PROJECT REPORT

Team ID : PNT2022TMID05809

CHAPTER 1
INTRODUCTION

1.1 PROJECT OVERVIEW

In today's society, agriculture is the most significant industry. An extensive range of bacterial and fungal diseases harm the majority of plants. Plant diseases severely limited productivity and posed a serious threat to food security. To achieve maximum quantity and optimum quality, early and accurate identification of plant diseases is crucial. The variety of pathogen strains, adjustments to production practices, and insufficient plant protection systems have all contributed to an increase in the number of plant diseases in recent years, as well as the severity of the damage they inflict.

An automated system is introduced in order to identify different diseases in plants by checking the symptoms shown on the leaves of the plant by capturing the images. Deep learning techniques are employed to identify the plant diseases in both fruits and vegetables, and suggest appropriate fertilizers that can be taken for those diseases. In addition to this, recommended precautions and immediate actions are suggested to the end users as a means to prevent any further infestation due to potentially harmful diseases and to create awareness to users irrespective of their role in farming.

The system is capable of capturing images from multiple angles to achieve higher model accuracy. Thus the trained model employed to predict any further crop infestation is essentially a precautionary tool for farmers who are unaware of the crop patterns in leaves that can potentially lead to widespread infection.

1.2 PURPOSE

The main objective of this project is to design an automated system to address the issues and difficulties faced by the farmer while harvesting crops. Thus addressing these issues becomes essential in order to enable farmers to lead a profitable business and thereby increasing their crop yield.

Moreover, the system is designed in order to solve the financial loss that can be possibly incurred as a result of lack of constant supervision and care and utilizing generous amounts of fertilizers, resulting in unnecessary expenses and crop damage.

Additionally, the developed model enables farmers to determine the recommended usage and dosage of suitable fertilizers and prevent them from treating plants with undesirable fertilizers and other treatment procedures.

Inappropriate chemical fertilizer application to plants depletes essential soil nutrients, thus lowering the amount of minerals and vitamins in food products. Giving the right nutrients to damaged plants enables them to overcome nutritional shortages.

CHAPTER 2
LITERATURE SURVEY

2.1 EXISTING PROBLEMS

PROBLEM - 1:

Indian farmers are already squeezed by a massive shortage of fertilizers and are turning to the black market, paying exorbitant prices for supplies. Additionally, if they use the incorrect fertilizers for their damaged crops, it results in further financial loss and fertilizer waste.

In an interview with Business Standard Farmer Patidar said:

“On the black market, a 45-kilogram bag of di-ammonium phosphate is selling for 1,500 rupees ($20), above the maximum retail price of 1,200 rupees. A bag of urea costs as much as 400 rupees compared with the usual price of 266 rupees.”

By applying fertilizers properly, this growing inflation can be substantially slowed down.

PROBLEM - 2:

Mineral deficiencies lead to plant diseases. Application of incorrect chemical fertilizers to plants consumes an indispensable part of the nutrients in soil reducing the amount of minerals and vitamins in the food items. Providing the appropriate fertilizers helps in overcoming nutrition deficiencies in the diseased plants.

2.2 REFERENCES

LITERATURE SURVEYS:

[1] Dhruvi Gosai, Binal Kaka, Dweepna Garg, Radhika Patel, Amit Ganatra, "Plant Disease Detection and Classification Using Machine Learning Algorithm",
2022
https://ieeexplore.ieee.org/document/9726036

In this paper, the researchers have trained a model to recognize some unique harvests and 26 diseases from the public dataset which contains 54,306 images of the diseases and healthy plant leaves that are collected under controlled conditions. This paper worked on the ResNets algorithm.

[2] Shloka Gupta, Nishit Jain, Akshay Chopade, Aparna Bhonde, "Farmer's Assistant: A Machine Learning Based Application for Agricultural Solutions", 2022
https://arxiv.org/pdf/2204.11340.pdf

In this paper, the researchers proposed a system which helps farmers detect plant disease, recommend the ideal crop for their soil and recommend fertilizers for them to get the best yield possible. They used the EfficientNet deep learning model, which achieves 99.8% validation accuracy on the choice of dataset for plant disease detection. Random Forest model for crop recommendation based on the soil (N, P, K, pH) and weather features, and a rule-based classification system for fertilizer recommendation

[3] Humberto M. Beneduzzi , Eduardo G. de Souza , Wendel K. O. Moreira, "Fertilizer Recommendation Methods For Precision Agriculture – A Systematic Literature Study", 2022
https://www.scielo.br/j/eagri/a/9PXBwFNVkxVvSSz6gkDSFVg/?format=pdf&lang=en

In this paper, the researchers studied and worked on the fertilizer recommendation methods used in site-specific nutrient management and the calculation methodologies for N, P, and K recommendations. For this purpose, a systematic literature study (SLS), consisting of systematic literature mapping, snowballing, and systematic literature review was performed. The analyzed studies were grouped into five domains (precision agriculture, soil fertility, site-specific nutrient application, fertilizer recommendation methods, and recommendation software for site-specific nutrient application). As a result, the SLS identified 12 methods for recommending N, nine for recommending P, and six for recommending K, in addition to five computer programs for precision agriculture that perform fertilizer recommendations at varying rates.

