Somaiya Vidyavihar University

Crop Disease Detection and Soil Analysis

Mini-Project by

Shreeraj Rane - 16010420086

Surin Shah - 16010420087

Mayur Bhore - 16010420090

Shreyas Panchikattil - 16010420096

Guide

Prof. Chirag Desai

Department of Information Technology

K.J. SOMAIYA COLLEGE OF ENGINEERING, VIDYAVIHAR, MUMBAI-77

(A Constituent College of Somaiya Vidyavihar University)

Certificate

This is to certify that the report entitled **Crop Disease Detection and Soil Analysis** is bonafide record of Mini-Project work done by **Shreeraj Rane, Surin Shah, Mayur Bhore, Shreyas Panchikattil** in the Sem VI, year 2023 under the guidance of **Prof. Chirag Desai** of Department of Information Technology in partial fulfilment of requirement for the completion of Mini-Project.

Guide

Head of the Department

Date: 16/05/2023

Certificate of Approval of Examiners

We certify that this report entitled **Crop Disease Detection and Soil Analysis** is bonafide record of Mini-Project work done by **Shreeraj Rane, Surin Shah, Mayur Bhore, Shreyas Panchikattil**. This project is approved for the award of credits for completing Mini-Project course.

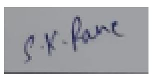
Internal Examiner

External Examiner

Date: 16/05/2023

DECLARATION

We declare that this written report submission represents the work done based on our and / or others' ideas with adequately cited and referenced the original source. We also declare that we have adhered to all principles of intellectual property, academic honesty and integrity as we have not misinterpreted or fabricated or falsified any idea/data/fact/source/original work/ matter in our submission. We understand that any violation of the above will be cause for disciplinary action by the college and may evoke the penal action from the sources which have not been properly cited or from whom proper permission is not sought.

 **Signature of Student** **16010420086** **Roll No:**	 **Signature of Student** **16010420087** **Roll No:**
 **Signature of Student** **16010420090** **Roll No:**	 **Signature of Student** **16010420096** **Roll No:**

Date: 16/05/2023

Contents

1 Introduction……………………………………………………………………... 7

1.1 Problem Definition…………………………………………………….….. 7

1.2 Motivation …………………………………………………………………... 7

1.3 Scope of Project ……………………………………………………………….. 8

1.4 Functional and Non-Functional Requirements …………………………... 9

2 Background Work……………………………………………………………. 11

3 Implementation 13

3.1 Technologies used ……………………………………………………… 13

3.2 Algorithm / Methodology ……………………………………………… 14

3.3 Implementation ………………………………………………………… 15

4 Results and Discussion 17

5 Conclusions and Learnings…………………………………….……………… 31

5.1 Conclusions………………………………………………………………. 31

5.2 Scope for Future Work…………………………………………………. 31

5.3 Learnings…………………………………………………………………. 32

References ……………………………………………………………………...... 33

Acknowledgement ……………………………………………………………...... 35

List of Tables

3.1 Sequence Diagram of the implementation……………………………………….. 34

List of Figures

Figure 4.1: Login/Signup Page……………………………………………………….….23

Figure 4.2:Login/Signup Page Part-2………..……………………………………………...23

Figure 4.3: Login/Signup Page Part 3………………………………………………………24

Figure 4.4 Home Page Part-1…………..…………………………………………………...24

Figure 4.5: Home Page Part-2………..……………………………………………………..25

Figure 4.6: Home Page Part-3……………………………………………………………....25

Figure 4.7: About Crop/Diseases Page ………………………………………………………26

Figure 4.8: About Soil Analysis Page ……………………………………………….…..26

Figure 4.9: Uploading a crop image for diagnosis…………………………………….……27

Figure 4.10: Diagnosis Result Part-1..………………………………………………………28

Figure 4.11: Diagnosis Result Part-2..………………………………………………………28

Figure 4.12: Soil Analysis Page…..…………………………………………………...…29

Figure 4.13: Soil Analysis Page after giving input location…………………………………29

Figure 4.14: Predicting the crop suitable according to the location...………… …….………30

Chapter 1

Introduction

1.1 Problem Definition:

This project aims to develop a computer vision-based website that can automate the identification of diseases in maize cultivation. The objective is to provide farmers with an efficient and accurate tool to protect their crops and minimize potential losses. Additionally, the project proposes the use of machine learning algorithms to estimate agricultural output, which can assist farmers in selecting suitable crops based on soil type and weather conditions. The website aims to provide a user-friendly portal that offers farmers a onestop-shop for information about crop selection and soil type, facilitating informed decision-making. Ultimately, the project aims to improve crop yield and increase profitability for farmers by providing them with advanced technology and data-driven insights.

1.2 Motivation:

The motivation behind choosing the topic of developing a computer vision-based website to identify diseases in maize cultivation and utilizing machine learning algorithms to estimate agricultural output is to address the significant challenges faced by maize farmers worldwide. Maize is a vital crop, and its cultivation is essential to ensuring food security for millions of people worldwide. However, diseases can significantly impact maize crops, resulting in lower yield and economic losses. The current disease detection methods often rely on visual inspection by trained personnel, which can be time-consuming, expensive, and prone to human error. Additionally, farmers may not have access to timely and accurate information about crop selection and soil type, leading to suboptimal yields and lower profits.

The development of a computer vision-based website that can automate the identification of diseases in maize cultivation and utilize machine learning algorithms to estimate agricultural output has the potential to revolutionize maize farming practices. The website can provide farmers with accurate and timely information, enabling them to take proactive measures to protect their crops and make informed decisions about their agricultural practices. By doing so, farmers can increase their crop yields, profitability, and ensure food security for millions of people worldwide.

Overall, this project has significant implications for the agriculture industry, and its successful implementation can significantly benefit farmers, consumers, and society as a whole.

1.3 Scope of Project:

The scope of this project is to develop a computer vision-based website that can automate the identification of diseases in maize cultivation and utilize machine learning algorithms to estimate agricultural output. The website will provide farmers with a user-friendly portal that offers information about crop selection and soil type, facilitating informed decision-making.

The primary focus of the project is to train a computer vision model to accurately identify diseases in maize plants. This will involve collecting and labeling a large dataset of images of healthy and diseased maize plants. The model will then be integrated into the website, allowing farmers to upload images of their crops and receive automated disease identification results.

The project will also involve the use of machine learning algorithms to estimate agricultural output based on soil type and weather conditions. This will require collecting and analyzing data on various factors that can affect crop yield, such as soil type, temperature, precipitation, and sunlight.

The website's design will be another crucial aspect of the project, as it will need to be user-friendly and accessible to farmers with varying levels of technological proficiency. The website will need to allow farmers to upload images, receive disease identification results, and access data-driven insights about crop selection and soil type.

