacity of the soil and increase its fertility.

In a modern days, the former uses a various techniques for identifing the disease in the plants. By providing the application to predict the Disease they can easly rectify the problem arises in the plant , the application is not only used for the the former but also the people who have a garden in their house can make use of this app.

The Application turns your mobile phone into a mobile crop doctor with which you can accurately detect pests and diseases on crops within seconds. It serves as a complete solution for crop production and management.Visual inspection plays an important role in plant disease identification and diagnosing plant problems, as a lab test is not a practical tool for day-to-day diagnosis. Due to costs and turnover time, growers make decisions based on the symptoms seen on plants.. An artificial intelligence-based app that identifies plant diseases became possible thanks to the big leap in performance engineered by the artificial intelligence research community.

In recent years, software products can complete vision-based tasks with expert accuracy. Why not apply this technology to plant disease identification and agronomy advisory then? The widespread

use of mobile devices makes this question even more relevant as such devices made it possible to distribute a solution on a large scale and help growers diagnose plant problems.

It is an Web-based application developed to do just that. It helps growers to manage plant protection more sustainably as disease identification is a crucial component in the plant protection routine.The app is valuable to large and diverse audiences. Aside from farmers and crop advisors, home growers seeking professional advice are a large part of the user base. They leverage the disease identification feature for their needs. This is aligned with a growing trend of millennials that use technology to help grow food in their homes. In addition, The Application suggestes biological and organic treatments as part of the integrated pest management protocols, making it even more relevant for gardeners and home growers.

Every pathogen or insect will require a different treatment. In addition, pathogens can develop resistance to some treatments in some geographics; this information should be considered when treatment is advised.When the problems are already observed in the plot, exact identification can help prevent spread. The strategy on which measures should be applied depends on the precise identification of the cause.When it comes to prevention, growers should consider the implications for the following seasons. Crop rotation or soil treatment might be advised when the pest is expected to survive in the soil or plant debris. In the case of a viral disease, it might be advised to plant virus-resistant varieties in the following seasons once the virus has been identified in the region. Other factors that require soil preparation before planting are saline soil, wrong soil pH, presence of nematodes, and more. In all the examples above, exact identification of the problem must be obtained to prevent future losses.

State-of-the-art image recognition is based on the concept of artificial neural networks. Similar to how agronomy students learn, the neural network is presented with examples of diseased plants that were tagged by experts. In the learning process, the network of neurons adapts until it maximizes the performance score. The result is an app that identifies plant diseases and constantly improves as more examples are presented. Based on the growers' feedback and observations done in the fields, Agrio learns which treatment protocols are more effective.

2. LITERATURE SURVEY

2.1 Existing problem:

[1]The disease and suggest the fertilizer. The proposed method is compared with the existing CNN based leaf disease prediction. The proposed SVM technique gives a better result when compared to existing CNN.

Advantages:

- The prediction and diagnosing of leaf diseases are depending on the segmentation such as segmenting the healthy tissues from diseased tissues of leaves.

Disadvantages:

- This further research is implementing the proposed algorithm with the existing public datasets. Also, various segmentation algorithms can be implemented to improve accuracy.
- The proposed algorithm can be modified further to identify the disease that affects the various plant organs such as stems and fruit.

[2] Detection of Leaf Diseases and Classification using Digital Image Processing International Conference on Innovations in Information, Embedded and Communication Systems(ICIIECS), IEEE, 2017.

Advantages:

- The system detects the diseases on citrus leaves with 90% accuracy

Disadvantages:

- System only able to detect the disease from citrus leaves.
- The main objective of this paper is image analysis & classification techniques for detection of leaf diseases and classification.
- The leaf image is firstly preprocessed and then does the further work. K-Means Clustering used for image segmentation and then system extract the GLCM features from disease detected images.
- The disease classification done through the SVM classifier.

[3] Semi-automatic leaf disease detection and classification system for soybean culture IET Image Processing, 2018.

Advantages:

- The system helps to compute the disease severity

Disadvantages:

- The system uses leaf images taken from an online dataset, so cannot implement in real time.
- This paper mainly focuses on the detecting and classifying the leaf disease of soybean plant. Using SVM the proposed system classifies the leaf disease in 3 classes like i.e. downy mildew, frog eye, and septoria leaf blight etc.
- The proposed system gives maximum average classification accuracy reported is ~90% using a big dataset of 4775 images.

[4] Cloud Based Automated Irrigation And Plant Leaf Disease Detection System Using An Android Application. International Conference on Electronics, Communication and Aerospace Technology, ICECA 2017.

Advantages:

- It is simple and cost effective system for plant leaf disease

Disadvantages:

- Any H/w failures may affect the system performance.
- The current paper propose an android application for irrigation and plant leaf disease detection with cloud and IoT.
- For monitoring irrigation system they use soil moisture and temperature sensor and sensor data send to the cloud.
- The user can also detect the plant leaf disease.

[5] The author proposes a method which helps us predict crop yield by suggesting the best crops. It also focuses on soil types in order to identify which crop should be planted in the field to increase

productivity. In terms of crop yield, soil types are vital. By incorporating the weather details of the previous year into the equation, soil information can be obtained.

Advantages:

- It allows us to predict which crops would be appropriate for a given climate. Using the weather and disease related data sets, the crop quality can also be improved.
- Pre-diction algorithms help us to classify the data based on the disease, and data extracted from the classifier is used to predict soil and crop.

Disadvantages:

- Due to the changing climatic conditions, accurate results cannot be predicted by this system.

[6] The current work examines and describes image processing strategies for identifying plant diseases in numerous plant species. BPNN, SVM, K-means clustering, and SGDM are the most common approaches used to identify plant diseases.

Disadvantages :

- Some of the issues in these approaches include the impact of background data on the final picture, optimization of the methodology for a specific plant leaf disease, and automation of the technique for continuous automated monitoring of plant leaf diseases in real-world field circumstances.

2.2 References:

[1] Semi-automatic leaf disease detection and classification system for soybean culture IET Image Processing, 2018

[2] Cloud Based Automated Irrigation And Plant Leaf Disease Detection System Using An Android Application. International Conference on Electronics, Communication and Aerospace Technology, ICECA 2017.

[3] Ms. Kiran R. Gavhale, Ujwalla Gawande, Plant Leaves Disease detection using Image Processing Techniques, January 2014
https://www.researchgate.net/profile/UjwallaGawande/publication/314436486_An_Overview_of_the_Research_on_Plant_Leaves_Disease_detection_using_Image_Processing_Techniques/links/5d3710664585153e591a3d20/An

[4] Duan Yan-e, Design of Intelligent Agriculture Management Information System Based on IOT‖, IEEE,4th, Fourth International reference on Intelligent Computation Technology and Automation, 2011
https://ieeexplore.ieee.org/document/5750779

[5] R. Neela, P. Fertilizers Recommendation System For Disease Prediction In Tree Leave International journal of scientific & technology research volume 8, issue 11, november 2019
http://www.ijstr.org/final-print/nov2019/Fertilizers-Recommendation- System-For-Disease-Prediction In-Tree-Leave.pdf.

[6] Swapnil Jori1, Rutuja Bhalshankar2, Dipali Dhamale3, Sulochana Sonkamble , Healthy Farm: Leaf Disease Estimation and Fertilizer Recommendation System using Machine Learning,International Journal of All Research Education and Scientific Methods (IJARESM), ISSN: 2455-6211

[7] Detection of Leaf Diseases and Classification using Digital Image Processing International Conference on Innovations in Information, Embedded and Communication Systems(ICIIECS), IEEE, 2017.

[8] Shloka Gupta ,Nishit Jain ,Akshay Chopade, Farmer's Assistant: A Machine Learning BasedApplication for Agricultural Solutions.

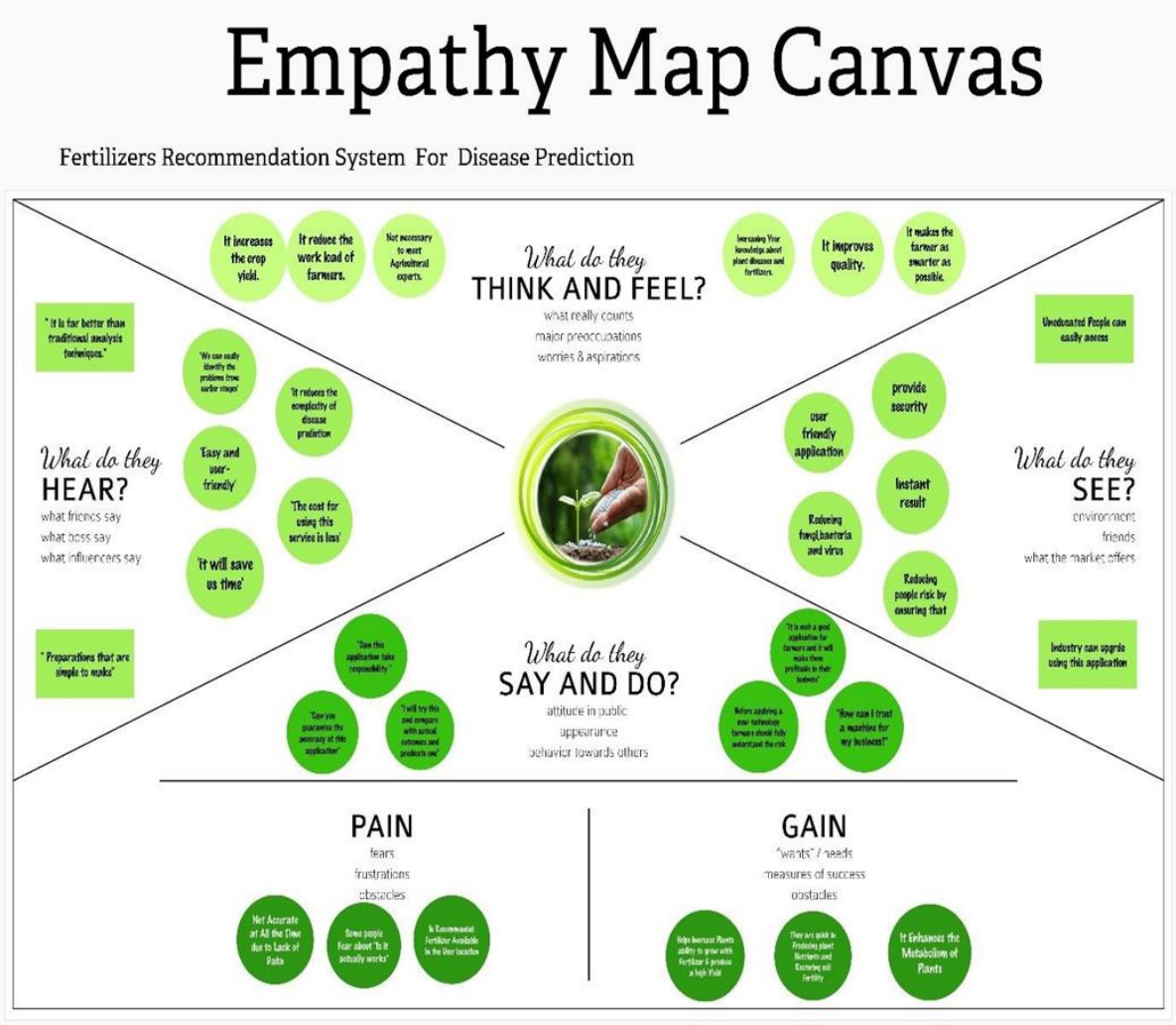
2.3 Problem Statement Definition:

- In agricultural aspects, if the plant is affected by leaf disease, then it reduces the growth and productiveness.
- The plant diseases are caused by the abnormal physiological functionalities of plants.
- The issue occurs in agriculture practicing areas, particularly in rural regions.
- It is required for the growth of better quality food products. It is important to maximise the crop yield.
- An automated system is introduced to identify different diseases on plants by checking the symptoms shown on the leaves of the plant.

3.IDEATION AND PROPOSED SOLUTION

3.1 Empathy Map Canvas:

- An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviour and attitudes.
- It is a useful tool to helps teams better understand their users.
- Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.



3.1.1 Empathy Map Diagram

What you they think and feel?

- With the help of this application farmers can improve their productivity and quality of the crop.
- Suppose new disease will come, there is no need to meet the agriculture experts to cure this.
- By using this farmers fell smarter.
- Frequently using this application we improve knowledge about plant diseases and fertilizers.
- Reducing the workload of the farmers.

What you they see?

- Even though uneducated also easily access this application because of its simplicity.
- Promoting the healthy lifestyle for the farmers.
- People with no prior knowledge can access.
- It will give the solution as quick as possible so lost of time is saved for some other process.

What you they hear?

- This is more helpful in identifying the disease in the crop and makes us more profit.
- Fix our problems from early stages with this application.
- Making revolutionary changes in farming industry.
- We get a clear report that gives us a better understanding of the problem.
- If it makes a wrong prediction, it leads to a huge loss.
- If reduces the complexity of disease prediction.
- It is far better than traditional analysis techniques.

What you they say and do?

- Can you guarantee the accuracy of this application?
- Can this application take responsibility for the losses that may happen due to this application?

- Before applying a new technology farmers should fully understand the risk.
- It is such a good application for farmers and it will make them profitable in their business.

Pain:

- Not accurate at all the time due to lack of data.
- Some people fear about is it actually works.
- Is recommended fertilizer available in the user location.

Gain:

- One of the most efficient and rapid methods of disease detection.
- Early detection and management of problems.
- Constantly of constant work.
- It enhances the metabolism of plants.