[4] Senthil Kumar Swami Durai, Mary Divya Shamili, "Smart farming using Machine Learning and Deep Learning techniques",
Decision Analytics Journal, Volume 3, 2022, 100041, ISSN 2772-6622,
https://www.sciencedirect.com/science/article/pii/S277266222200011X

In this research paper, the researchers have identified the insects and pests present in farms and also have suggested pesticides for the predicted insect. To predict the Weeds RESNET 152 V2 pre-trained algorithm was used. It resulted in an accuracy of 0.98. As the epochs increased the accuracy also increased and also the loss decreased. This showed that the model learned progressively as the epochs increased.

[5] Anis Ahmad, Dharmendra Saraswat, Aly El Gamal, "A Survey On Using Deep Learning Techniques For Plant Disease Diagnosis And Recommendations For Development Of Appropriate Tools, Smart Agricultural Technology", Volume 3, 2023, 100083, ISSN 2772-3755,
https://www.sciencedirect.com/science/article/pii/S277237552200048X

This study presents a comprehensive overview of several studies on deep learning applications and the trends associated with their use for disease diagnosis and management in agriculture. The review is focused on providing a detailed assessment and considerations for developing deep learning-based tools for plant disease diagnosis.

REFERENCES:

[1] J. Liu, X.Wang, "Plant diseases and pests detection based on deep learning: a review", Plant Methods 17, 22 (2021).
https://plantmethods.biomedcentral.com/articles/10.1186/s13007-021-00722-9

[2] A. V. Panchal, S. C. Patel, K. Bagyalakshmi, P. Kumar, I. Raza Khan, M. Soni, "Image-based Plant Diseases Detection using Deep Learning" (2021)

https://www.sciencedirect.com/science/article/pii/S2214785321051403?via%3Dihub

[3] Sharma Abhinav, Jain Arpit, Gupta Prateek, Chowdary Vinay, "Machine Learning Applications for Precision Agriculture: A Comprehensive Review", IEEE (2020)
https://ieeexplore.ieee.org/abstract/document/9311735

[4] Neethu K.S1, P .Vijay Ganesh, "Leaf Disease Detection and Selection of Fertilizers using Artificial Neural Network" (2017)
https://www.irjet.net/archives/V4/i6/IRJET-V4I6354.pdf

[5] Paustian Margit, Theuvsen Ludwig, "Adoption of Precision Agriculture Technologies by German Crop Farmers", Springer (2016)
https://link.springer.com/article/10.1007/s11119-016-9482-5

2.3 PROBLEM STATEMENTS:

Agriculture is the major factor contributing to the Indian Economy. According to the statistics its GDP sector composition is 17.9%. In order to produce more agricultural products without any wastage, technical advancements are required in this domain. The farmers usually have little control over the usage of fertilizers. There is need for proper guidance for optimal usage of these fertilizers and is required by farmers in order to get more yields and prevent wastage.