Overall, the scope of this project is ambitious, and it will require a significant amount of effort and resources to complete successfully. However, the potential benefits to the agriculture industry and society as a whole make this project worthwhile.

1.4 Functional and Non-Functional Requirements

1.4.1 Functional Requirements

1) Disease Detection and Diagnosis: The website should be able to analyze images of crops and accurately identify the presence of diseases.
2. Customized Treatment Options: The website should be able to provide information on the type of disease identified and recommend appropriate treatment options.
3. Smarter Crop Selection: The website should provide farmers with information on crop selection based on soil type and weather conditions.
4. Predictive Yield Analysis: The website should be able to estimate agricultural output based on various factors such as weather, soil type, and crop characteristics.

1.4.2 Non-Functional Requirements:

1. Security and Privacy: The website should be secure and protect the privacy of the users.
2. Scalability and Performance: The website should be able to handle large volumes of data and user requests without experiencing downtime or performance issues.
3. Cross-Platform Compatibility: The website should be accessible from different devices and platforms.

4. Speed and Responsiveness: The website should be fast and responsive to ensure a seamless user experience.
5. Easy Maintenance: The website should be easy to maintain and update, with minimal downtime required for maintenance activities.
6. User-Friendly Design: The website should provide a user-friendly interface for farmers to access and interact with the different features and functionalities.

Chapter 2

Background Work

We researched three research papers each on Crop Disease Detection and Soil Analysis. The First paper by E. Akanksha, N. Sharma and K. Gulati.[1] Proposes a method which uses an optimized probabilistic neural network (OPNN) to classify the maize plant images into healthy or diseased categories. The OPNN model is optimized using a particle swarm optimization (PSO) algorithm to improve its accuracy. Experimental results demonstrate that the proposed OPNN model achieves high accuracy in detecting maize plant diseases, outperforming traditional machine learning algorithms.

The Next paper by S. K. Tichkule and D. H. Gawali [2] provides a survey of various methods used for plant leaf disease detection using image processing techniques. The k-nearest-neighbor method is simple but slow, while neural networks are tolerant to noise but have complex structures. Support vector machines (SVM) were found to be competitive with other machine learning algorithms for classifying high-dimensional data sets.

The last paper by X. Zhang, Y. Qiao, F. Meng, C. Fan and M. Zhang on disease detection [3] proposes an improved CNN model to identify maize leaf diseases. The model consists of convolutional and pooling layers, fully connected layers, and a soft-max classifier. To enhance performance, techniques like data augmentation, dropout regularization, and transfer learning were applied. Experimental results show that the proposed model achieved an accuracy of 96.5%, outperforming several deep learning and traditional machine learning models.

The next research paper on Soil Analysis [4] talks about the Soil nutrient content, including nitrogen, phosphorus, and potassium were estimated using near-infrared (NIR) spectroscopy and machine learning algorithms. The authors achieved good prediction accuracy using partial least squares regression (PLSR) and random forest algorithms.

The next paper by Jagdeep Yadav, Shalu Chopra and Vijayalakshmi M [5] proposes a model for predicting soil fertility, crop prediction, and crop yield based on soil and crop datasets from the Indian region using machine learning algorithms. The model was tested using different algorithms and found that the Artificial Neural Network (ANN) had the highest accuracy in all three predictions.

The last paper by S. Prakash, A. Sharma and S. S. Sahu [6] presents a study on the use of machine learning methods to predict soil moisture in advance. The study utilized multiple linear regression, support vector regression models for the prediction. The results showed that multiple linear regression outperformed support vector regression and recurrent neural network.

Chapter 3

Implementation

3.1 Technologies Used:

1. **Frontend Technology: React**

An open-source JavaScript package called React is used to create user interfaces. Its capacity to produce dynamic and responsive user interfaces led to its selection as the frontend technology for this system of crop disease diagnosis. The frontend of the system was made to be user-friendly, with simple forms and plain visualisations of the outcomes. React was a smart choice for this project because it made frontend development and deployment speedy.

2. **Backend Technology: Flask**

A simple web framework called Flask is used to create web apps in Python. Due to its ease of use and adaptability, Flask was selected as the backend technology for this crop disease detection system. The backend of the system was created to manage user input, interact with the machine learning model, and return the findings to the frontend. Flask was a perfect fit for this project because of its lightweight design, which made it possible to process requests quickly.

3. **Deep Learning Technology: EfficientNetB2**

EfficientNetB2 is a convolutional neural network model used for image classification tasks. Due to its great accuracy and efficiency, it was selected as the machine learning technique for this crop disease detection system. The programme can precisely identify numerous crop diseases because it was trained on a vast sample of crop photos. TensorFlow, a deep learning framework, was used to construct EfficientNetB2, making it simple to integrate with the Flask backend.

4. **Machine Learning Technology: Gradient Boosting using Skicit Learning**

   The Gradient Boosting Algorithm is widely used in crop recommendation systems because it effectively handles complex relationships, outliers, and missing data. It captures intricate patterns, provides robust predictions, and estimates feature importance. Its scalability and efficiency make it suitable for analyzing large agricultural datasets. By leveraging these capabilities, crop recommendation systems can offer accurate guidance to farmers, helping them make informed decisions and optimize crop yields.

5. **Coding: VS Code**

   Visual Studio Code is a source-code editor that can be used with a variety of programming languages, including Java, JavaScript, Go, Node.js, Python and C++. Visual Studio Code also ships with IntelliSense for JavaScript, TypeScript, JSON, CSS, and HTML, as well as debugging support for Node.js. Support for additional languages can be provided by freely available extensions on the VS Code Marketplace. Due to the availability of various extensions, in-built terminal, intelligence, etc. We have used VS Code to increase the speed of development.

3.2 Algorithm / Methodology:

1. **Convolutional Neural Networks (CNN):**

CNN is a type of deep neural network that is widely used for image recognition and classification tasks. Convolutional, pooling, and fully connected layers are only a few of the many layers that make up a CNN's architecture. The input image is subjected to filters by the convolutional layers, which enables the network to learn features at various degrees of abstraction. The fully connected layers carry out the final classification, while the pooling layers lower the dimensionality of the feature maps produced by the convolutional layers.

2. **EfficientNetB2**

EfficientNetB2 is a state-of-the-art convolutional neural network model for image classification tasks. It is a member of the EfficientNet model family, which was created to achieve great accuracy while using little computer power. In addition to having more layers and parameters, the EfficientNetB2 model is wider and deeper than the EfficientNetB0 model. It demonstrated great accuracy on several image classification benchmarks, including ImageNet, after being trained on a sizable collection of images. In computer vision tasks like object identification, segmentation, and classification, the EfficientNetB2 model is frequently utilized. It is a desirable option for real-world applications, especially those with constrained computer resources, because to its high accuracy and efficiency.