3.2 Ideation & Brainstorming:

- Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

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

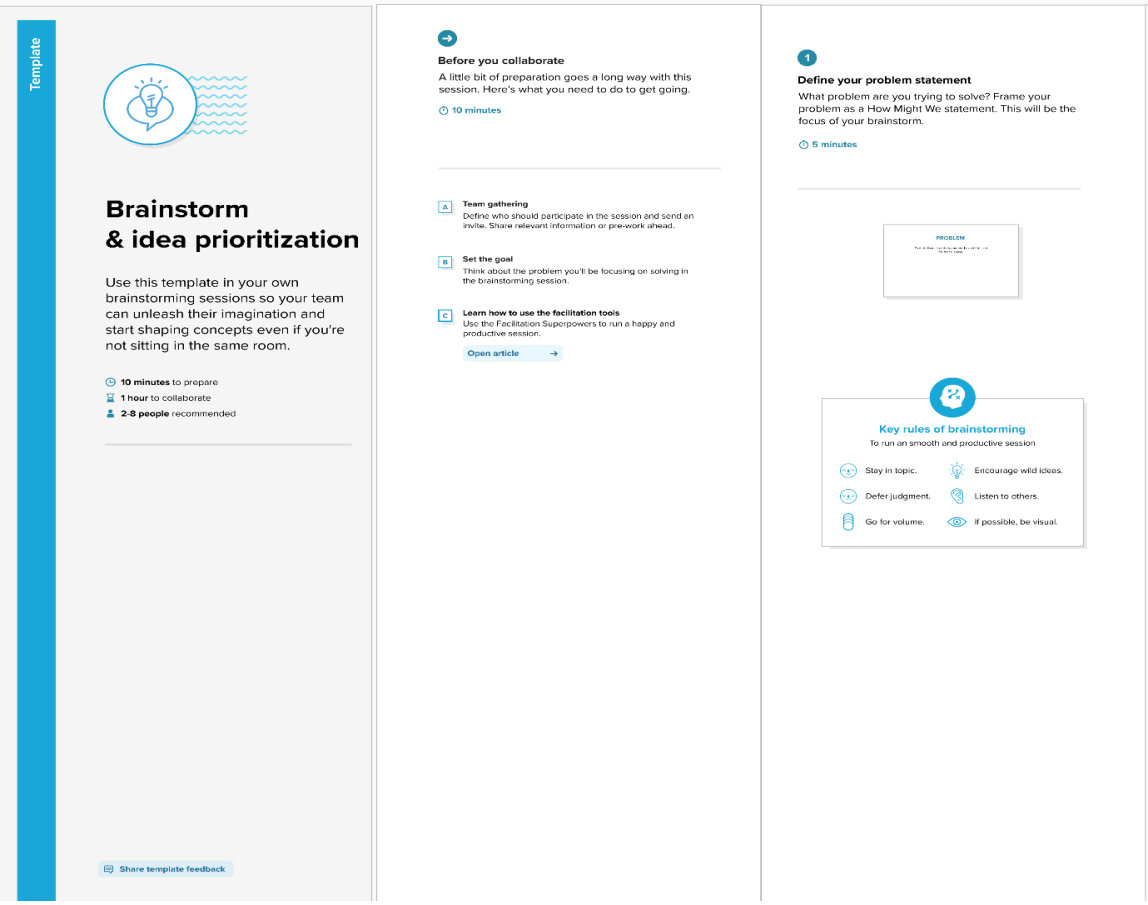
Steps:

Step-1: Team Gathering, Collaboration and Select the problem statement

Step-2: Brainstorm, Idea Listing and Grouping

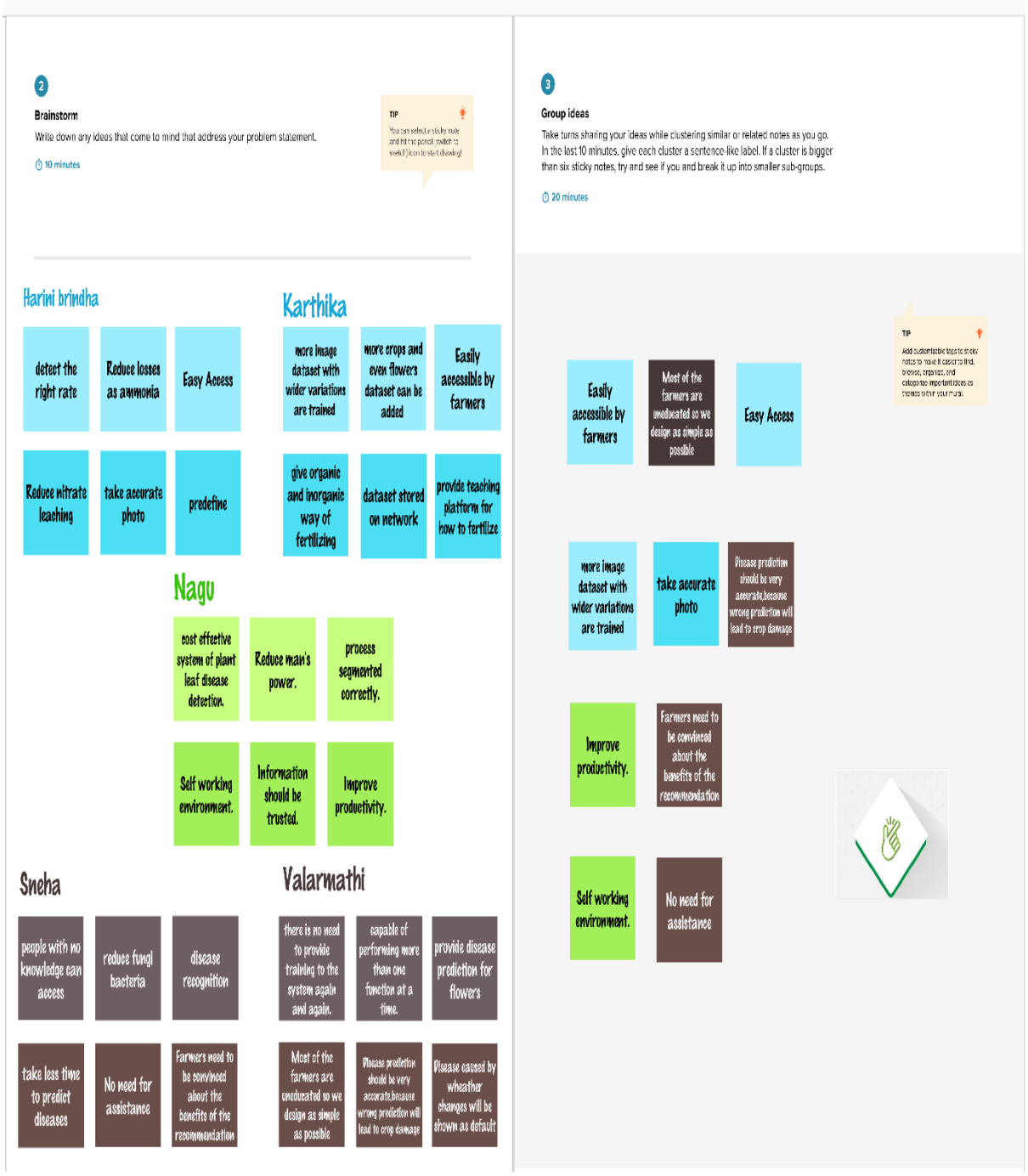
Step-3: Idea Prioritization

Step-1: Team Gathering, Collaboration and Select the Problem Statement



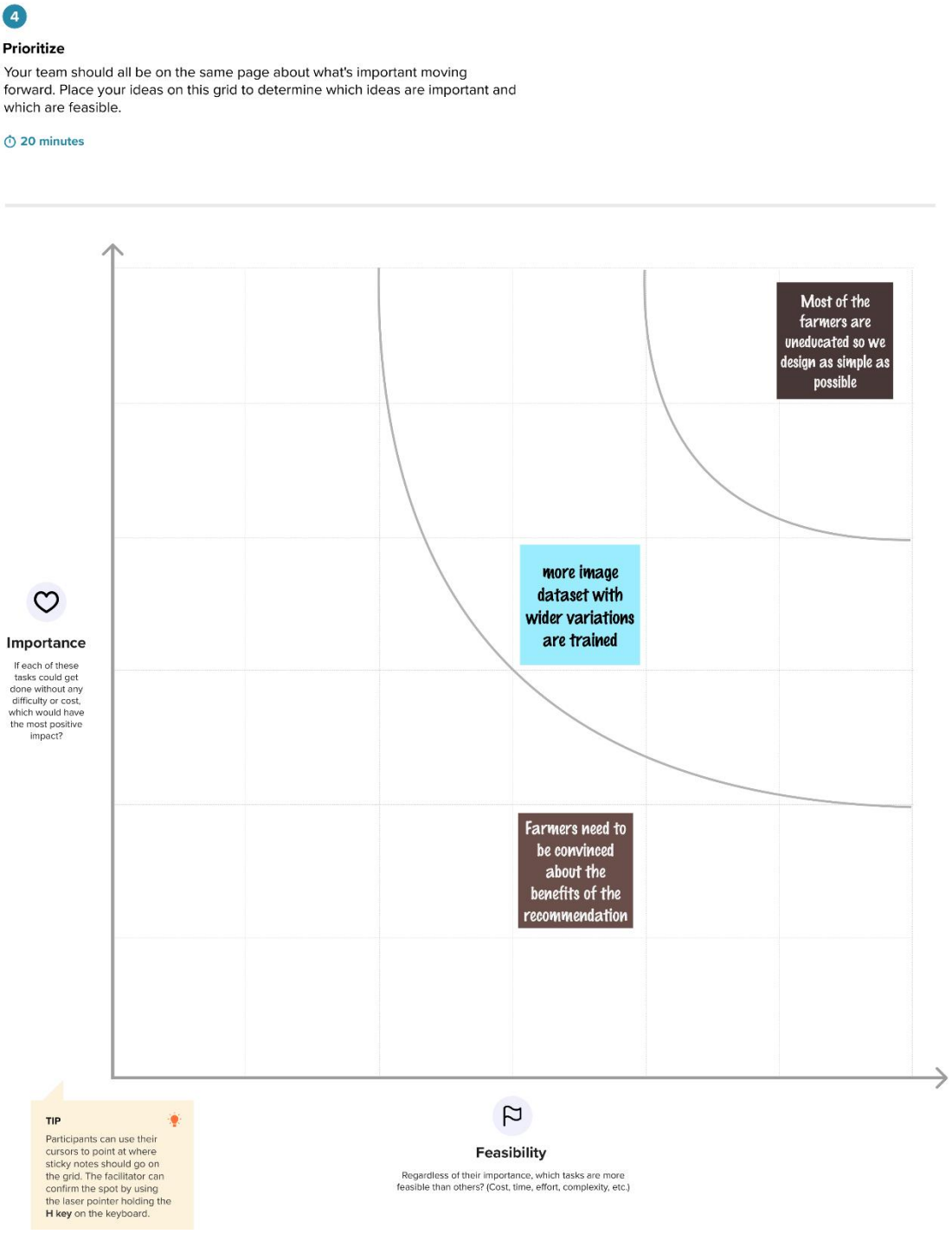
3.2.1 Team Gathering, Collaboration and Select the Problem Statement Diagram

Step-2: Brainstorm, Idea Listing and Grouping



3.2.2 Brainstorm, Idea Listing and Grouping Diagram

Step-3: Idea Prioritization



3.2.3 Idea Prioritization Diagram

3.3 Proposed Solution:

Problem Statement:

To find disease and recommend exact fertilizerfor that disease

Idea / Solution description:

- Plant disease reduces the production andquality of food, fiber and biofuel crops.
- It has been a major factors that influencingthe farmers life as well as our life.
- To overcome this problem we develop this project to predict the plant disease and recommend the fertilizer.

Novelty / Uniqueness:

- To predict the accurate disease for plant andcrops we add more image dataset with widervariations are trained.
- Most of the farmers are uneducated so we develop the system which is easily accessible by
- anyone.

Social Impact / Customer Satisfaction:

- It helpful for farmers to increase productivity.
- Increase the usability of nature manure.

Business Model (Revenue Model):

- As long as this system is beneficial to users, subscriptions will increase which gives benefits to industry.

Scalability of the Solution:

Useful for those who don't know the basic about cultivation.

3.4 Problem Solution fit

Customer statement:

Farmers who get low yieldbecause of fungal and bacterial diseases in plant.

Problems and pains:

Recommended fertilizer is not available in the customerlocation.

Triggers:

Seeing their neighbour installing the application, there is no amount to pay for this services.

Emotions:

Most farmers depends on these solution to effectively control different types of pests while using the application they fell smart.

Available solutions:

Existing system sometimes could not find the disease accurately because of lack of dataset, so we provide large dataset for the disease predictions.

Customer limitations:

It requires some basic level of technical skill, infrastructure to use this applications.

Behaviour:

By identifying the right solution ityield the better crop production.

Channels of behaviour:

Online: It is necessary that the customer willbe online for disease prediction.

Offline: Farmers take the photos of diseaseaffected plants in offline mode.

Problem root/cause:

Exploitation by middlemen by purchasing crops at very low pricefrom farmers.

Your solution:

By the developing the applications weprovide valuable service to the farmers as well as common people.

3.4.1 Problem Solution Fit Diagram

4. REQUIREMENT ANALYSIS

4.1 Functional Requirement:

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. So, it's important to make them clear both for the development team and the stakeholders. Generally, functional requirements describe system behavior under specific conditions

Functional requirement of the project is,

- User Registration
- User Confirmation
- User Work
- Prediction Result
- Final part.