CHAPTER 3

IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP

FIG: EMPATHY MAP

3.2 IDEATION AND BRAINSTORMING

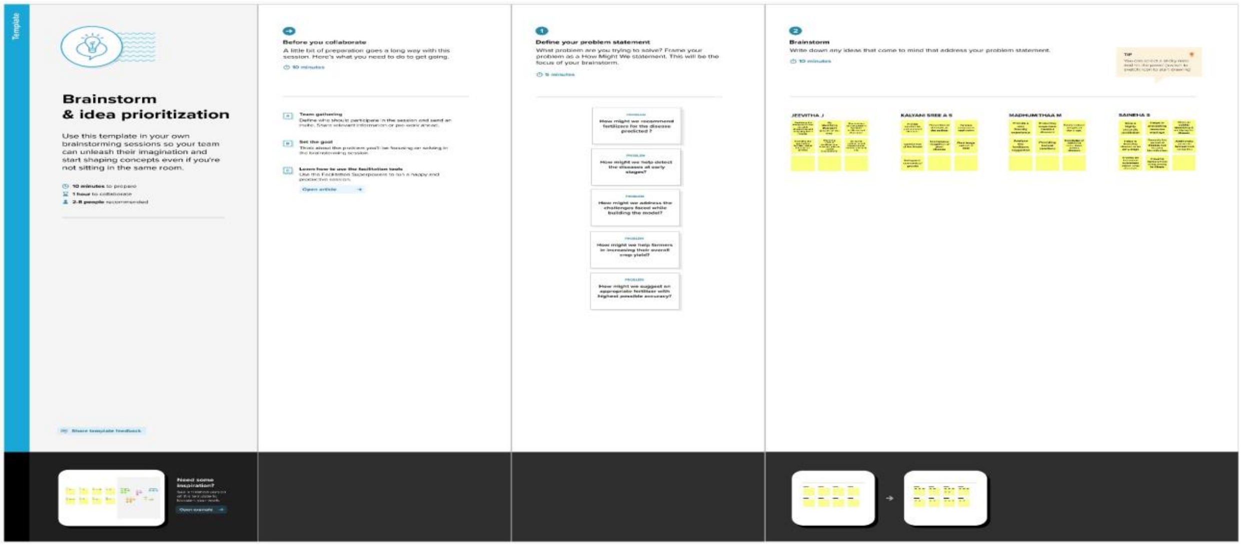

FIG: BRAINSTORMING AND IDEATION - 1

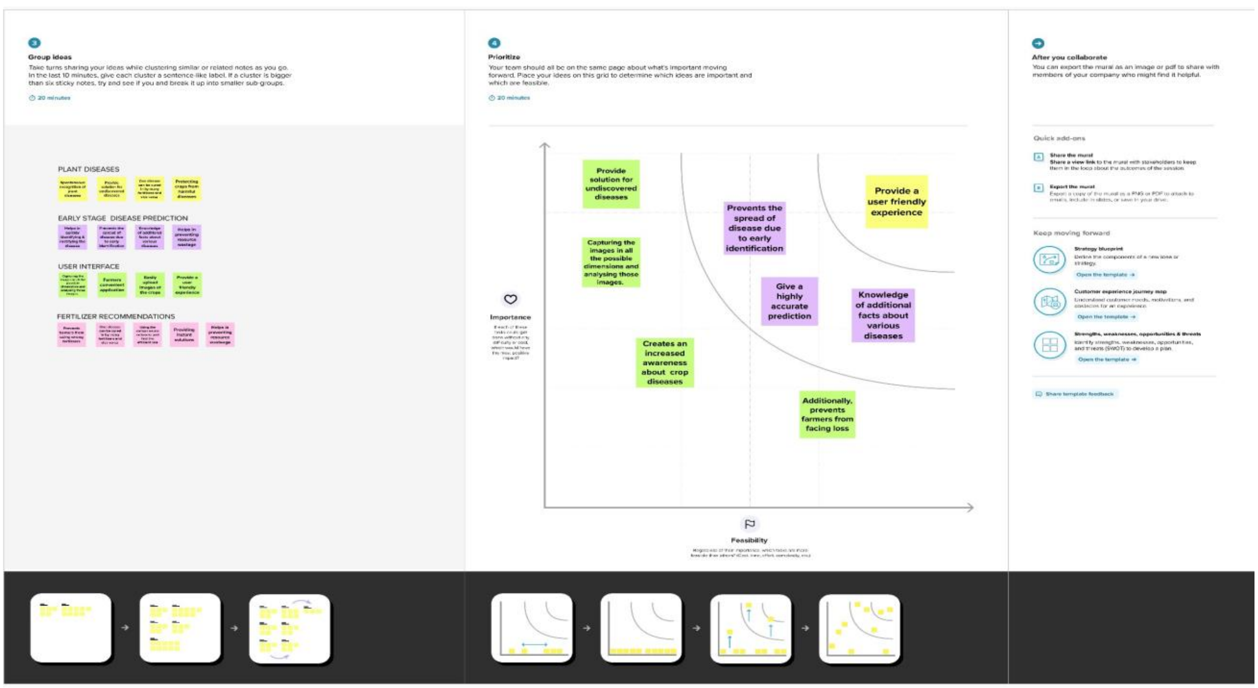

FIG: BRAINSTORMING AND IDEATION - 2

3.3 PROPOSED SOLUTION

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	- To solve the financial loss and fertilizer wastage. - To solve usage of incorrect fertilizers.
2.	Idea / Solution description	- An automated system is introduced to identify different diseases on plants by checking the symptoms shown on the leaves of the plant by capturing the images. - Deep learning techniques are used to identify the diseases and suggest appropriate fertilizer that can be taken for those diseases.
3.	Novelty / Uniqueness	- Instant solutions for farmer’s queries. - Capturing images from multiple angles to achieve higher model accuracy.
4.	Social Impact / Customer Satisfaction	- Creates an increased awareness about crop diseases. - Prevents farmers from using wrong fertilizers.
5.	Business Model (Revenue Model)	- Helpline support for resolving app related issues. - Service availability depends on the plan subscribed by the farmers.
6.	Scalability of the Solution	- Can be used on a large or small scale, making it practical for farmers as well as common people. - Future diseases that are found and the preventative fertiliser for them can easily be incorporated into the current model, making it highly scalable.