3. **Gradient Boost**

Gradient Boosting is a powerful boosting approach that turns numerous weak learners into strong learners. Each successive model is trained using gradient descent to minimize the loss function of the preceding model, such as mean square error or cross-entropy. The algorithm calculates the gradient of the loss function with respect to the current ensemble's predictions in each iteration, and then it trains a new weak model to try to minimize this gradient. The ensemble is then updated with the new model's predictions, and the procedure is continued up until a stopping requirement is satisfied.

3.3 Implementation

1. Data Preparation:

   The preparation of the dataset was the initial stage in using the EfficientNetB2 model for the crop disease detection system. pictures of both healthy and diseased crops were included in the dataset of crop pictures that was compiled. The photos were converted to RGB, their sizes were adjusted to a standard, and the pixel values were normalised as part of the preprocessing of the dataset.

2. Training and Validation:

The crop disease dataset was used to train and test the EfficientNetB2 model. Using backpropagation and gradient descent, the weights of the network's final layer were updated during the training process. The network was assessed on a validation set throughout training in order to track its effectiveness and avoid overfitting. Until the target level of accuracy was reached, the training and validation process was repeated.

3. Testing:

   Following training, the EfficientNetB2 model was put to the test on a collection of cropped photos. In order to make sure that the network's performance was assessed on untrained data, neither training nor validation sets of photos were included in the test set. Several criteria, including accuracy, confusion matrix, were used to assess the effectiveness of the EfficientNetB2 model.

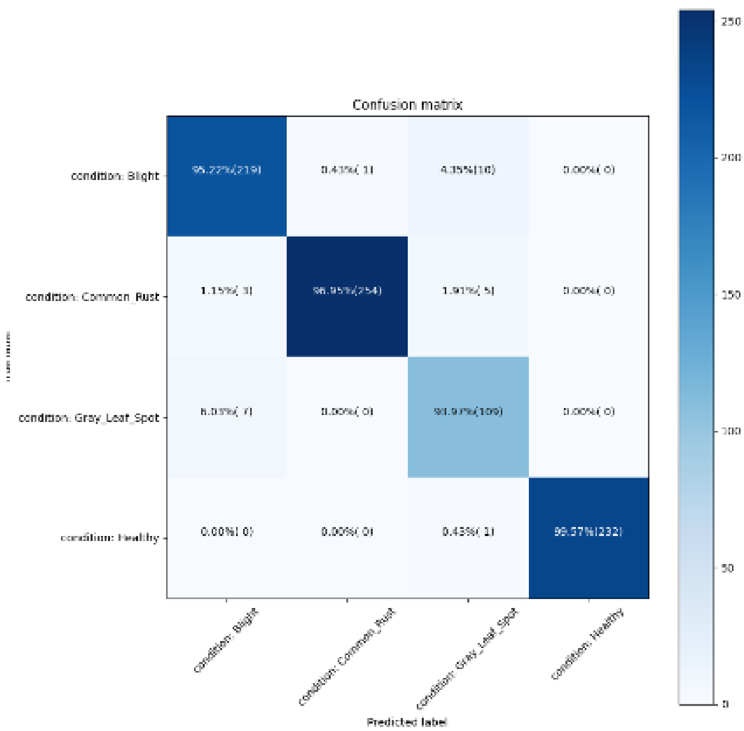
4. Implementation:

   The crop disease detection system implemented the EfficientNetB2 model. The EfficientNetB2 model, a backend, and a frontend made up the system. React was used to build the front end, Flask was used to build the back end, and the back end was integrated with the EfficientNetB2 model. Users can upload pictures of crops and the system will tell them whether they were healthy or infected.

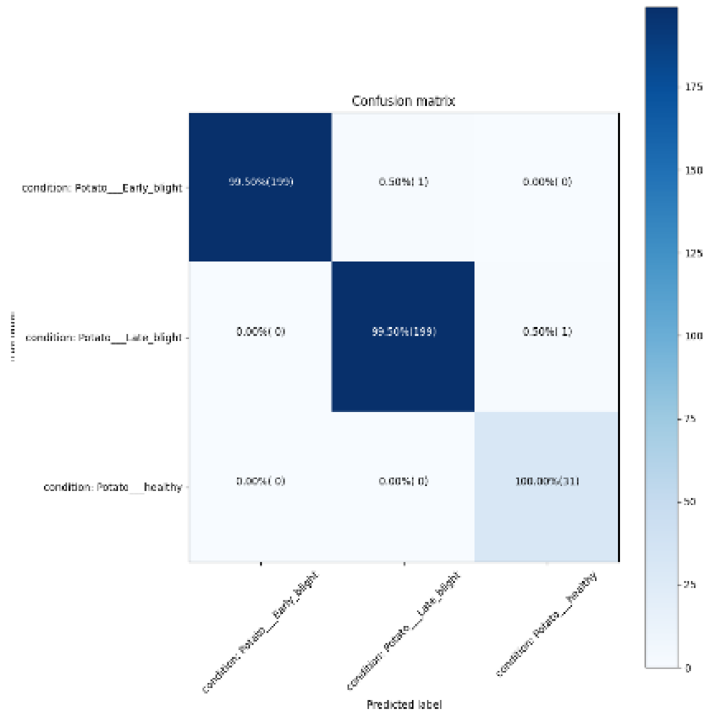
Chapter 4

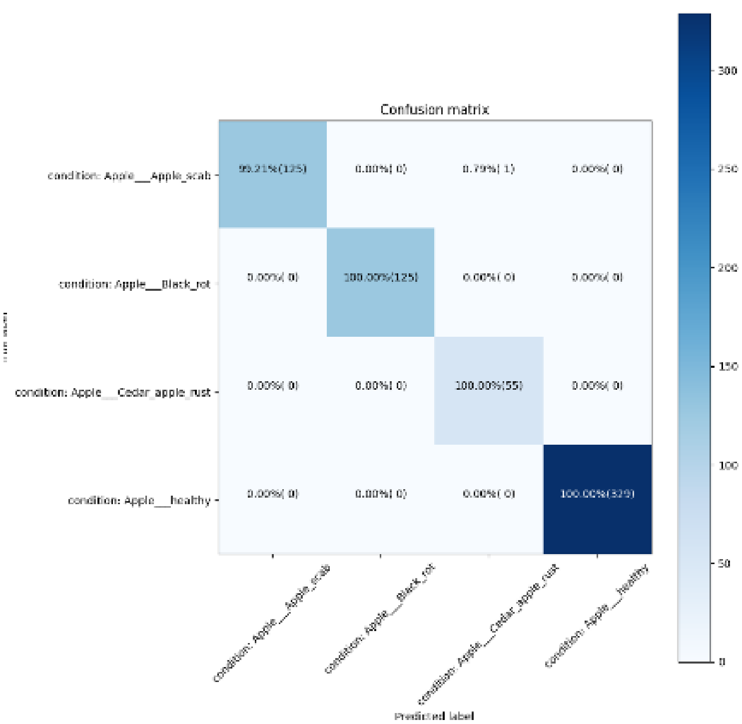
Results and Discussion

Accurate detection of crop diseases: The machine learning model has been taught to identify numerous diseases that affect the maize crop and fruits. To analyze crop imagery and display the results appropriately, we used computer vision. The model's high level of recall and precision enables farmers to make quick corrections.