User Registration:

As a user, I can register for the application by entering my email as a user, password, and confirming my password. Once A user register for the application No need to repeat the process. Directly log into the application by entering given email and password.

User registration is an important prerequisite for the success of many websites by enabling users to gain access to domain information and personalized content. It is not always desirable for users, however, because they need to disclose personal information.

User Confirmation:

Confirmation emails are transactional emails which verify that an action taken by a customer has been successful. These automated emails give the customer peace of mind, assuring them that what they wanted to happen has been achieved and acknowledged by your company.

Registration confirmation emails are sent to users after they sign up to your email and complete a registration process on a website. Confirmation emails are Ready to access the application on the given user.

- They verify that an action taken by a customer has been successful

- They act as a receipt and include all relevant information.

- They act as a touchpoint between your website and customers.

User Work:

Collection of more images about the plants to upload from the gallery. It is used to predict the plants affected by disease or not. Once the user click the predict button ,then the uploaded image match the model to identify the disease. If the image does not matched repeat the process. Most of the farmers are uneducated so we develop the system which is easily accessible by anyone.

The traditional method that some farmers use has no modern techniques to automate plant disease detection and classification. This leads to a huge reduction in the quantity and quality of agricultural production, because, farmers fail to detect plant diseases in large farms.

Prediction Result:

Most plants are affected by a wide variety of bacterial and fungal diseases. Diseases on plants placed a major constraint on the production and a major threat to food security.

User can upload the images, then the process will be started to predict the plant disease. If the disease are identified, that name of disease will be displayed. After Fertilizer for that disease.

By using this application easily identify the disease and predict, then recommend exact fertilizer for that disease. So this application very useful to the increasing yield. Hence, early and accurate identification of plant diseases is essential to ensure quantity and best quality.

Final Part

As a result user satisfied with the process of disease prediction and provide the fertilizer about the disease. User can fulfill with the process they can move for logout.

4.2 Non-Functional Requirements:

Non-functional requirements are a set of specifications that describe the system's operation capabilities and constraints and attempt to improve its functionality. Non functional requirements, also called Quality Attributes . It specify system characteristics that are required for acceptance of the system by the end user. It is therefore imperative that your document such requirements in the functional specification.

It Requires defining and implementing non functional requirements will ensure that the system runs smoothly and features quality attributes that are necessary for meeting the needs of the end-user.

Non-Functional Requirements of the project is,

- Usability
- Security
- Reliability
- Performance
- Availability
- Scalability.

Usability:

Usability is a quality attribute that assesses how easy user interfaces are to use. In this method for improving easy to use during the process.

The quality of a user's experience when interacting with system. The end user's ability to learn the system and complete tasks. Dataset of all the leaf is used to detecting the disease that present in the leaf.

Usability have a five quality components, Learnability, Efficiency, Memorability, Errors, Satisfaction.

Learnability : How easy is it for users to accomplish basic tasks. In this project Uneducated farmers also useful to the application.

Efficiency : The ability to achieve an end goal with little to no waste, effort and Energy.
Memorability: In this process very easily accessible by anyone. So they access the application with the basic knowledge.
Errors: It can effortlessly identify the errors in the application and rectified it quick As possible.
Satisfaction: We finally found a solution that will satisfied everyone needs about plant disease prediction.

Security:

Protect software against malicious attack and other hacker risks so that the software continues to function correctly. The information belongs to the user and leaf are secured highly. The ability to prevent and forbid restricted actions. security is the identification of potential risks and implementation of strategies which will protect or preserve the confidentiality integrity and availability of project resources.

- Each page in the system shall authenticate the user.
- User that are not allowed to view specific areas(any viruses) of the systems will be redirected to the login page

Reliability:

The extent to which the system will execute without failure for a specific period of time. In the application using less time to identify the disease then exact fertilizer about that disease. So it is very useful for farmers.

Failure-free detection of an application for plant disease. The leaf quality is important for the predicting the disease in leaf. And also without any overlapping the application can easily process the image which is uploaded and go for prediction.

Performance:

Specifying and analyzing quantitative behavior from the very beginning of a system, through to its deployment and evolution.

Farmers who get low yield because of fungal and bacterial diseases in plant. So it is such a application to detect the problem at plant by identifying the correct disease. The performance is based on the quality of the leaf used for disease prediction.

The systems ability to meet latency, throughput and resource utilization requirements.

- All web pages must download within few seconds .
- While executing a search, the system must be able to display exact disease about the plant.

Business performance of the project ,As long as this system is beneficial to users, subscriptions will increase which gives benefits to industry.

Availability:

The system is operating properly when it is requested for use. It is available for all user to predict the disease in the plant.

By the developing the application we provide valuable service to the farmers as well as common peoples.

Scalability :

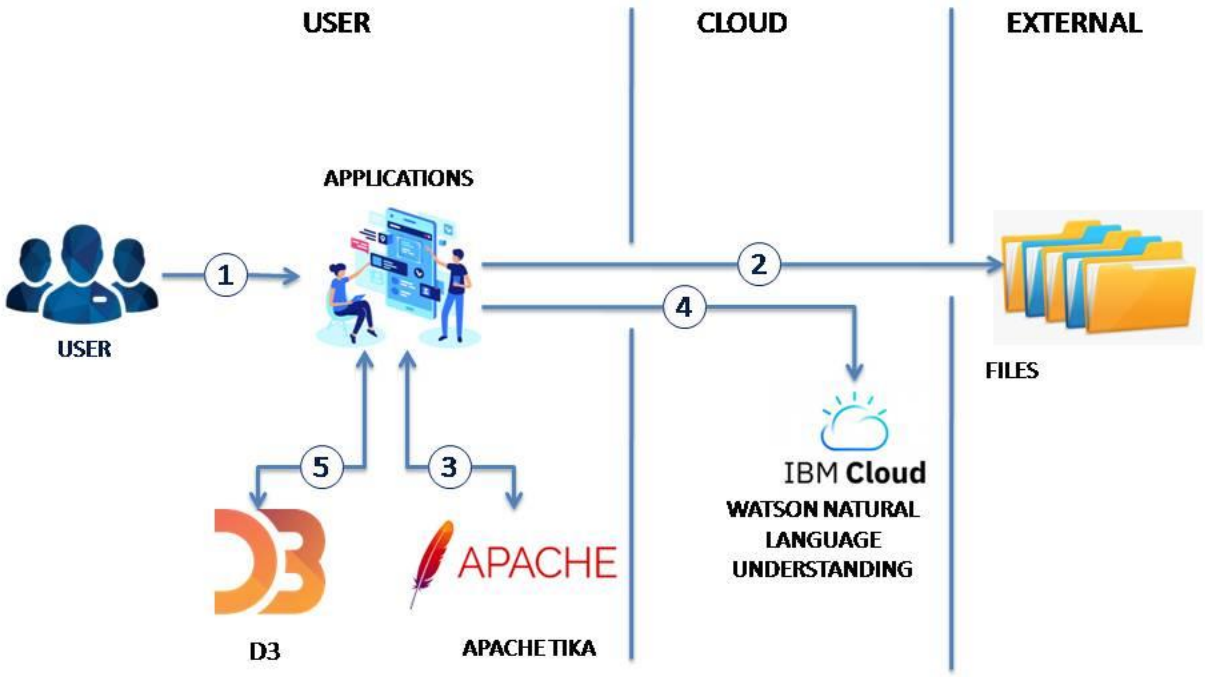
The measure of a system's ability to increase or decrease in performed and cost in response to changes in application and system processing demands.

The system's ability to handle a large amount of data. In this project add large amount of dataset with wider variations are trained. Increase the prediction of the disease in the leaf. Increase the prediction of the disease in the leaf. Useful for those who don't know the basic about cultivation.

5.PROJECT DEISGN:

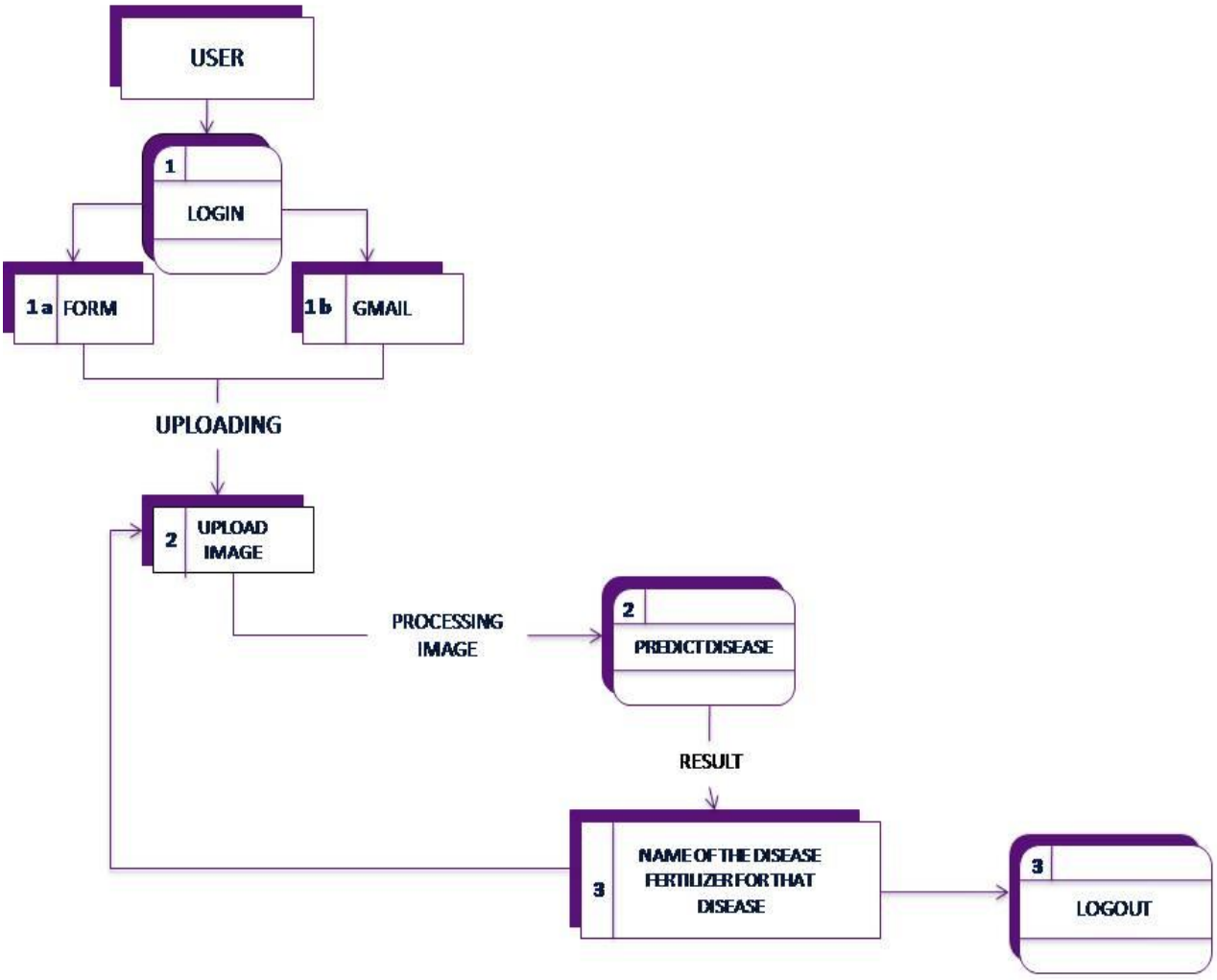
5.1DATA FLOW DIAGRAM:

A data flow diagram shows the way information flows through a process or system. It includes data inputs and outputs, data stores, and the various sub-processes the data moves through. DFD's are built using standardized symbols and notation to describe various entities and their relationships.It is visually represent systems and processes that would be hard to describe in just words. You can use these diagrams to map out an existing system and make it better or to plan out a new system for implementation. Visualizing each element makes it easy to identify inefficiencies and produce the best possible system.



5.1.1 Data Flow Diagram

From the above Data Flow Diagram, it shows the user can access the Application by the use of files stored as predifined with the help of APACHE they can communicate over network from user and application, and the IBM Cloud watson is used to upload the Dataset in it and asign the name for the uploaded Dataset.



5.1.2 Process Data Flow Diagram

USER:

In the Data Flow Diagram, the user can Login the Application either by the form or Gmail, by entering the mail address and password of the user once it will successed no need to repeat the process for the login.In the Way of the User can directly access the Application by entering mail and Password.

UPLOAD IMAGE:

After the Confirmation the User can go for Uploading the Picture from the album for process of Image Processing.Image processing is a method to perform some operations on an image, in order to get an enhanced image or to extract some useful information from it.It is a type of signal processing in which input is an image and output may be image or characteristics/features associated with that image.

PREDICT DISEASE:

By the Process of uploading the picture,the User can go for Prediction by clicking the Predict Button, then the uploaded Picture will Compared with the Predifined Dataset that is loaded in the Application, if the Uploaded Image doesn’t matches the User can repeat the Process.