3.4 PROBLEM SOLUTION FIT

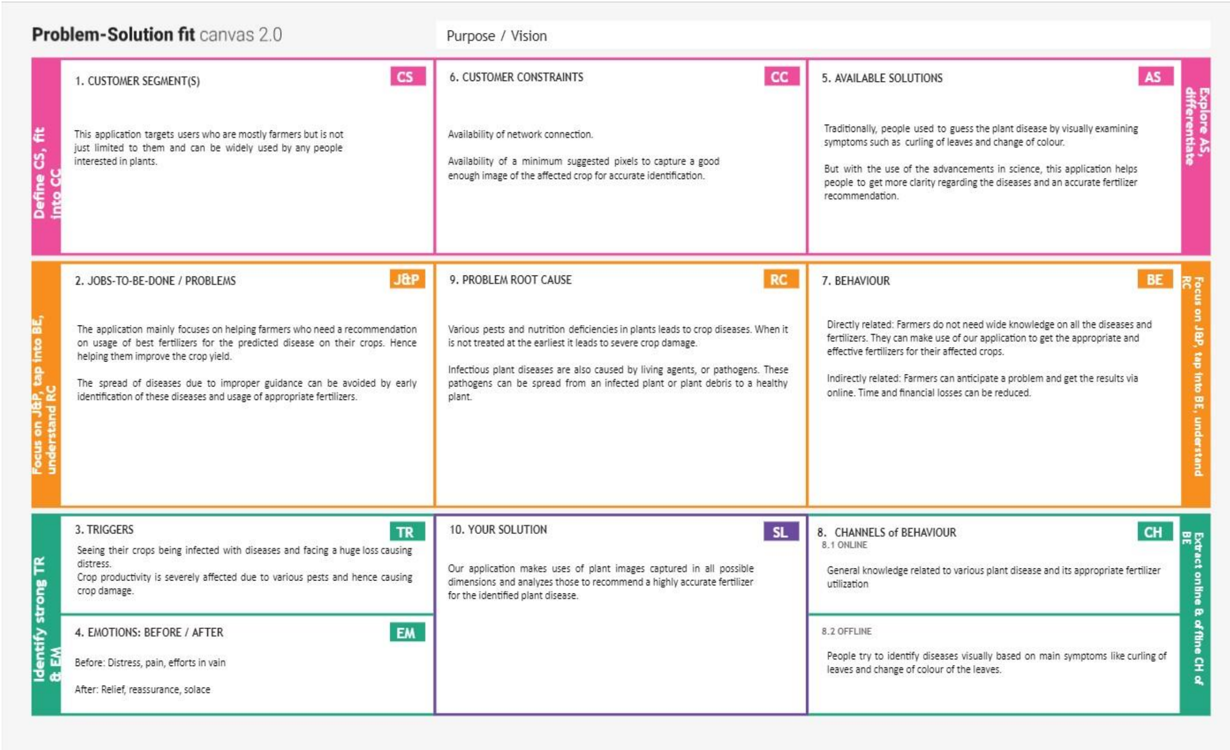

FIG : PROBLEM SOLUTION FIT

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through application
FR-2	User Confirmation	Confirmation via Email
FR-3	Image Uploading	Capturing the images of the leaf from multiple angles
FR-4	Image Processing	Filtering out diseased leaves from the uploaded leaf images.
FR-5	Disease Prediction	Type of disease present in the leaf is identified
FR-6	Fertilizer Recommendation	Appropriate fertilizer is recommended

4.2 NON-FUNCTIONAL REQUIREMENTS:

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

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The application targets a wide range of end users and allows them to access the app anywhere at ease.
NFR-2	**Security**	The application provides security by collecting data from the users that is only necessary for disease prediction models and stores in an encrypted format.
NFR-3	**Reliability**	Final disease prediction & fertilizer suggestions are highly accurate.
NFR-4	**Performance**	The uploaded leaf images will be discarded after a specific time period, ensuring the applications performance is not compromised.
NFR-5	**Availability**	The application is available at all times for users with stable internet connection.
NFR-6	**Scalability**	Service availability depends on the plan subscribed by the farmers. Additionally, helpline support provided for resolving app related issues.