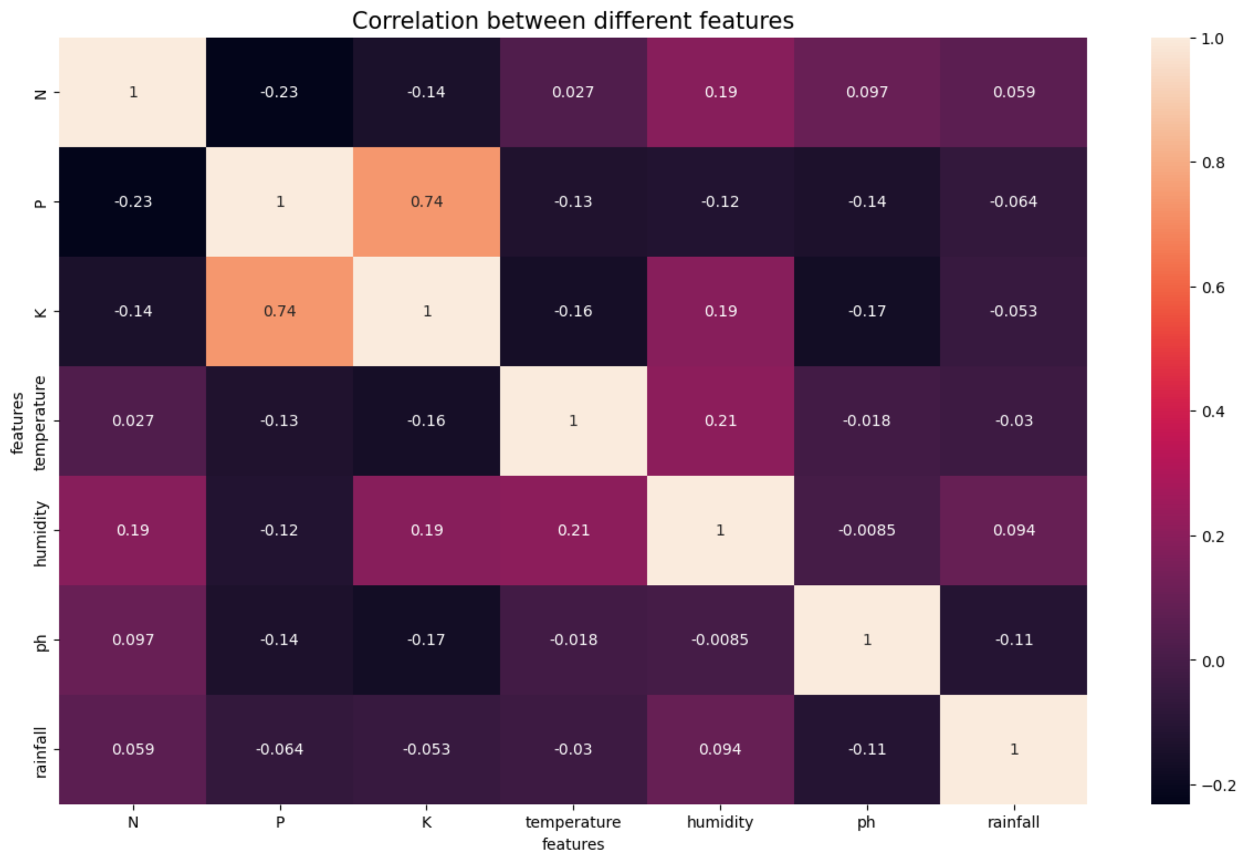


Confusion Matric for Maize





Confusion Matrix For Potato And Apple




```
from sklearn.metrics import accuracy_score
accuracy=accuracy_score(y_pred, y_test)
print('Model accuracy score: {0:0.2f}'.format(accuracy_score(y_test, y_pred)*100))
```

[33] Python

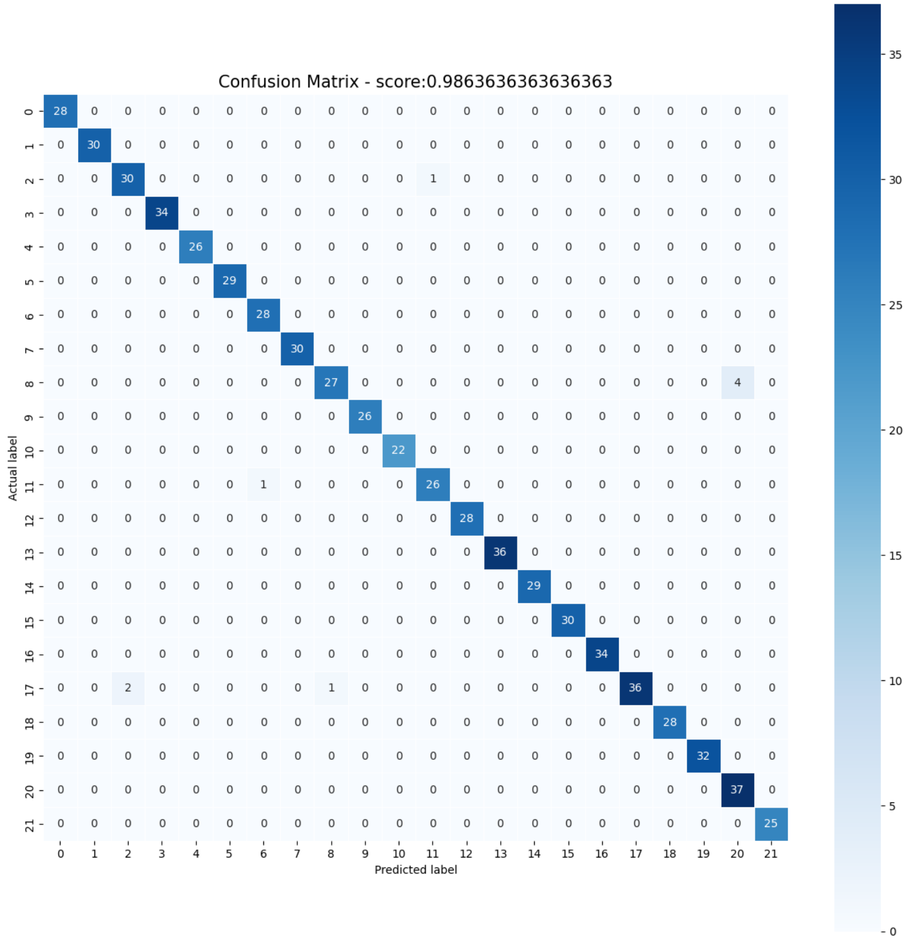
```
Model accuracy score: 98.64
```

```
print('Training set score: {:.4f}'.format(result.score(X_train, y_train)*100))
print('Test set score: {:.4f}'.format(result.score(X_test, y_test)*100))
```