NAME OF THE DUSEASE & FERTILIZER FOR DISEASE:

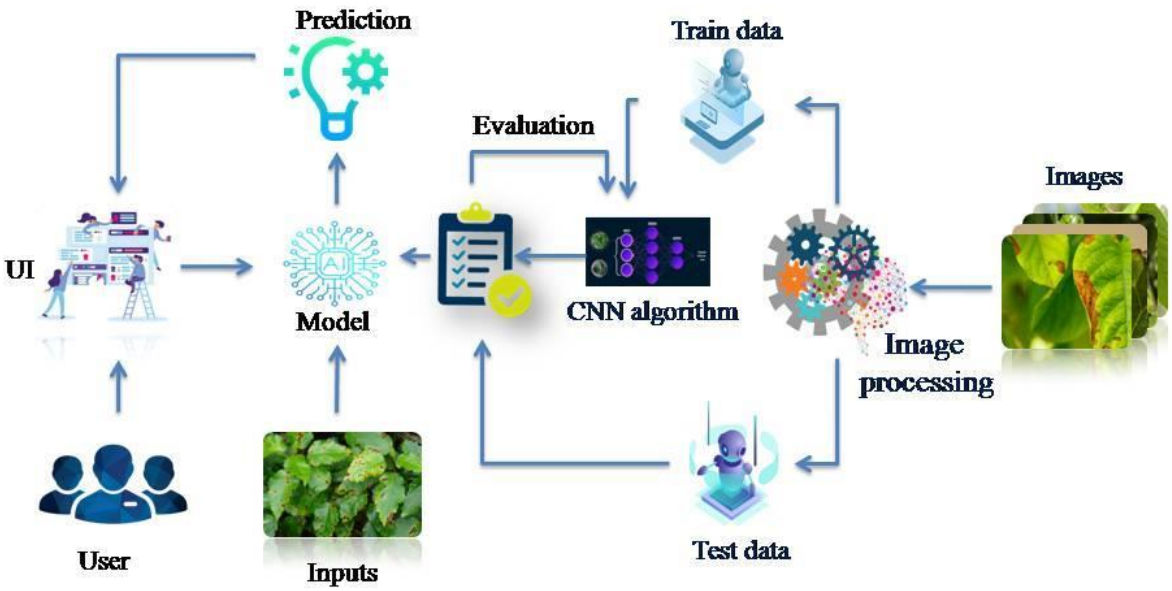
After the Prediction process in go for Displaying the type of Disease identified and the Fertilizer for that particular Disease. As a Result, by the Effective use of the Application the User can Effortlessly identifing the Disease of the Plant and Recommend the Fertilizer for that accordingly.

LOGOUT:

By the complition of the process of Predicting the Disease in the Plant and Provide the Fertilizer if the User is Satisfied with this Process they can go for logout for the Application by Stopping being connected to it.

5.2 SOLUTION AND TECHNICAL ARCHITECTURE:

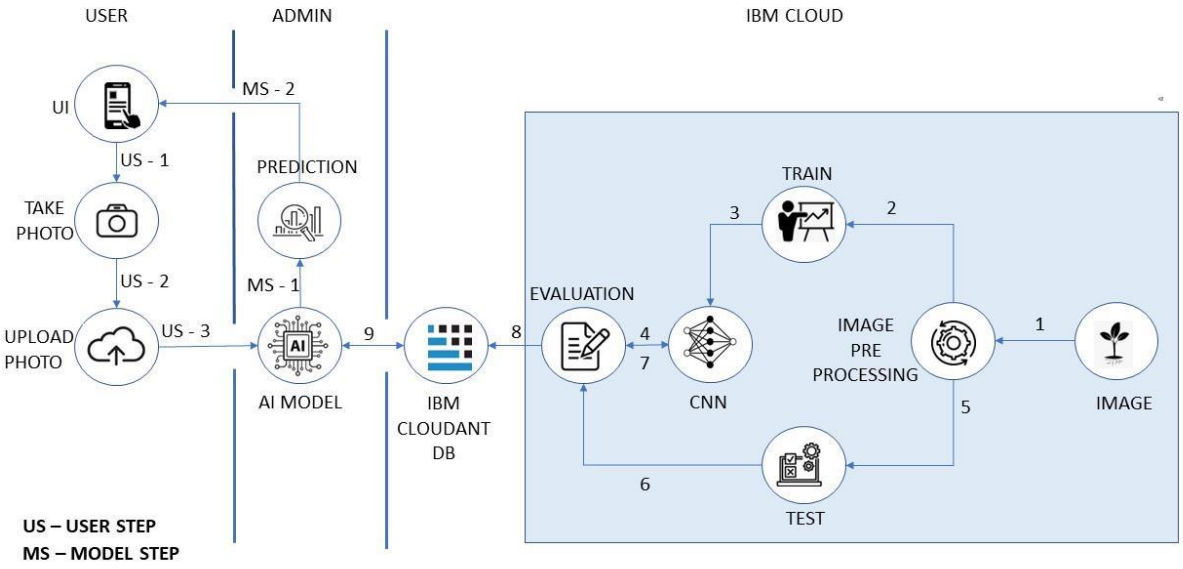
SOLUTION ARCHITECTURE:



5.2.1 Solution Architecture Diagram

From the above solution Architecture, it shows the uploaded image in the Application via gallery ,the image is go for the futher process of image Processing and its undergoes the Process of test and train the data.In machine learning, datasets are split into two subsets. The first subset is known as the training data, it's a portion of our actual dataset that is fed into the machine learning model to discover and learn patterns.another is Training the data, we want to feed the model with as much data as possible to find and learn meaningful patterns. Once data from our datasets are fed to a machine learning algorithm, it learns patterns from the data and makes decisions.and we make use of CNN Algorithm ,it is a kind of network architecture for deep learning algorithms and is specifically used for image recognition and tasks that involve the processing of pixel data.after that the model go for prediction with the given input.

TECHNICAL ARCHITECTURE:



5.2.2 Technical Architecture Diagram

Overhead of the architecture, it is divided into three major groups,

1. USER
2. ADMIN
3. IBM CLOUD

USER:

- As a User they can Log on to the Application by Registering with the help of either by the mail id and password or by the form .by this process only the authorised user can access the Appplication.
- The User has to take the picture of the plant for the further process.
- Once the user log on to the Application,they can go for Upload the photo of the Plant in the Application.

ADMIN:

- As a Admin ,We Predefine the train dataset and load the model .
- The Admin also Predifine the correct Prediction Result of the Plant either if it is Affected by the diseases or not and also We Recommend the Fertilizer for the given Diseases.

IBM CLOUD:

- In this portion shows that the uploaded image undergoes the Process of image Processing, It is the process of transforming an image into a digital form and performing certain operations to get some useful information from it. The image processing system usually treats all images as 2D signals when applying certain predetermined signal processing methods.
- The *training data is the biggest (in -size) subset of the original dataset, which is used to train or fit the machine learning model*. Firstly, the training data is fed to the ML algorithms, which lets them learn how to make predictions for the given task.Once we train the model with the training dataset, it's time to test the model with the test dataset. This dataset evaluates the performance of the model and ensures that the model can generalize well with the new or unseen dataset. *The test dataset is another subset of original data, which is independent of the training dataset.*
- A CNN is a kind of network architecture for deep learning algorithms and is specifically used for image recognition and tasks that involve the processing of pixel data. By using the CNN algorithm we use many layer to reduce the Noise in the Uploaded picture of the Plant.
- After the completion of before Processes it Evaluate the input with the Predefined dataset and Provide the Result as like the given image of the Plant is Affected by the Disease or not.
- Overall the IBM cloud is used to store the Developed of the Application and the given information are used further.

5.3 USER STORIES:

USN-1:

User task: As a user I can register for the application by entering my email as a user, password, and confirming my password.
Acceptance criteria: I can access my profile / dashboard.
Priority: High
Release: Sprint 1

USN-2:

User task: As a user, I will receive confirmation email once I have registered for the application.
Acceptance criteria: I can receive confirmation email & click confirm.
Priority: High
Release: Sprint 1

USN-3:

User task: As a user, I can register for the application through Facebook.
Acceptance criteria: I can register & access the dashboard with Facebook Login.
Priority: Low
Release: Sprint 2

USN-4:

User task: As a user, I can register for the application through Gmail.
Acceptance criteria: I can Access the Application via Gmail.
Priority: Medium
Release: Sprint 1

USN-5:

User task: As a user, I can log into the application by entering email & password.

Acceptance criteria: I can use the Application for Disease Prediction.

Priority: High

Release: Sprint 1

USN-6:

User task: As a Web user, I can register with a User ID on the System.

Acceptance criteria: I can able to access the app as a website

Priority: High

Release: Sprint 1

USN-7:

User task: As a Supporter, I can Understand exactly how customer use the product.

Acceptance criteria: I can create Guidelines and Practices for Customer

Priority: Low

Release: Sprint 2

USN-8:

User task: As a Admin, I can Update many dataset about the Plant Diseases.

Acceptance criteria: I can able to store large Amount of Data

Priority: High

Release: Sprint 1

USN-9:

User task: It uses AI to identify the Plants Disease within the Captured photos and Live View of Prediction.

Acceptance criteria: I can Predict Disease of the Plant

Priority: High

Release: Sprint 1

6.PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation:

Sprint_1:

- Data collection - In this we collect all the required datasets for train and test the models.
- Image processing - Image processing library mainly focused on real-time computer vision with application in wide-range of areas like 2D and 3D feature toolkits, facial & gesture recognition, Human-computer interaction, Mobile robotics, Object identification and others.

Duration: 24 oct 2022-29 oct 2022(6 days)

Sprint_2:

- Model building – In this we built model for both fruit disease prediction and vegetable disease prediction.
- Test both the model – test the two models that are build in the above process.

Duration:31 oct 2022-05 nov 2022(6 days)

Sprint_3:

- Application building – An application building describes the patterns and techniques used to design and build an application. The architecture gives you a roadmap and best practices to follow when building an application, so that you end up with a well-structured app. we use HTML and python code for build this application building.

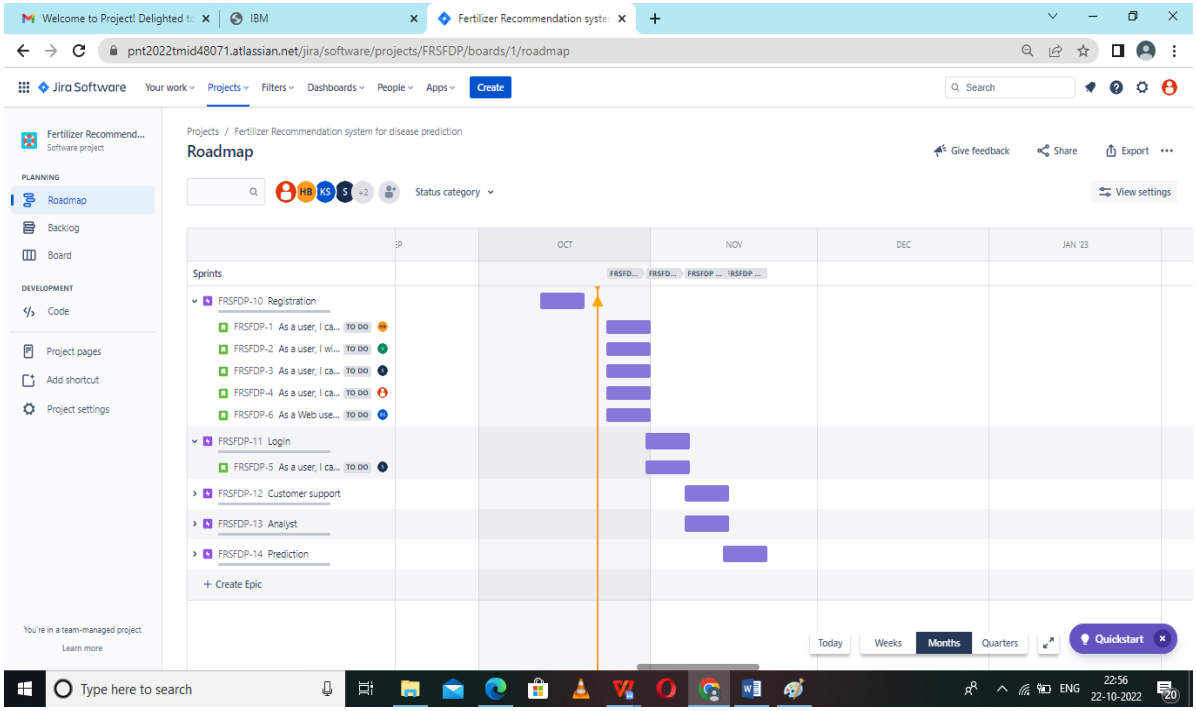
Duration:07 nov 2022-12 nov 2022(6 days)

Sprint_4:

- Train the model on IbM – This includes the register for IBM cloud and train model on IBM.