CHAPTER 5

PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

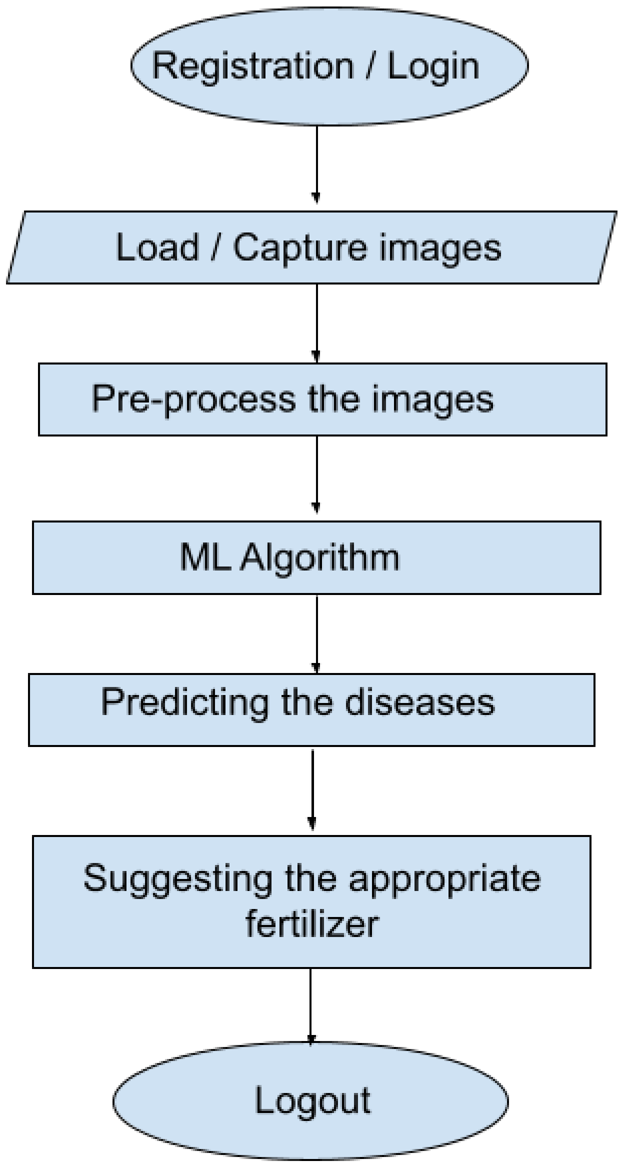

FIG : DATA FLOW DIAGRAMS

5.2 SOLUTION & TECHNICAL ARCHITECTURE

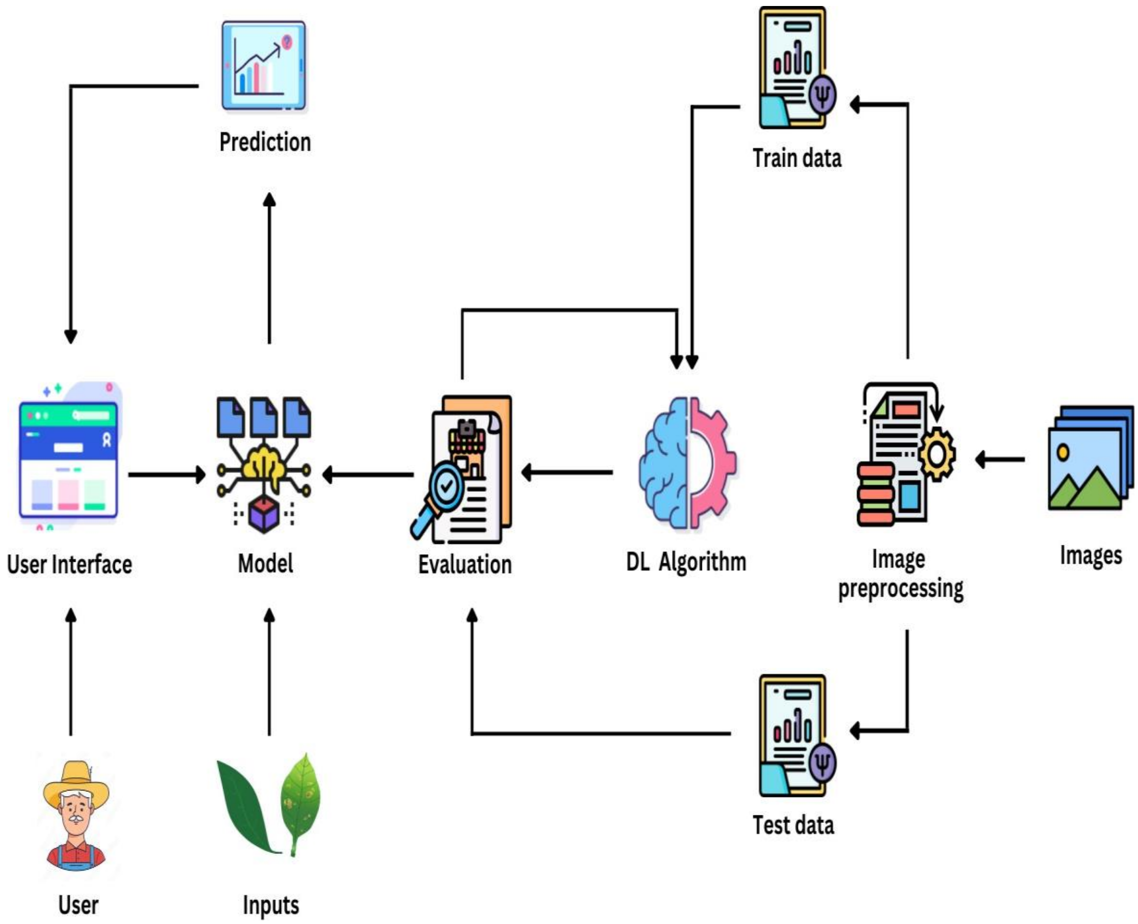

FIG : SOLUTION AND TECHNICAL ARCHITECTURE

5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through registration form	I can register & access the dashboard by logging in using the a desired password	Medium	Sprint-1
	Login	USN-4	As a user, I can log into the application by entering email & password	The email and password is entered by the user is verified	High	Sprint-1
	Dashboard	USN-5	As a user, I can search for diseases and necessary precautions that must be taken to prevent them.	The diseases can either be searched directly or by capturing the image of the disease.	High	Sprint-2