[34] Python

```
Training set score: 100.0000
Test set score: 98.6364
```

Gradient Boost Machine Accuracy



Confusion Matrix For Gradient Boost Machine

	precision	recall	f1-score	support
apple	1.00	1.00	1.00	28
banana	1.00	1.00	1.00	30
blackgram	0.94	0.97	0.95	31
chickpea	1.00	1.00	1.00	34
coconut	1.00	1.00	1.00	26
coffee	1.00	1.00	1.00	29
cotton	0.97	1.00	0.98	28
grapes	1.00	1.00	1.00	30
jute	0.96	0.87	0.92	31
kidneybeans	1.00	1.00	1.00	26
lentil	1.00	1.00	1.00	22
maize	0.96	0.96	0.96	27
mango	1.00	1.00	1.00	28
mothbeans	1.00	1.00	1.00	36
mungbean	1.00	1.00	1.00	29
muskmelon	1.00	1.00	1.00	30
orange	1.00	1.00	1.00	34
papaya	1.00	0.92	0.96	39
pigeonpeas	1.00	1.00	1.00	28
pomegranate	1.00	1.00	1.00	32
rice	0.90	1.00	0.95	37
watermelon	1.00	1.00	1.00	25
accuracy			0.99	660
macro avg	0.99	0.99	0.99	660
weighted avg	0.99	0.99	0.99	660

Model Accuracy

```
[ ] y_pred = knn.predict(Xtest)

[ ] from sklearn.metrics import accuracy_score
    accuracy = accuracy_score(Ytest, y_pred)
    print("Accuracy:", accuracy)

    Accuracy: 0.9772727272727273
```

KNN Model Accuracy

Soil analysis recommendations: The technology can recommend farmers on the best crops to plant on a particular area of land. The advice will be based on a number of factors, including the amount of soil moisture, the air quality index, the pH level, the presence of particular chemicals, etc. Farmers will have a range of options that, given the conditions, are suited for growing a variety of crops.

Improved crop yields and quality: Farmers will be able to maximize crop yields and enhance crop quality with the support of advice from soil analysis and reliable crop disease identification. Farmers will be able to prevent crop diseases and increase soil fertility, resulting in healthier crops with higher yields and better quality.

Map integration for ease of use: The integration of Mapbox API into the project allowed farmers to select the desired piece of land by clicking on the location on the map and obtaining detailed information about the soil and crops that can be grown in that area. With the aid of this feature, farmers now have an easy-to-use interface for gaining access to crucial data that will aid them in choosing the right crop and using the land. Overall, using Mapbox API improved the project's functionality and helped it achieve its goal of encouraging sustainable agricultural practices.

Screenshots:

Login/Signup Page:

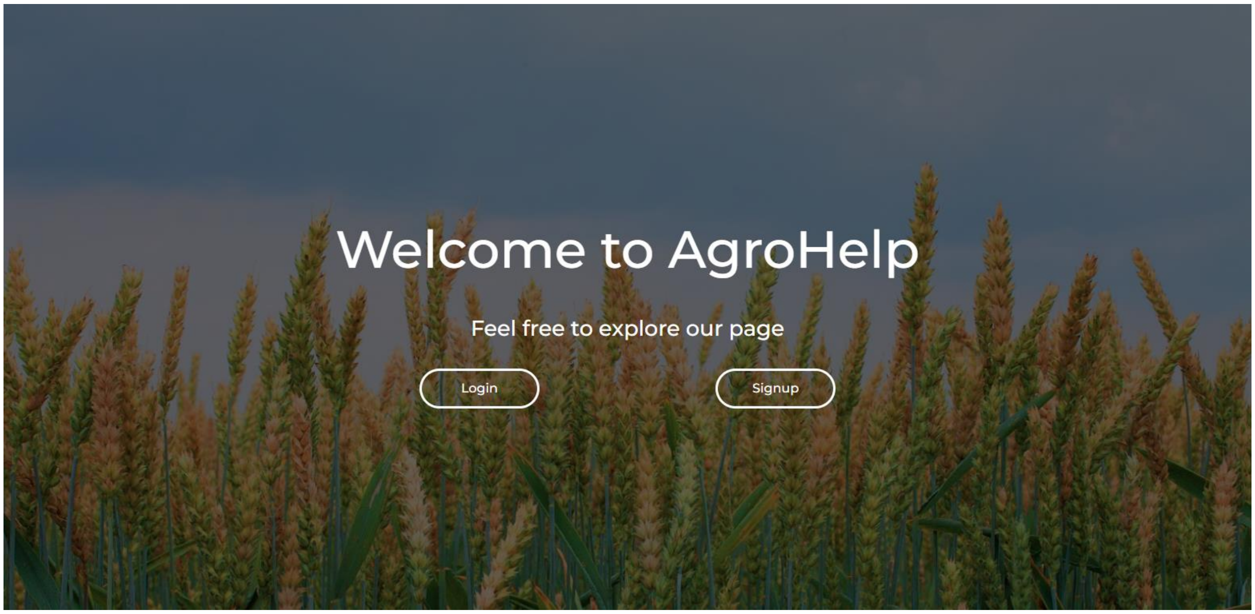
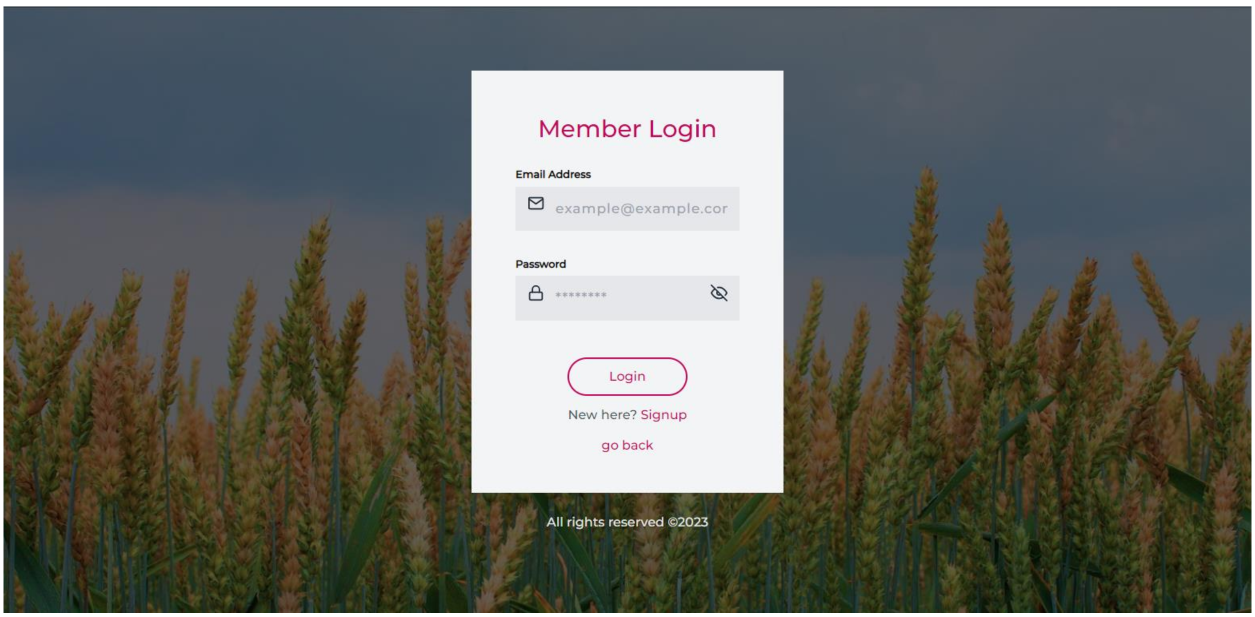