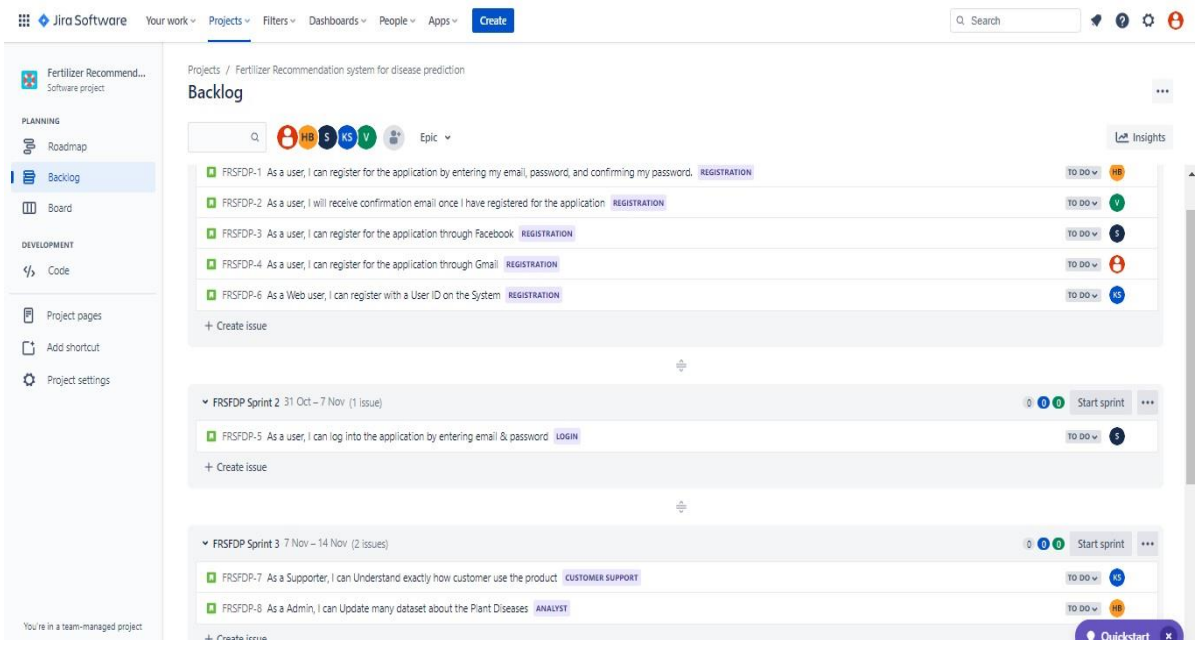
Duration:14 nov 2022-19 nov 2022(6 days)

6.2 Sprint Delivery Schedule:

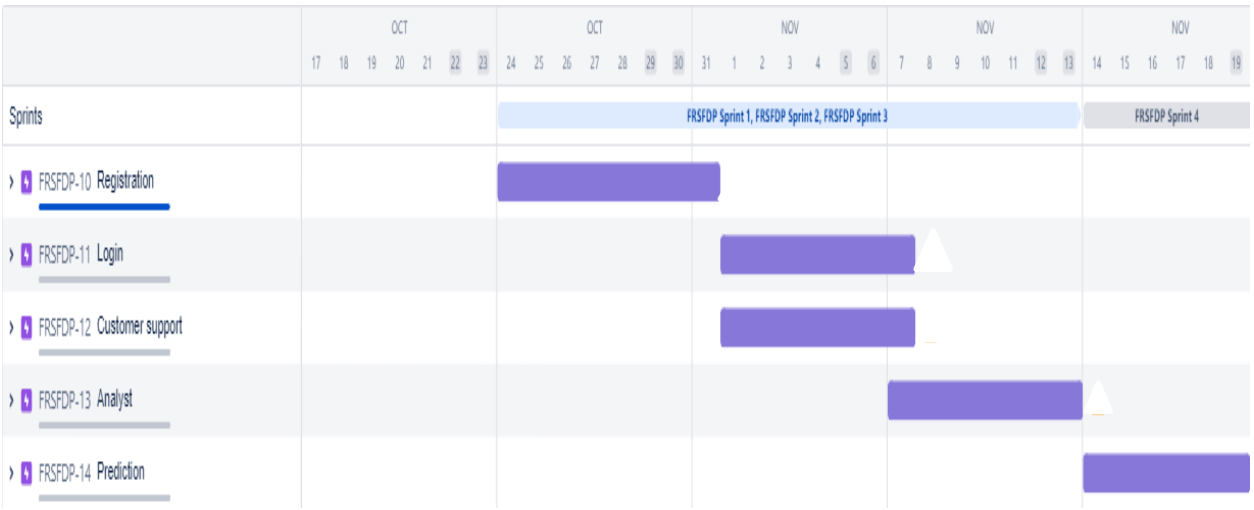


6.2.1 Sprint Delivery Schedule Diagram

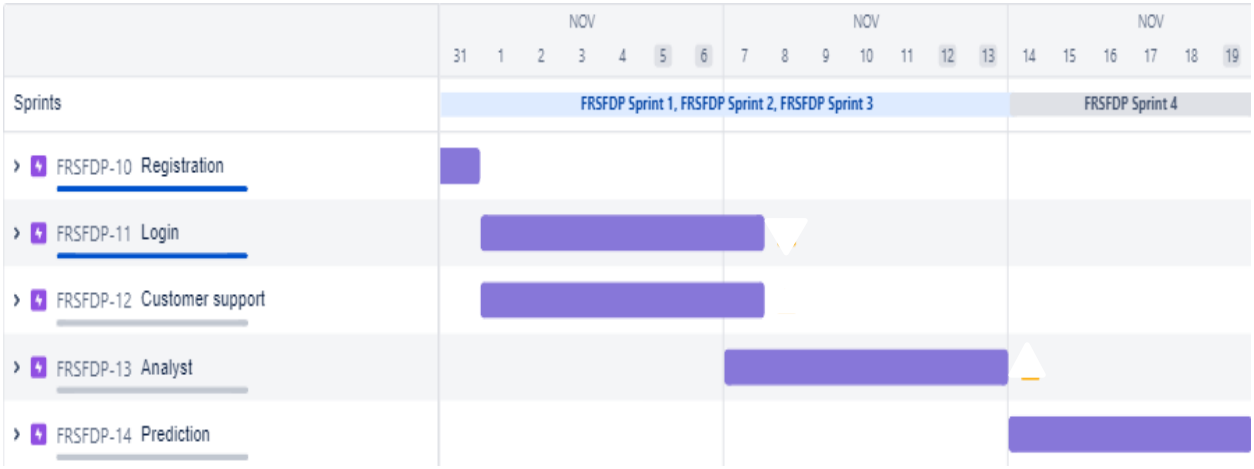
6.3 Reports from JIRA:



6.3.1 Reports From JIRA Diagram



6.3.2 Sprint Schedule 1



6.3.3 Sprint Schedule 2

7.CODING & SOLUTIONING

7.1 Feature 1:

In Our project we use convolution neural network to classify the picture which is loaded to our model the following shows the feature about convolutional neural network

Convolutional Neural Network

A convolutional neural network (CNN or convnet) is a subset of machine learning. It is one of the various types of artificial neural networks which are used for different applications and data types.

A CNN is a kind of network architecture for deep learning algorithms and is specifically used for image recognition and tasks that involve the processing of pixel data.

There are other types of neural networks in deep learning, but for identifying and recognizing objects, CNNs are the network architecture of choice. This makes them highly suitable for computer vision (CV) tasks and for applications where object recognition is vital, such as self-driving cars and facial recognition.

Inside Convolutional Neural Networks

Artificial neural networks (ANNs) are a core element of deep learning algorithms. One type of an ANN is a recurrent neural network (RNN) that uses sequential or time series data as input. It is suitable for applications involving natural language processing (NLP), language translation, speech recognition and image captioning.

The CNN is another type of neural network that can uncover key information in both time series and image data. For this reason, it is highly valuable for image-related tasks, such as image recognition, object classification and pattern recognition. To identify patterns within an image, a CNN leverages principles from linear algebra, such as matrix multiplication. CNNs can also classify audio and signal data.

A CNN's architecture is analogous to the connectivity pattern of the human brain. Just like the brain consists of billions of neurons, CNNs also have neurons arranged in a specific way. In fact, a CNN's neurons are arranged like the brain's frontal lobe, the area responsible for processing visual stimuli.

This arrangement ensures that the entire visual field is covered, thus avoiding the piecemeal image processing problem of traditional neural networks, which must be fed images in reduced-resolution pieces. Compared to the older networks, a CNN delivers better performance with image inputs, and also with speech or audio signal inputs.

CNN Layers

A deep learning CNN consists of three layers:

- A Convolutional Layer,
- A Pooling Layer And
- A fully connected (FC) layer.

The convolutional layer is the first layer while the FC layer is the last.From the convolutional layer to the FC layer, the complexity of the CNN increases. It is this increasing complexity that allows the CNN to successively identify larger portions and more complex features of an image until it finally identifies the object in its entirety.

Convolutional Layer

The majority of computations happen in the convolutional layer, which is the core building block of a CNN. A second convolutional layer can follow the initial convolutional layer. The process of convolution involves a kernel or filter inside this layer moving across the receptive fields of the image, checking if a feature is present in the image.

Over multiple iterations, the kernel sweeps over the entire image. After each iteration a *dot product* is calculated between the input pixels and the filter. The final output from the series of dots is known as a feature map or convolved feature. Ultimately, the image is converted into numerical values in this layer, which allows the CNN to interpret the image and extract relevant patterns from it.

CODING:

```
#Add convolutional layer to model
```

```
From tensorflow.keras.layers import Convolution2D

model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))
```

Pooling layer

Like the convolutional layer, the pooling layer also sweeps a kernel or filter across the input image. But unlike the convolutional layer, the pooling layer reduces the number of parameters in the input and also results in some information loss. On the positive side, this layer reduces complexity and improves the efficiency of the CNN.

CODING:

```
#Add Pooling layer to model
From tensorflow.keras.layers import MaxPooling2D

model.add(MaxPooling2D(pool_size=(2,2)))
```

Fully connected layer

The FC layer is where image classification happens in the CNN based on the features extracted in the previous layers. Here, *fully connected* means that all the inputs or nodes from one layer are connected to every activation unit or node of the next layer.

All the layers in the CNN are not fully connected because it would result in an unnecessarily dense network. It also would increase losses and affect the output quality, and it would be computationally expensive.

How convolutional neural networks work

A CNN can have multiple layers, each of which *learns* to detect the different features of an input image. A filter or kernel is applied to each image to produce an output that gets progressively better and more detailed after each layer. In the lower layers, the filters can start as simple features.

At each successive layer, the filters increase in complexity to check and identify features that uniquely represent the input object. Thus, the output of each convolved image -- the partially

recognized image after each layer -- becomes the input for the next layer. In the last layer, which is an FC layer, the CNN recognizes the image or the object it represents.

With convolution, the input image goes through a set of these filters. As each filter activates certain features from the image, it does its work and passes on its output to the filter in the next layer. Each layer learns to identify different features and the operations end up being repeated for dozens, hundreds or even thousands of layers. Finally, all the image data progressing through the CNN's multiple layers allow the CNN to identify the entire object.

SCREENSHOT

7.2 Feature 2:

Our project we use flask python framework for building web application. In the following section gives the feature about using this flask frame work.

FLASK

Flask was designed to be easy to use and extend. The idea behind Flask is to build a solid foundation for web applications of different complexity. From then on you are free to plug in any extensions you think you need. Also you are free to build your own modules.

Flask is great for all kinds of projects. It's especially good for prototyping. Flask depends on two external libraries: the Jinja2 template engine and the Werkzeug WSGI toolkit.

Still the question remains why use Flask as your web application framework if we have immensely powerful Django, Pyramid, and don't forget web megaframework Turbogears?

Those are supreme Python web frameworks BUT out-of-the-box Flask is pretty impressive too with its:

- Built-In Development Server And Fast Debugger
- Integrated Support For Unit Testing
- Restful Request Dispatching
- Jinja2 Templating
- Support For Secure Cookies (Client Side Sessions)
- Wsgi 1.0 Compliant
- Unicode Based

Plus Flask gives you so much more CONTROL on the development stage of your project. It follows the principles of minimalism and let's you decide how you will build your application.

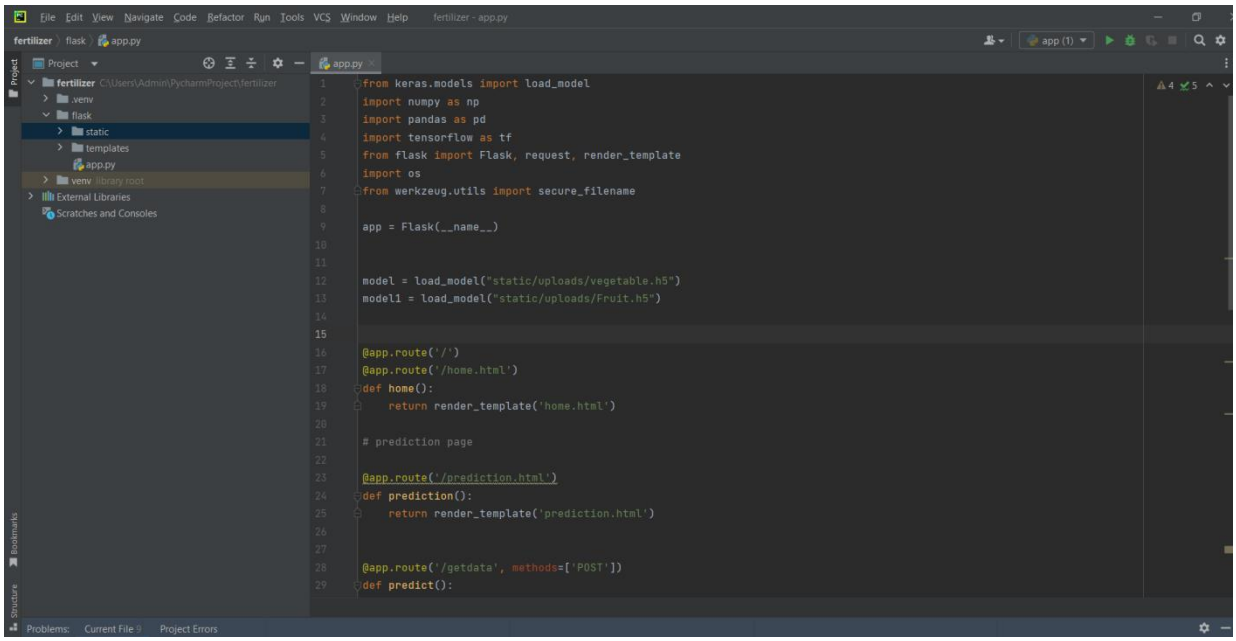
- Flask has a lightweight and modular design, so it easy to transform it to the web framework you need with a few extensions without weighing it down
- ORM-agnostic: you can plug in your favourite ORM e.g. SQLAlchemy.