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	Dashboard	USN-6	As a user, I can search the fertilizers recommended by the application and the preventative measures	The diseases are predicted for the uploaded leaf images and	High	Sprint-2
Customer Care Executive	Dashboard	USN-7	As a user, helpline support can be sought out for using the provided services.	Common people, especially farmers can easily access the helpline support option where customerI care representatives can get in touch for speedy advice.	Medium	Sprint-3
Administrator	Admin Login	USN-8	As an administrator, the user activity can be monitored. Moreover, application level features can be modified.	Admins have higher privileges and can monitor the app functions at any point of time.	Medium	Sprint-4

CHAPTER 6

PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	5	High
		USN-2	As a user, I can register for the application through Gmail	3	Low
Sprint-1	Login	USN-3	As a user, I can log into the application by entering email & password	5	High
Sprint-1	Dashboard	USN-4	As a user, I can view the diseases and appropriate fertilizers which have been discovered earlier for the farmers reference.	4	Medium
Sprint-1	Image upload	USN-5	As a user, I can upload a crop image to predict diseases and get fertilizer.	3	Low

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority
Sprint-2	Registration	USN-6	As a user, I will receive confirmation email once I have registered for the application	6	High
Sprint-2	Login	USN-7	As a user, I can login via google account directly	3	Low
Sprint-2	Dashboard	USN-8	As a user, I can view the diseases and appropriate fertilizers with more details.	5	Medium
Sprint-2	Image upload	USN-9	As a user, I can upload crop images from various angles.	6	High
Sprint-3	Registration	USN-10	As a user, I can register for the application through Facebook	4	Low
Sprint-3	Dashboard	USN-11	As a user, I can view the dashboard with segregated tabs like fruit diseases and vegetable diseases.	8	High
Sprint-3	Image upload	USN-12	As a user, I can upload the images which will get preprocessed in the back end.	8	High
Sprint-4	Dashboard	USN-13	As a user, I can view the dashboard with various fertilizers available in the nearby market along with the procedure for using it.	10	High
Sprint-4	Image upload	USN-14	As a user, I can get the appropriate fertilizers for the disease predicted.	10	High

6.2 SPRINT DELIVERY SCHEDULE

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

6.3 REPORTS FROM JIRA

Velocity:

Sprint 1 average velocity: Average velocity= 20/6 = 3.33

Sprint 2 average velocity: Average velocity= 20/6 = 3.33

Sprint 3 average velocity: Average velocity= 20/6 = 3.33

Sprint 4 average velocity: Average velocity= 20/6 = 3.33

Burndown chart:

CHAPTER 7

CODING & SOLUTIONING

7.1 FEATURE -1

PREDICTING THE DISEASE OF AFFECTED LEAVES:

FRUIT LEAF DISEASE PREDICTION:

Importing the required libraries:

To build the model, required libraries or packages need to be imported. This model is being trained in IBM cloud service where all these packages are imported and used.

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

Image Preprocessing and Data Augmentation:

The aim of pre-processing is an improvement of the image data that suppresses unwilling distortions or enhances some image features important for further processing. Since we are using image to predict the disease of the leaf, the image has to be preprocessed before passing to the model. Rescaling, zooming, horizontal flipping and shear range on the image has to be applied. Now to classify the diseases based on the category, image preprocessing needs to be done on the dataset and splitted as training and testing dataset. Then keys or diseases list have been obtained through class_indices.keys().

```
train_data=ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_range=0.2,
horizontal_flip=True)

test_data =ImageDataGenerator (rescale = 1)

X_train=train_data.flow_from_directory(r'D:\Clg-studies\sem7\IBM\IBM-Project-
9509-1659014053\Project Development Phase\Sprint 1\Data Collection\Dataset
Plant
Disease\fruit-dataset\train',target_size=(128,128),batch_size=32,class_mode="cate
gorical")

X_test=test_data.flow_from_directory(r'D:\Clg-studies\sem7\IBM\IBM-Project-95
09-1659014053\Project Development Phase\Sprint 1\Data Collection\Dataset Plant
Disease\fruit-dataset\test',target_size=(128,128),batch_size=32,class_mode="categ
orical")

X_train.class_indices.keys()

X_test.class_indices.keys()
```

Initializing The Model:

Model has been initialized in the variable. To the initialized variable, convolution layer(CNN), relu activation and fixing the image input to 128 * 128px has been added. Pooling the layer has also been done. The purpose of the pooling layers is to reduce the dimensions of the hidden layer by combining the outputs of neuron clusters at the previous layer into a single neuron in the next layer. Then it has been flattened. Flattening is used to convert all the resultant 2-Dimensional arrays from pooled feature maps into a single long continuous linear vector. The flattened matrix is fed as input to the fully connected layer to classify the image. The final layers in a CNN are fully (densely) connected layers. In Keras, these layers are created using the Dense() class. The Multilayer Perceptron (MLP) part in a CNN is created using multiple fully connected layers. In Keras, a fully connected layer is referred to as a Dense layer. After this model has been compiled with loss and metrics. Validation of the model has also been done to validate the model using epoch.

```
model=Sequential()
Add CNN Layers
model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))
#pooling layer
model.add(MaxPooling2D(pool_size=(2,2)))
#flatten layer
model.add(Flatten())
model.summary()
Model: "sequential"
Add Dense Layers
#dense layers
model.add(Dense(300,kernel_initializer='uniform', activation = 'relu'))
model.add(Dense(250,kernel_initializer = 'random_uniform', activation = 'relu'))
model.add(Dense(6,kernel_initializer = 'random_uniform', activation = 'softmax'))
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=["accuracy"])
model.fit_generator(X_train,steps_per_epoch=len(X_train),epochs=12,validation_data=X_test,validation_steps=len(X_test))
model.fit_generator(X_train,steps_per_epoch=len(X_train),epochs=12,validation_data=X_test,validation_steps=len(X_test))
```