Fig – 4.1 Login/Signup Page Part 1



Fig—4.2 Login/Signup Page Part 2

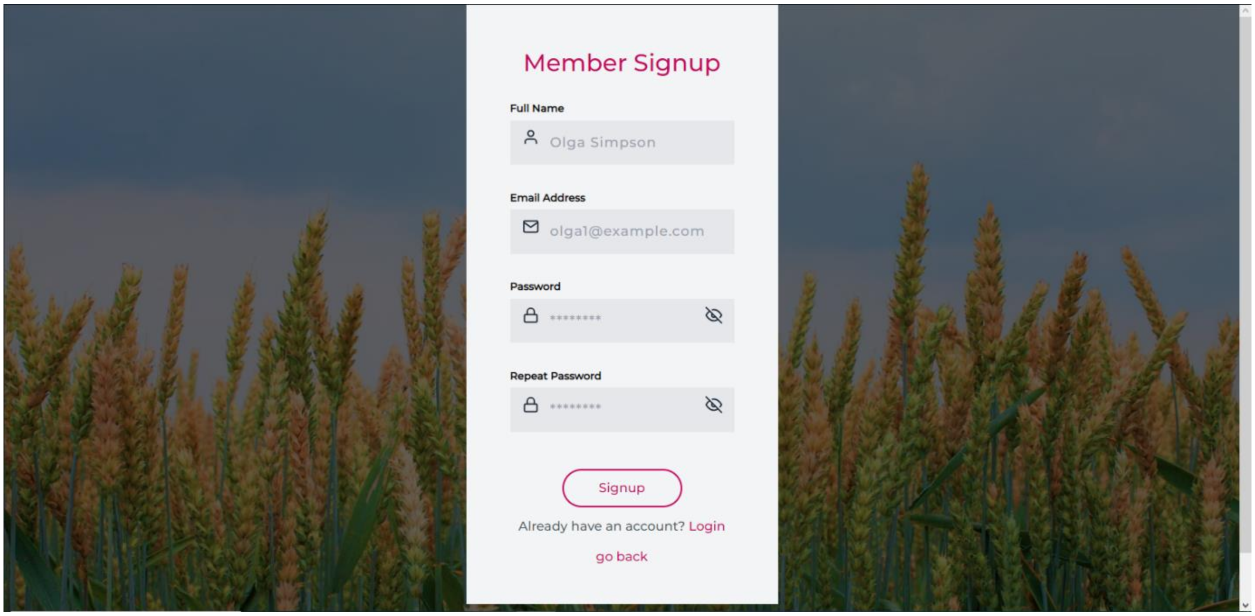


Fig -4.3 Login/Signup Page Part 3

Home Page:

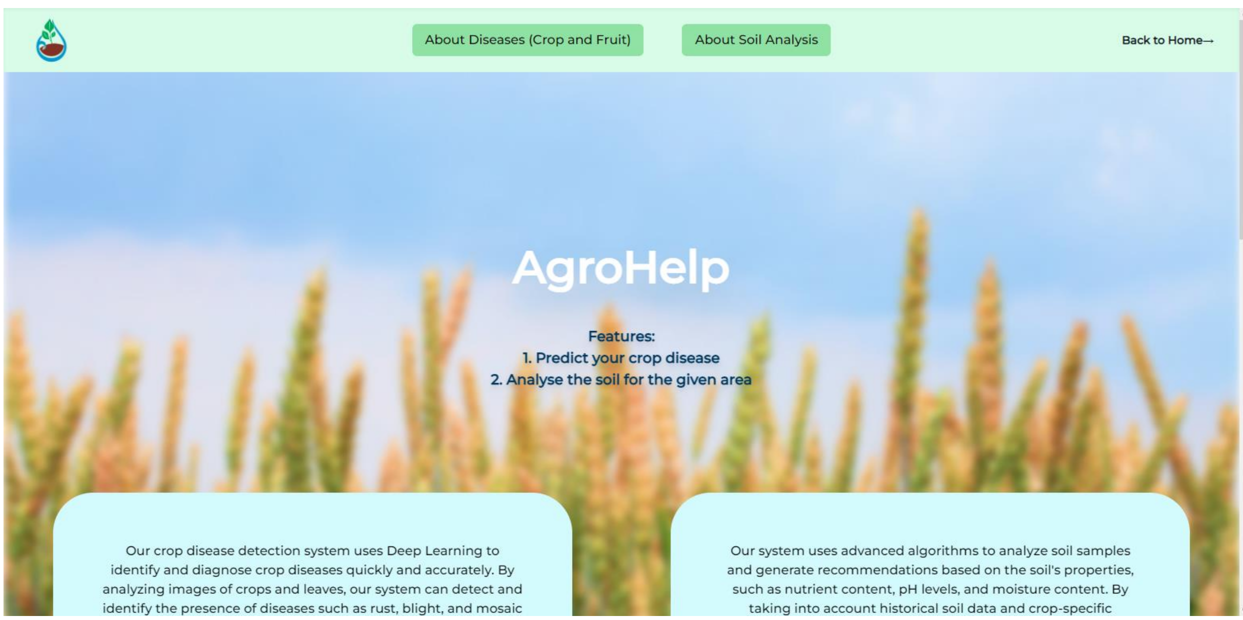


Fig – 4.4 Home Page Part-1

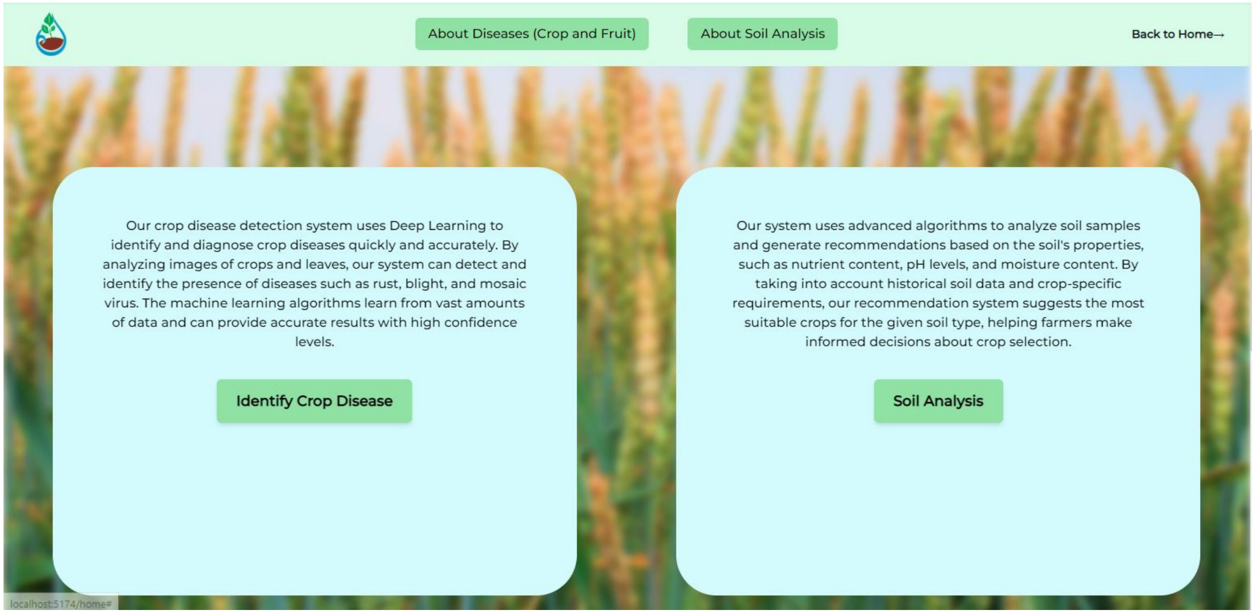


Fig – 4.5 Home Page Part-2

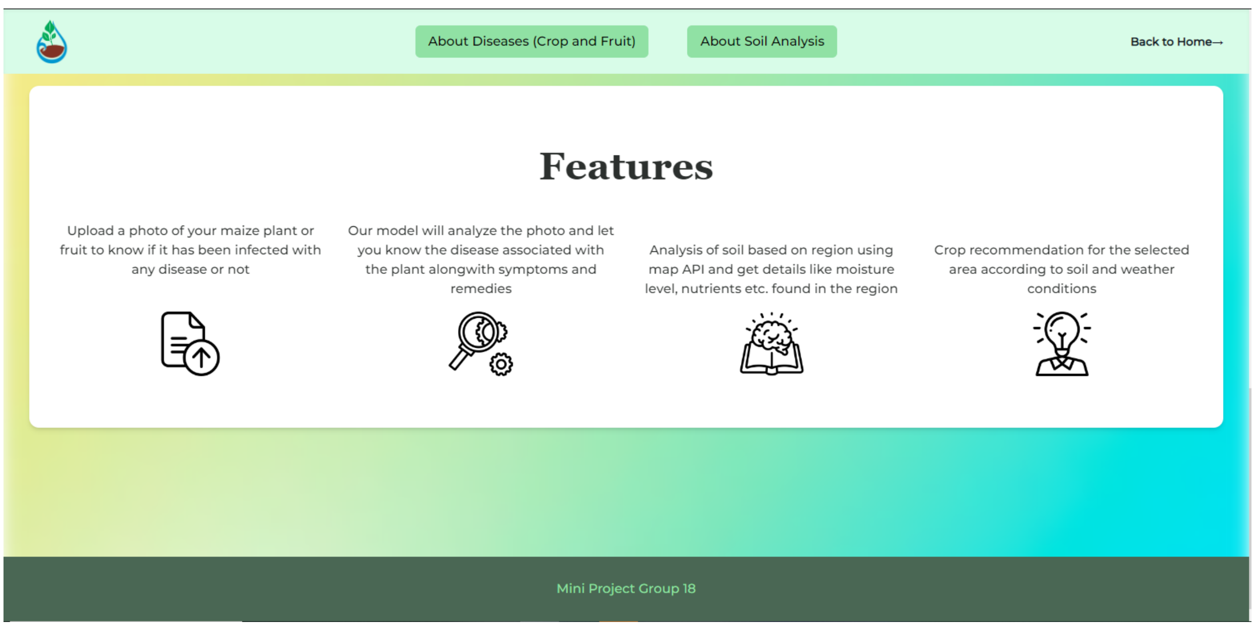


Fig – 4.6 Home Page Part-3