  Basic foundation API is nicely shaped and coherent.
- Flask documentation is comprehensive, full of examples and well structured. You can even try out some sample application to really get a feel of Flask.
- It is super easy to deploy Flask in production (Flask is 100% WSGI 1.0 compliant")
- HTTP request handling functionality

- High Flexibility
- The configuration is even more flexible than that of Django, giving you plenty of solution for every production need.

To sum up, Flask is one of the most polished and feature-rich micro frameworks available. Still young, Flask has a thriving community, first-class extensions, and an elegant API**.** Flask comes with all the benefits of fast templates, strong WSGI features, thorough unit testability at the web application and library level, extensive documentation**.**

SREENSHOT:



8.TESTING

8.1 TEST CASES:

Test cases help guide the Tester through a Sequence of steps to validate whether a Software Application is free to bugs, and Working as required by the End-user. Learning how to write test cases for Software requires basic Writing Skills, an attention to detail, and good understanding of the Application Under Test.

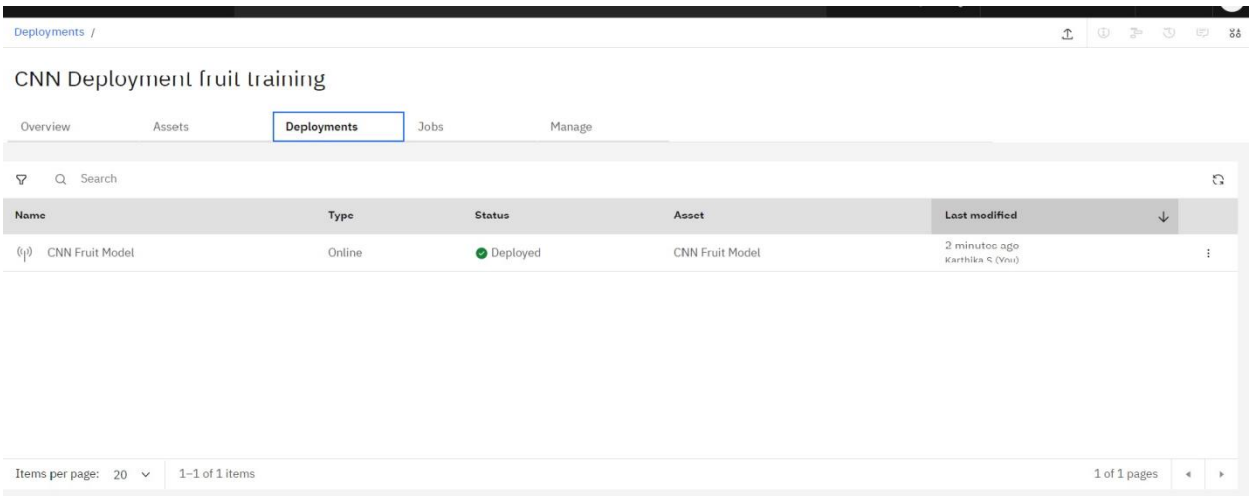
Typically, Test cases for a given module or part of an application, are grouped into a test suite. More often than not, a test session will include many Test cases because there will usually be more than one Specific Scenario to be Tested.

```
pwd
!pip install keras
!pip install tensorflow


from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Convolution2D
from tensorflow.keras.layers import MaxPooling2D
from tensorflow.keras.layers import Flatten
from keras.preprocessing.image import ImageDataGenerator


train_datagen=ImageDataGenerator(rescale = 1./255,shear_range = 0.2,zoom_range =
0.2,horizontal_flip = True)
test_datagen = ImageDataGenerator(rescale = 1)
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3
def __iter__(self): return 0
# @hidden_cell
```

```
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='Hodlo4j3jBVhj6tl21ig6T2_v7KZZSoXz5ysLbksQx4a',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.ap.cloud-object-storage.appdomain.cloud')
bucket = 'fertilizerrecommendationsystem-donotdelete-pr-qmx4g9oclsopa8'
object_key = 'Fertilizers_Recommendation_ System_For_Disease_ Prediction.zip'
streaming_body_1 = cos_client.get_object(Bucket=bucket, Key=object_key)['Body']
# Your data file was loaded into a botocore.response.StreamingBody object.
# Please read the documentation of ibm_boto3 and pandas to learn more about the possibilities to load the data.
# ibm_boto3 documentation: https://ibm.github.io/ibm-cos-sdk-python/
# pandas documentation: http://pandas.pydata.org/
from io import BytesIO
import zipfile
unzip = zipfile.ZipFile(BytesIO(streaming_body_1.read()),'r')
file_paths = unzip.namelist()
for path in file_paths:
    unzip.extract(path)
import os
filenames = os.listdir('/home/wsuser/work/Dataset Plant Disease/fruit-dataset/fruit-dataset/train')
x_train =train_datagen.flow_from_directory("/home/wsuser/work/Dataset Plant Disease/fruit-dataset/fruit-dataset/train",target_size = (128,128),batch_size = 32,class_mode = 'categorical')
x_test =test_datagen.flow_from_directory("/home/wsuser/work/Dataset Plant Disease/fruit-dataset/fruit-dataset/test",target_size = (128,128),batch_size = 32,class_mode = 'categorical')
x_train.class_indices
model=Sequential()
model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))
```

```
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
model.summary()

model.add(Dense(6,activation='softmax'))
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
model.fit_generator(x_train,steps_per_epoch=len(x_train),validation_data=x_test,validation_ste
ps=len(x_test),epochs=10)
model.fit_generator(x_train,steps_per_epoch=len(x_train),validation_data=x_test,validation_ste
ps=len(x_test),epochs=10)
!tar -zcvf fruit-classification.tgz Fruit.h5
!pip install watson-machine-learning-client
!pip install ibm_watson_machine_learning
from ibm_watson_machine_learning import APIClient

wml_credentials ={
    "url":"https://jp-tok.ml.cloud.ibm.com",
    "apikey":"jofw6uEvdPqcQshABjVcEjRyRgbEvv-sl3j7l7-MVPlZ"
    }

client = APIClient(wml_credentials)
client
client.spaces.get_details()
def guid_from_space_name(client, space_name):
  space = client.spaces.get_details()
  return(next(item for item in space['resources'] if item['entity']['name'] ==
space_name)['metadata']['id'])
space_uid = guid_from_space_name(client, 'CNN Deployment fruit training')
print("Space UID = " +space_uid)
client.set.default_space(space_uid)
client.software_specifications.list()
```

```
software_spec_uid = client.software_specifications.get_id_by_name("tensorflow_rt22.1-py3.9")
software_spec_uid
model_details = client.repository.store_model(model='fruit-classification.tgz',meta_props={
    client.repository.ModelMetaNames.NAME:"CNN Fruit Model",
    client.repository.ModelMetaNames.TYPE:"tensorflow_2.7",
    client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_spec_uid
})

model_details
model_id = client.repository.get_model_id(model_details)
model_id
client.repository.download(model_id,'IBM_Fruit_model.tar.gb')
```



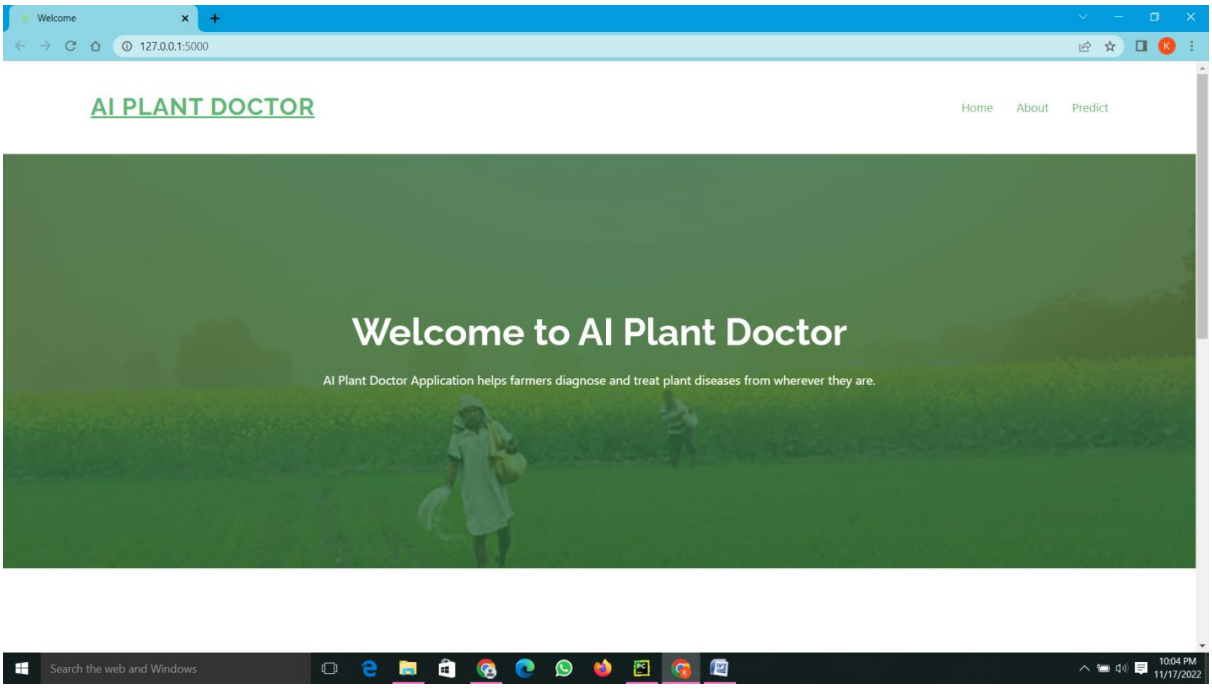
8.2 USER ACCEPTANCE TESTING:

User Acceptance Testing is the final testing stage in Software Development before Production. It's used to get feedback from users who test the Software and it's User Interface (UI).User Acceptance Testing is Usually done manually, with users Creating real-world situations and testing how the Software reacts and performs.

9.RESULTS

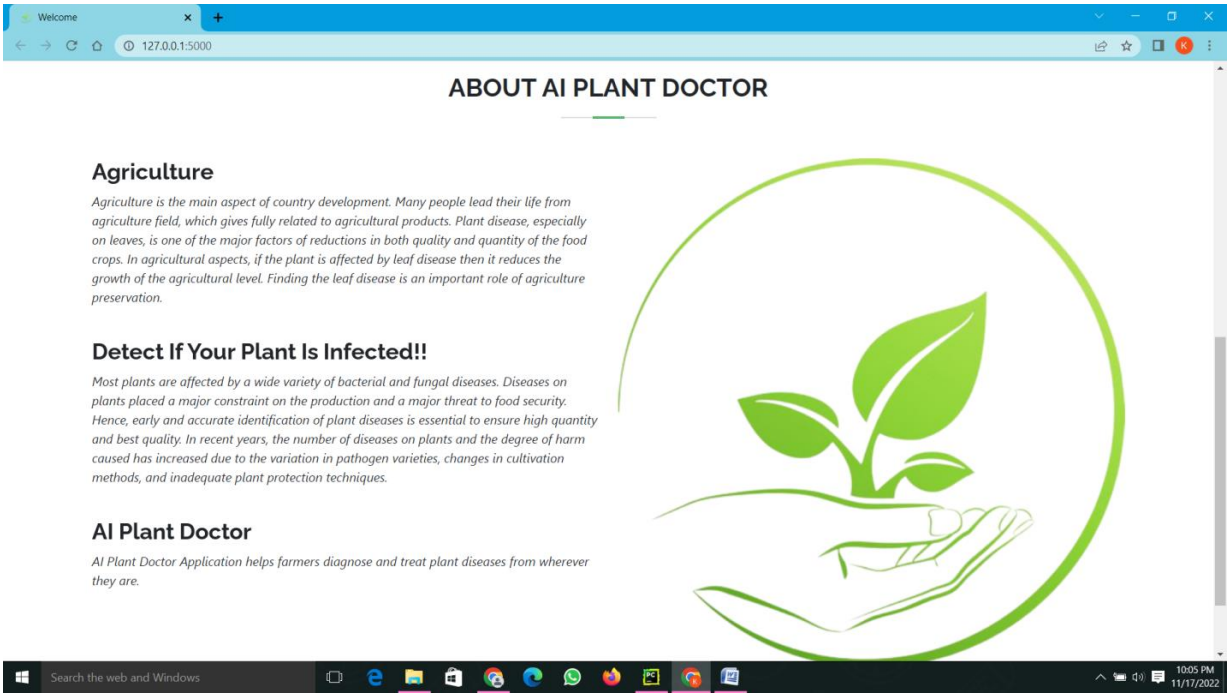
SCREENSHOT:

Home page:



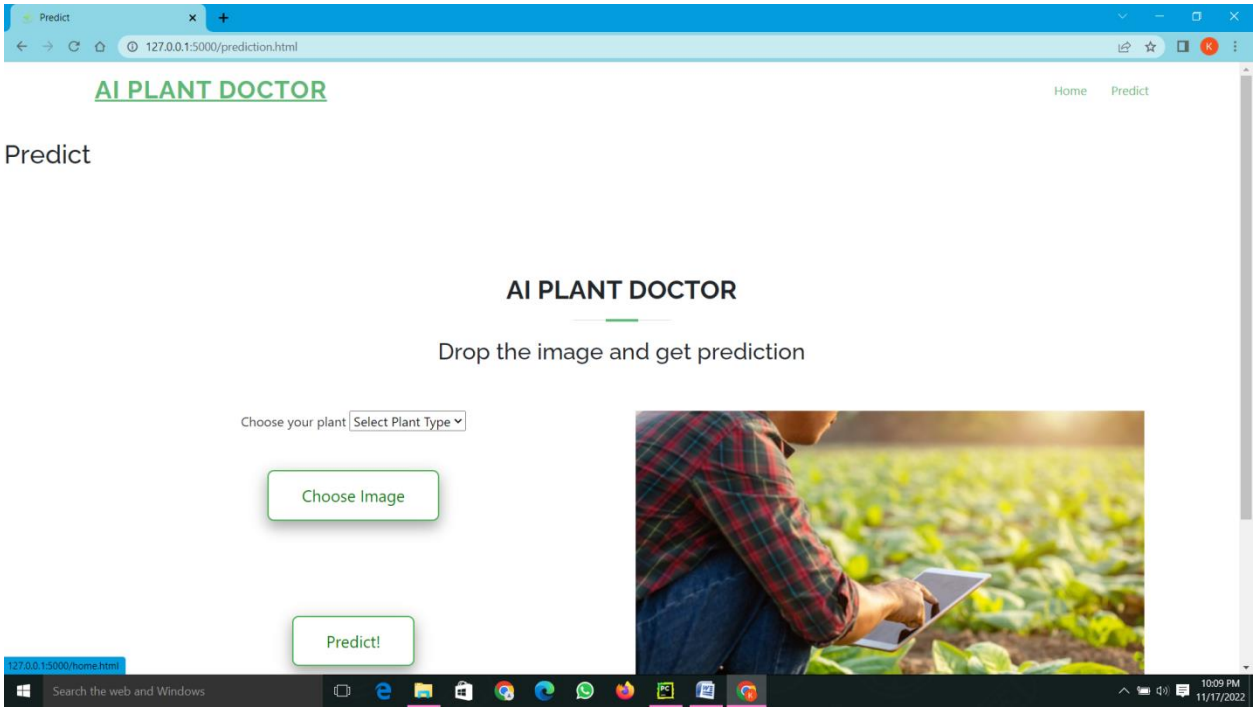
9.1 Home page

Navigate to about:



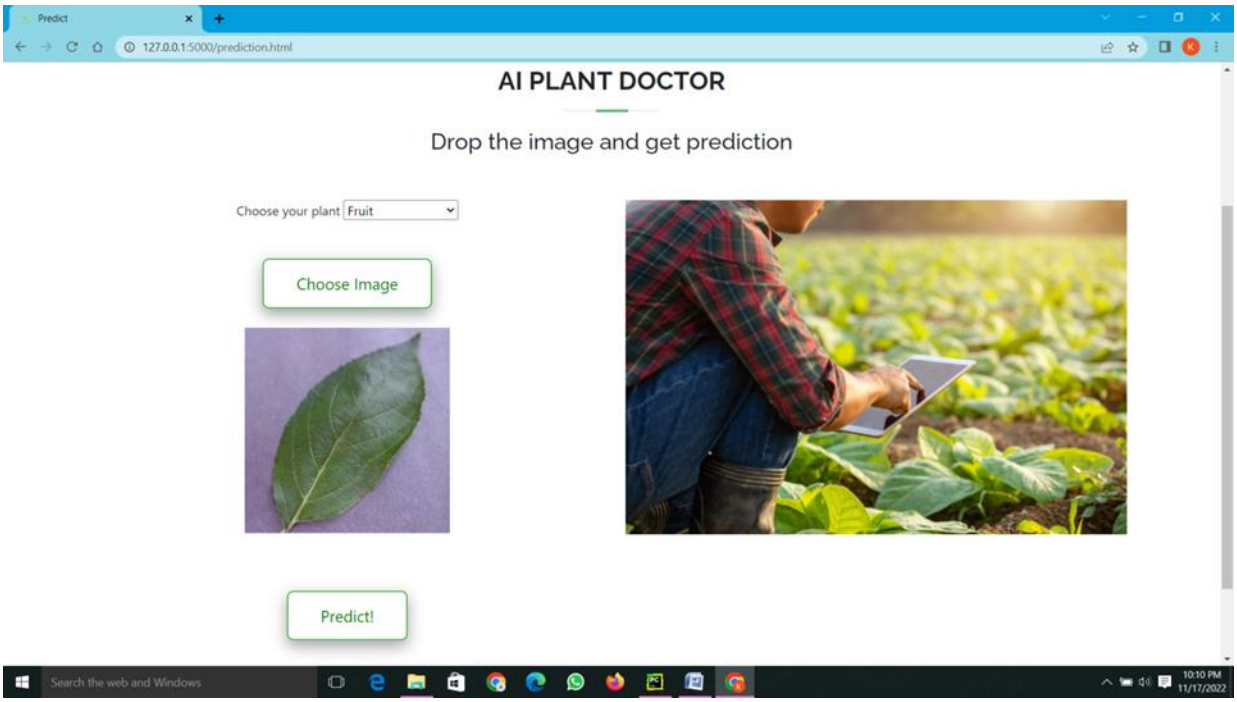
9.2 Navigate to About Page

Predict page



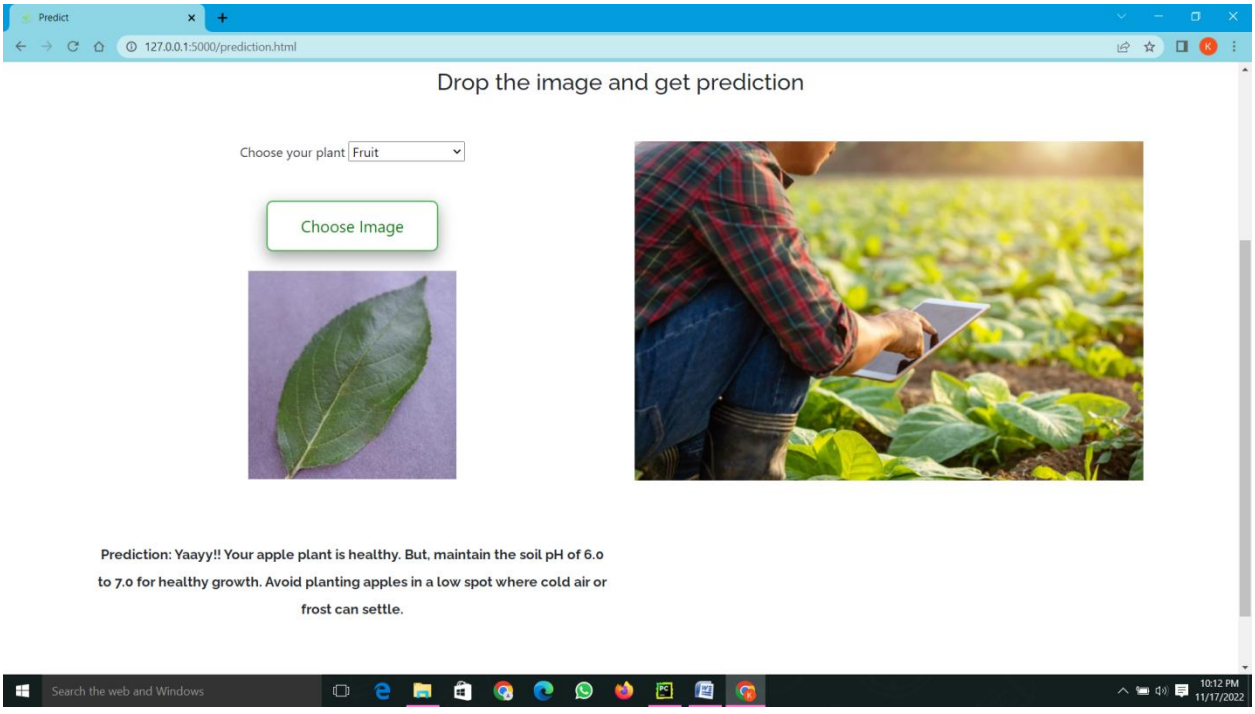
9.3 Predict page

After selecting plant type and choose image from local system



9.3 Choose Image From Local System

After predict



9.4 After Predict Page

10. ADVANTAGES AND DISADVANTAGES

10.1 Advantages:

- Disease analysis is done for the leaf diseases detection is addressed the analysis of the different diseases that are present on the leaves can be effectively detected in the early stage before it will damage the whole plant.
- The system presented can able to detect the diseases more accurately we can say that we can archive good productivity by preventing the different diseases.
- The system described works faster and gives better accuracy in prediction to predict the disease and exact fertilizer about the plant disease.
- There is no need to meet the agriculture experts.
- This prediction makes the farmers to improve the fertility, growth and quality of plants.
- Accuracy of system is its ability to differentiate the diseased and healthy case of plants correctly.
- One of the most efficient and rapid methods of disease detection.
- Early detection and management of problems.
- Constantly of constant work.
- It enhances the metabolism of plants.
- No knowledge is required to perform this process.
- This process operating on any operating system so that is platform independent.
- In this system is easy to understand for farmers and everyone.
- In this system provides the exact fertilizers become helpful for the good production of plants due to their numerous benefits which promote the fast and healthy growth of plants.
- System are quick in providing plant nutrients and easy to grow the plants.
- we provides the Fertilizers make plants more resistant to pests. As a result, they are using fewer insecticides and herbicides, which results in healthier crops. Hence, fewer illnesses are present, giving the crops an aesthetic value.

10.2 Disadvantages:

- Not accurate at all the time due to lack of data.
- Some people fear about is it actually works.
- Is recommended fertilizer available in the user location.
- For training and testing, the proposed model requires very high computational time.
- The neural network architecture used in this project work has high complexity.

11. CONCLUSION

Agricultural has always been the most important sector for survival. There are a lot of difficulties faced by our farmers these days due to various unpredictable reasons. Hence, as engineers, we need to collaborate with farmers and provide them a solution to improve the quality quantity of plants. Our project is the first step towards it. The convolutional neural network model was implemented in TensorFlow backend system to classify the plant diseases. The usage of classification and feature extraction processes has enhanced the performance of the system which provides better final results.

The model proposed here involves image classification of fruit datasets and vegetable datasets. The following points are observed during model testing and training:

- The accuracy of classification increased by increasing the number of epochs.
- For different batch sizes, different classification accuracies are obtained.
- The accuracies are increased by increasing more convolution layers.
- The accuracy of classification also increased by varying dense layers.
- Different accuracies are obtained by varying the size of kernel used in the convolution layer output.
- Accuracies are different while varying the size of the train and test datasets.

12.FUTURE SCOPE

- This further research is implementing the proposed algorithm with the existing public datasets. Also, various segmentation algorithms can be implemented to improve accuracy. The proposed algorithm can be modified further to identify the disease that affects the various plant organs such as stems and fruits.

- proposed method uses SVM to classify tree leaves, identify the disease and suggest the fertilizer. The proposed method is compared with the existing CNN based leaf disease prediction. The proposed SVM technique gives a better result when compared to existing CNN.

- The model in this project work can be extended to image recognition. The entire model can be converted to application software using python to exe software. The real time image classification, image recognition and video processing are possible with help OpenCV python library.

- As a smart farming methodologies increase , there would be a vast requirement for newer technologies to be implemented.

- The project which is now a web-based the application can be made into an app where farmers can be educated and informed about their crop field.

- Once a prediction is done we can improve on the automation process where the farmers can remotely control the field.

13.APPENDIX

Build Python Code

Importing Libraries

The first step is usually importing the libraries that will be needed in the program.

```
from keras.models import load_model
import numpy as np
import pandas as pd
import tensorflow as tf
from flask import Flask, request, render_template
import os
from werkzeug.utils import secure_filename

```

Importing the flask module into the project is mandatory. An object of the Flask class is our WSGI application. Flask constructor takes the name of the current module (__name__).