Saving the model:

After doing all the prerequisites, the model has been saved to integrate and use it in the application.

```
model.save("fruit_training.h5")
```

Same steps have been carried out to build the vegetable model.

7.2 FEATURE -2

After getting the disease name of the affected leaf which has been imputed from the web application, precautionary steps have been displayed. It will suggest the chemical combinations and treatments that are to be taken. This feature is added in the integration with great user interface and user experience. Precautions are stored in the xlsx file. If the specific disease has been identified, precautions required for the particular disease should have been retrieved from the xlsx file based on the index.

For instance, if the disease is identified as Pepper,_bell___Bacterial_spot, the precaution of the disease will be retrieved as “Oops!! Your pepper plant is infected by Bacterial Leaf Spot. The disease cycle can be stopped by using the Sango formula for disinfectants. Bleach treatment and hot water treatment is also helpful”.

```
if(family == 'vegetables'):
        preds=np.argmax(vegmodel.predict(x),axis=1)
        print(preds)
        df = pd.read_excel('precautions - veg.xlsx')
        print(df.iloc[preds[0]]['caution'])
        disease = veg_index[preds[0]]
    else:
        preds = np.argmax(fruitmodel.predict(x),axis=1)
        print(preds)
        df = pd.read_excel('precautions - fruits.xlsx')
        print(df.iloc[preds[0]]['caution'])
        disease = fruit_index[preds[0]]
```

CHAPTER 8

TESTING

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisit	Steps To Execute	Test Data	Expected Result	Actual Result	Status
LoginPage_TC_001	Functional	Home Page	Verify user is able to see the Login/Signup popup when user clicked on My account button	Check if the login button works	Verify login/Singup popup displayed or not		Login/Signup popup should display	Working as expected	Pass
HomePage_TC_002	UI	Home Page	Verify user is able to upload the images		Capturing the images of the leaf from multiple angles		Capturing of images according to the suggested formats	Working as expected	Pass
Homeage_TC_003	Functional	Image Uploading	Verify model is able to preprocess the uploaded images		Filtering out diseased leaves from the uploaded leaf images.		Uploaded images are processed	Working as expected	Pass
PredictPage_TC_004	Functional	Image Processing	Verify if the model is able to predict the diseases accurately		Type of disease present in the leaf is identified		Application should show the predicted diseases	Working as expected	Pass

CHAPTER 9

RESULTS

9.1 PERFORMANCE METRICS

1. MODEL SUMMARY

CNN Layers:

```
model.summary()

Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_1 (Conv2D)           (None, 126, 126, 32)      896

 max_pooling2d_1 (MaxPooling  (None, 63, 63, 32)       0
 2D)

 flatten_1 (Flatten)         (None, 127008)            0

=================================================================
Total params: 896
Trainable params: 896
Non-trainable params: 0
_________________________________________________________________
```

Dense Layers:

```
model.summary()
```

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_1 (Conv2D)           (None, 126, 126, 32)      896

 max_pooling2d_1 (MaxPooling  (None, 63, 63, 32)       0
 2D)

 flatten_1 (Flatten)         (None, 127008)            0

 dense_20 (Dense)            (None, 300)               38102700

 dense_21 (Dense)            (None, 150)               45150

 dense_22 (Dense)            (None, 75)                11325

 dense_23 (Dense)            (None, 9)                 684

=================================================================
Total params: 38,160,755
Trainable params: 38,160,755
Non-trainable params: 0
_________________________________________________________________
```

2.ACCURACY

TRAINING ACCURACY

```
print("Accuracy=",results[1]*100)
```

```
1/1 [==============================] - 0s 39ms/step
Expected: Potato___Early_blight
 Predicted: Potato___Early_blight
Found 3416 images belonging to 9 classes.
50/50 [==============================] - 4s 83ms/step - loss: 2019.1171 - accuracy: 0.3660
Accuracy= 36.59999966621399
```

Vegetable-model-training_Accuracy

```
print("Accuracy=",results[1]*100)
```

```
Found 1686 images belonging to 6 classes.
64/64 [==============================] - 5s 76ms/step - loss: 1231.0060 - accuracy: 0.6250
Accuracy= 62.5
```

Fruit-model-training_Accuracy

VALIDATION ACCURACY