Creating our flask application and loading

```
app = Flask(__name__)
```

Routing to the Html Page Here

the declared constructor is used to route to the HTML page created earlier. The '/' route is bound with the bot function. Hence, when the home page of a web server is opened in the browser, the HTML page will be rendered.

```
@app.route('/')
@app.route('/home.html')
def home():
    return render_template('home.html')
```

Main Function

This is used to run the application in localhost.

```
if __name__ == "__main__":
    app.run(debug=False)
```

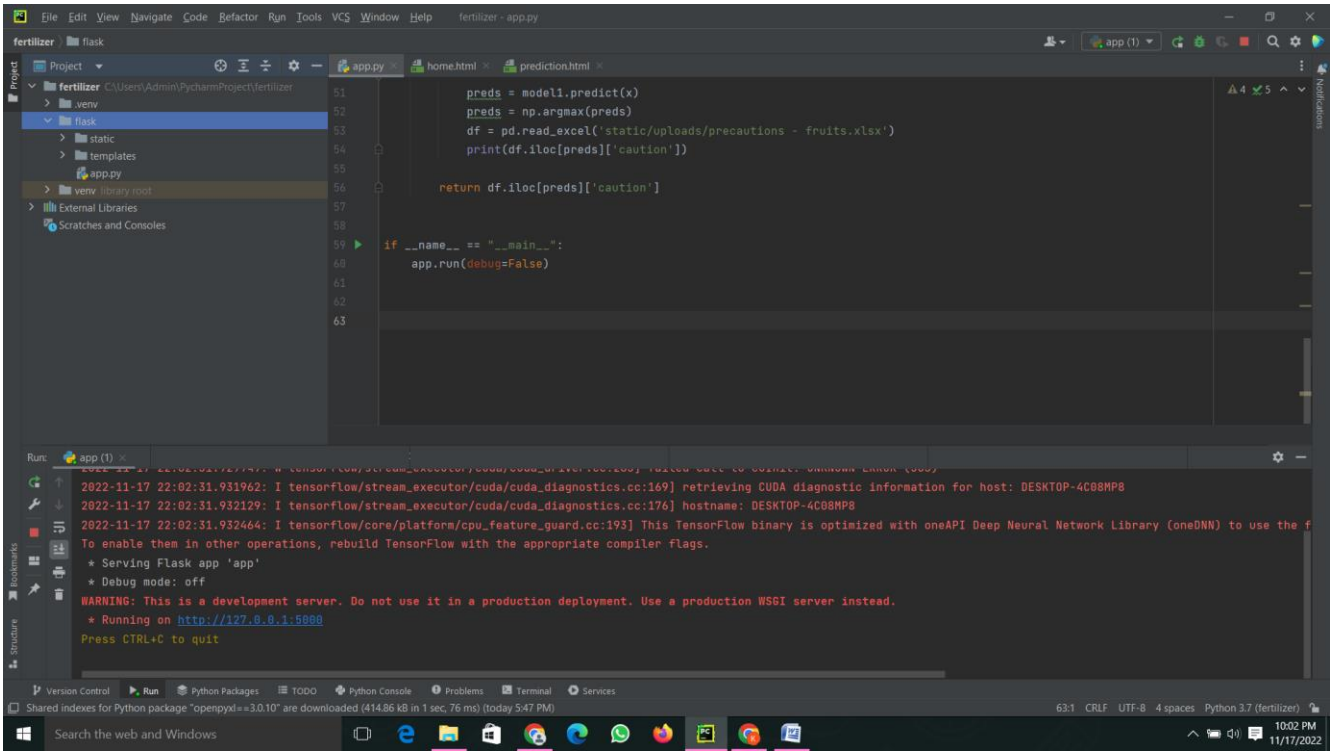
Build HTML Code

- We use HTML to create the front-end part of the web page add CSS for customize the web page.
- Here, we have created 2 HTML page
    - home.html - home.html displays the home page
    - prediction.html – prediction.html shows the prediction page there the image is uploaded from local system and after clicking the predict button it shows the disease of the plant and recomment fertilizer for the disease.

Run The Application

- Open the pycharm from the start menu.
- Navigate to the folder where your app.py resides.
- Run the app.py.
- It will show the local host where your app is running on http://127.0.0.1.5000/
- Copy that localhost URL and open that URL in the browser.

- It does navigate me to where you can view your web page



SOURCE CODE

Source Code For Html Page

home.html

```
<!DOCTYPE html>
<html lang="en">
<head>
 <meta charset="utf-8">
<meta content="width=device-width, initial-scale=1.0" name="viewport">
<title>Welcome</title>
<meta content="" name="description">
<meta content="" name="keywords">
<link href="static/logo.png" rel="icon">
<link    href="https://fonts.googleapis.com/css?family=Open+Sans:300,300i,400,400i,600
,600i,700,700i|Raleway:300,300i,400,400i,500,500i,600,600i,700,700i|Poppins:300,300i,4
00,400i,500,500i,600,600i,700,700i" rel="stylesheet">
<link rel= "stylesheet"  type="text/css" href= "{{ url_for('static',filename='style.css')}}">
<link rel= "stylesheet"  type="text/css" href= "{{ url_for('static',filename='style_1.css')}}">
<link rel= "stylesheet"  type="text/css" href= "{{
url_for('static',filename='bootstrap.min.css') }}">
</head>
<body>
<br>
<header id="header" class="d-flex align-items-center">
<div class="container d-flex align-items-center">
<h1 class="logo me-auto"><a href="home.html">AI Plant Doctor</a></h1>
```

```
<nav id="navbar" class="navbar">
<ul>
<li><a class="nav-link scrollto active" href="#hero">Home</a></li>
<li><a class="nav-link scrollto active" href="#about">About</a></li>
<li><a class="nav-link scrollto active" href="prediction.html">Predict</a></li>
<li><a class="getstarted scrollto" href="#about">Sign In</a></li>
</ul>
<i class="bi bi-list mobile-nav-toggle"></i>
</nav>
</div>
</header>
<br>
<link rel= "stylesheet"  type="text/css" href= "{{ url_for('static',filename='bootstrap.css') }}">
<section id="hero">
<div id="heroCarousel" data-bs-interval="5000" class="carousel slide carousel-fade" data-bs-ride="carousel">
<ol class="carousel-indicators" id="hero-carousel-indicators"></ol>
<div class="carousel-inner" role="listbox">
<div class="carousel-item active" >
<img src="{{url_for('static',filename='slide-1.jpg')}}" class="d-block w-100">
<div class="carousel-container">
<div class="container">
<h2 class="animate__animated animate__fadeInDown">Welcome to <span>AI Plant Doctor</span></h2>
<p class="animate__animated animate__fadeInUp" color="white">AI Plant Doctor Application helps farmers diagnose and treat plant diseases from wherever they are.</p>
</div>
</div>
</div>
</div>
```

```
</div>
</section>
<br><br>
<section id="about" class="about">
<div class="container">
<div class="section-title">
<h2>About AI Plant Doctor</h2>
</div>
<div class="row">
<div class="col-lg-6 order-1 order-lg-2">
<div class="row">
<img src="{{url_for('static',filename='logo.png')}}" >
</div>
</div>
<div class="col-lg-6 pt-4 pt-lg-0 order-2 order-lg-1 content">
<h3 class="animate__animated animate__fadeInDown">Agriculture</h3>
<p class="fst-italic">Agriculture is the main aspect of country development. Many people lead their life from agriculture field, which gives fully related to agricultural products. Plant disease, especially on leaves, is one of the major factors of reductions in both quality and quantity of the food crops. In agricultural aspects, if the plant is affected by leaf disease then it reduces the growth of the agricultural level. Finding the leaf disease is an important role of agriculture preservation.</p>
<br>
<h3 class="animate__animated animate__fadeInDown">Detect If Your Plant Is Infected!!</h3>
<p class="fst-italic">Most plants are affected by a wide variety of bacterial and fungal diseases. Diseases on plants placed a major constraint on the production and a major threat to food security. Hence, early and accurate identification of plant diseases is essential to ensure high quantity and best quality. In recent years, the number of diseases on plants and the degree of harm caused has increased due to the variation in pathogen varieties, changes in cultivation methods, and inadequate plant protection techniques.</p>
```

```
<br>
<h3 class="animate__animated animate__fadeInDown">AI Plant Doctor</h3>
<p class="fst-italic">AI Plant Doctor Application helps farmers diagnose and treat plant diseases from wherever they are.</p>
<br>
</div>
</div>
</div>
</section>
</body>
</html>
```

prediction.html

```
<!DOCTYPE html>
<html lang="en" xmlns=http://www.w3.org/1999/html xmlns="http://www.w3.org/1999/html">
<head>
<meta charset="utf-8">
<meta content="width=device-width, initial-scale=1.0" name="viewport">
<title>Predict</title>
<meta content="" name="description">
<meta content="" name="keywords">
<link href="static/logo.png" rel="icon">
<link
href="https://fonts.googleapis.com/css?family=Open+Sans:300,300i,400,400i,600,600i,700,700i|Raleway:300,300i,400,400i,500,500i,600,600i,700,700i|Poppins:300,300i,400,400i,500,500i,600,600i,700,700i" rel="stylesheet">
<link rel= "stylesheet"  type="text/css" href= "{{ url_for('static',filename='style.css') }}">
<link rel= "stylesheet"  type="text/css" href= "{{ url_for('static',filename='style_1.css') }}">
<link rel= "stylesheet"  type="text/css" href= "{{ url_for('static',filename='bootstrap.min.css') }}">
</head>
```

```
<body>
<!-- ======= Header ======= -->
<header id="header" class="d-flex align-items-center">
<div class="container d-flex align-items-center">
<h1 class="logo me-auto"><a href="home.html">AI Plant Doctor</a></h1>
<nav id="navbar" class="navbar">
 <ul>
<li><a  href="home.html">Home</a></li>
<li><a class="nav-link scrollto active" href="prediction.html">Predict</a></li>
</ul>
<i class="bi bi-list mobile-nav-toggle"></i>
</nav>
</div>
</header>
<br>
<div><h2 >Predict</h2></div>
<form id="upload-file">
<section class="inner-page">
<!-- ======= About Us Section ======= -->
<section id="about" class="about">
<div class="container">
<div class="section-title">
<h2>AI Plant Doctor</h2>
<p><h3>Drop the image and get prediction</h3></p>
<div class="row">
<div class="col-lg-6 order-1 order-lg-2">
<img src="{{url_for('static',filename='take picture.jpg')}}" class="img-fluid" >
</div>
<div class="col-lg-6 pt-4 pt-lg-0 order-2 order-lg-1 content">
<br>
<br>
```

```
<label for="plant">Choose your plant</label>
<select name="plant" id="plant">
<option value="select" style="color: 'red"'>Select Plant Type</option>
<option name="vegetable" value="vegetable" style="color: 'red"'>Vegetable</option>
<option name="fruit" value="fruit">Fruit</option>
</select>
<div >
<br><br>
<p><input type="file" name="image" id="imageUpload" accept=".png, .jpg, .jpeg"
onchange="loadFile(event)" style="display: none;"></p>
<button class ="button1" type="button"><label for="imageUpload" class="upload-label"
style="cursor: pointer;">Choose Image</label></button>
<br>
<br>
<div>
<p><img id='output'></p>
</div>
</div>
<div>
<br>
<br>
<br>
<button type="button" class="button1 " id="btn-predict" ><label>Predict!</label></button>
</div>
<div class="loader" style="display:none;"></div>
<h3><span id="result" style="font-size:17px; "> </span></h3>
</div>
</div>
</div>
</div>
</section>
```

```
</section>
</form>
</body>
<footer>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js"
type="text/javascript"></script>
 <script src="{{ url_for('static', filename='main.js') }}" type="text/javascript"></script>
</footer>
<script >
var loadFile = function(event) {
var image = document.getElementById('output');
image.src = URL.createObjectURL(event.target.files[0]);
};
</script>
</html>
```

Python Code

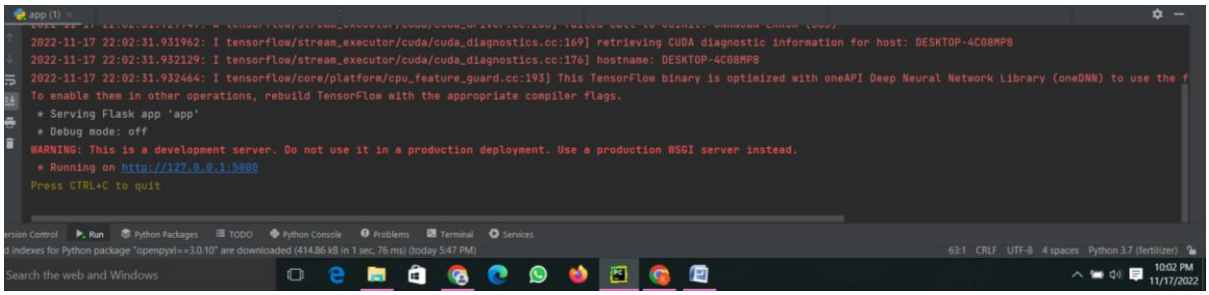
app.py

```
from keras.models import load_model
import numpy as np
import pandas as pd
import tensorflow as tf
from flask import Flask, request, render_template
import os
from werkzeug.utils import secure_filename
app = Flask(__name__)

model = load_model("static/uploads/vegetable.h5")
model1 = load_model("static/uploads/Fruit.h5")

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

```
@app.route('/home.html')
def home():
    return render_template('home.html')
# prediction page

@app.route('/prediction.html')
def prediction():
    return render_template('prediction.html')

@app.route('/getdata', methods=['POST'])
def predict():
    if request.method == 'POST':
        # Get the file from post request
        f = request.files['image']
        # Save the file to ./uploads
        basepath = os.path.dirname(__file__)
        file_path = os.path.join(basepath, 'static/uploads', secure_filename(f.filename))
        f.save(file_path)
        img = tf.keras.preprocessing.image.load_img(file_path,target_size=(128, 128))

        x = tf.keras.preprocessing.image.img_to_array(img)
        x = np.expand_dims(x, axis=0)

        plant = request.form['plant']
        print(plant)
        if (plant == "vegetable"):
            preds = model.predict(x)
            preds = np.argmax(preds)
            print(preds)
            df = pd.read_excel('static/uploads/precautions - veg.xlsx')
            print(df.iloc[preds]['caution'])
```

```
        else:
            preds = model1.predict(x)
            preds = np.argmax(preds)
            df = pd.read_excel('static/uploads/precautions - fruits.xlsx')
            print(df.iloc[preds]['caution'])

        return df.iloc[preds]['caution']


if __name__ == "__main__":
    app.run(debug=False)
```

Demo link:



GITHUB LINK: https://github.com/IBM-EPBL/IBM-Project-1396-1658386731

PROJECT LINK: https://youtu.be/aKH3Q9AJILE