```
Epoch 1/12
169/169 [==============================] - 96s 565ms/step - loss: 0.0780 - accuracy: 0.9733 - val_loss: 1158.5586 - val_accuracy: 0.6311
Epoch 2/12
169/169 [==============================] - 90s 534ms/step - loss: 0.0568 - accuracy: 0.9803 - val_loss: 1336.4701 - val_accuracy: 0.5890
Epoch 3/12
169/169 [==============================] - 115s 681ms/step - loss: 0.0476 - accuracy: 0.9825 - val_loss: 2344.9492 - val_accuracy: 0.4727
Epoch 4/12
169/169 [==============================] - 134s 794ms/step - loss: 0.0458 - accuracy: 0.9848 - val_loss: 2427.1777 - val_accuracy: 0.4822
Epoch 5/12
169/169 [==============================] - 165s 973ms/step - loss: 0.0844 - accuracy: 0.9749 - val_loss: 785.4515 - val_accuracy: 0.6168
Epoch 6/12
169/169 [==============================] - 153s 904ms/step - loss: 0.0527 - accuracy: 0.9818 - val_loss: 1581.9740 - val_accuracy: 0.6044
Epoch 7/12
169/169 [==============================] - 94s 554ms/step - loss: 0.0422 - accuracy: 0.9881 - val_loss: 1144.1666 - val_accuracy: 0.6376
Epoch 8/12
169/169 [==============================] - 111s 656ms/step - loss: 0.0497 - accuracy: 0.9831 - val_loss: 973.7907 - val_accuracy: 0.6151
Epoch 9/12
169/169 [==============================] - 176s 1s/step - loss: 0.0349 - accuracy: 0.9879 - val_loss: 1441.4517 - val_accuracy: 0.5955
Epoch 10/12
169/169 [==============================] - 171s 1s/step - loss: 0.0516 - accuracy: 0.9840 - val_loss: 1224.8441 - val_accuracy: 0.5510
Epoch 11/12
169/169 [==============================] - 159s 940ms/step - loss: 0.0408 - accuracy: 0.9868 - val_loss: 2179.4436 - val_accuracy: 0.4976
Epoch 12/12
169/169 [==============================] - 184s 1s/step - loss: 0.0416 - accuracy: 0.9866 - val_loss: 1186.5657 - val_accuracy: 0.6358
```

CHAPTER 10

ADVANTAGES AND DISADVANTAGES

10.1 ADVANTAGES

Traditionally, people used to guess the plant disease by visually examining symptoms such as curling of leaves and change of colour. But with the help of scientific advancements, this application helps people to get more clarity regarding the diseases and an accurate fertilizer recommendation.

As a result, farmers especially do not need significant knowledge on all the diseases and fertilizers. They can make use of the application in order to determine the appropriate and effective fertilizers for their affected crops. Essentially, farmers can anticipate any problems or abnormalities and obtain the results online instantly. Moreover, financial losses due to widespread infections leading to poor crop yield can be reduced effectively.

The application mainly focuses on helping farmers who need a recommendation on usage of best fertilizers for the predicted disease on their crops thereby helping them improve the crop yield. The spread of diseases due to improper guidance can be avoided by early identification of these diseases and usage of appropriate fertilizers.

10.2 DISADVANTAGES

Firstly, the Deep Learning model requires extremely high time complexities due to the enormous amounts of data utilized by the model for training and testing purposes. As a result, these time complexities affect the performance of the model on training and validation datasets.

Secondly, the application requires the availability of a good quality internet connection for obtaining results, which is yet another struggle in a developing country like India in remote areas. Moreover, the images of the crop should be captured using a device that meets the availability of a minimum suggested pixels in order to capture the images of the affected crop with greater clarity for accurate identification.

Lastly, this application is built basically for identifying a set of diseases targeting fruits and vegetables having 6 classes and 9 classes respectively. However, the sky's the limit to such diseases that farmers may or may not have a fair knowledge of. Thus handling various classes of diseases in both fruits and vegetables is an extremely challenging task that takes a prolonged period in order to collect training and validation data and subsequently train and test the model.

CHAPTER 11

CONCLUSION

The proposed model employs Deep Learning techniques in order to identify diseases observed both in fruits and vegetables and suggest appropriate fertilizers that can be taken for those diseases. In this image classification problem, during model training it was observed that increasing the convolutional layers as well as the Dense layers resulted in significant improvement in accuracies during evaluation. However, the time complexities are extremely high and may take up to a few hours for model training and testing purposes.

The model trained in IBM Watson Studio cloud platform using Machine Learning Client was observed to produce results with greater accuracies during model training in addition to relatively better time complexities compared to model built and trained in the local system using Anaconda. Subsequently, the trained model is integrated with the web application using a light-weight framework that is open to all the end users.

CHAPTER 12

FUTURE SCOPE

The proposed model is deployed in a web application, which can be extended to a mobile application in both iOS and Android platforms. The application is built such that offline usage is supported since the internet connection may not be reliable in all situations.

Additionally helpline support for resolving app related issues can address problems and challenges commonly faced by farmers. Service availability depends on the plans devised and subscribed by the farmers. Moreover, the services provided can be used on a large or small scale, making it practical for farmers as well as common people. Future diseases that are found and the preventative fertilizer for them can easily be incorporated into the current model, making it highly scalable.