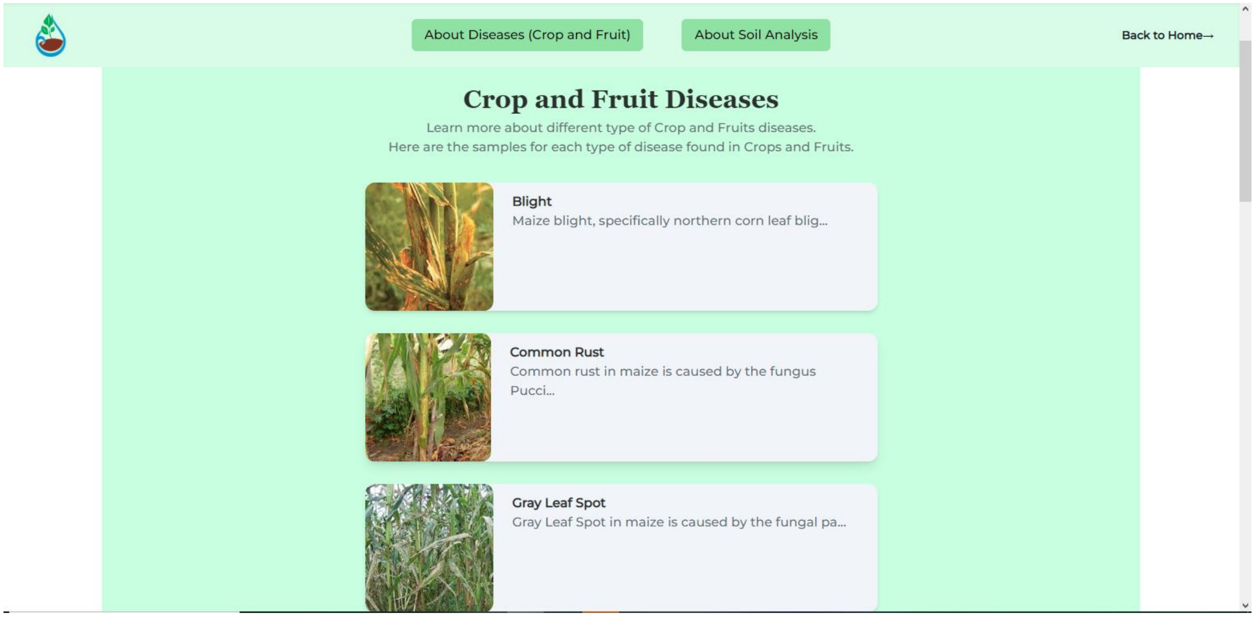


Fig – 4.7 About Crop/Fruit Diseases Page

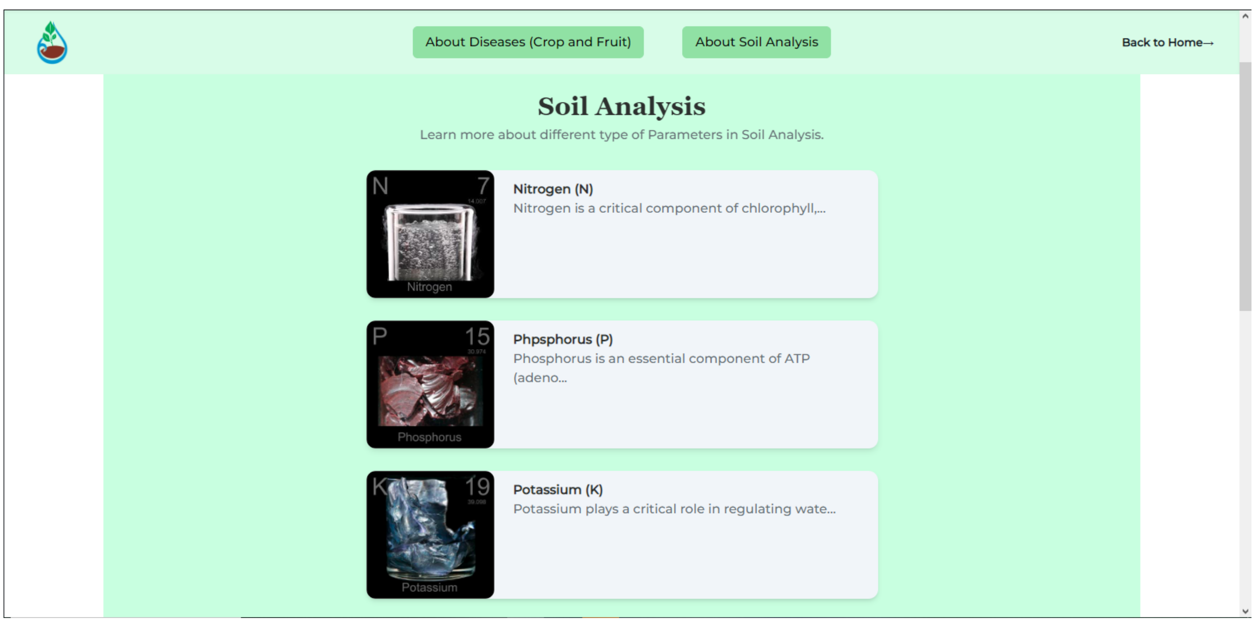


Fig – 4.8 About Soil Analysis Page

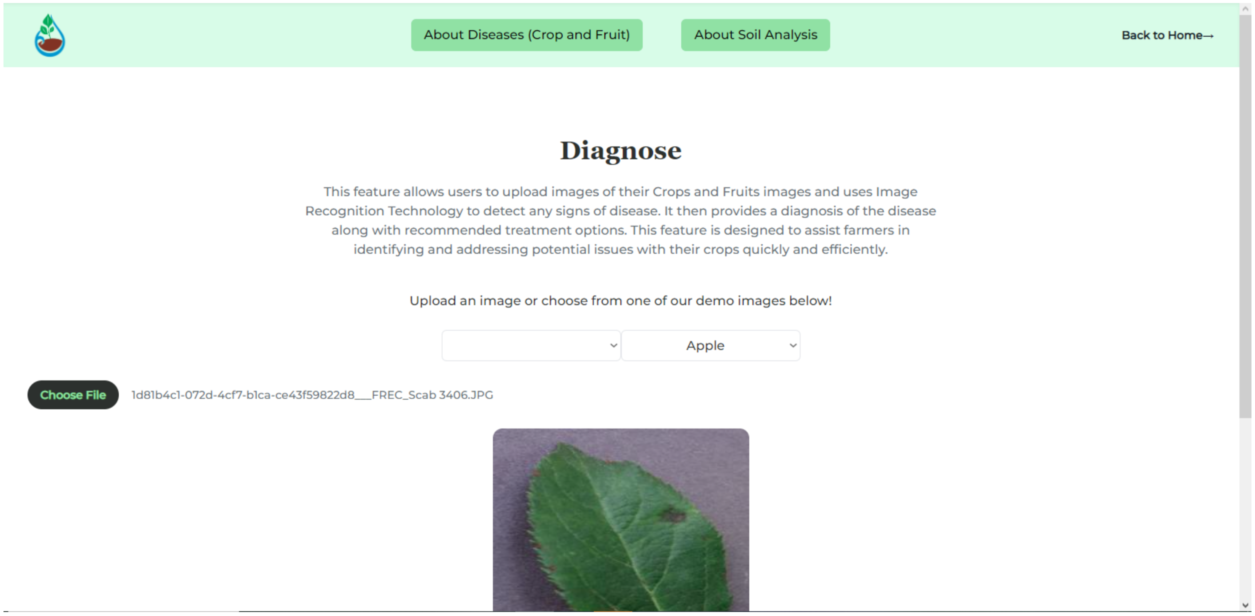
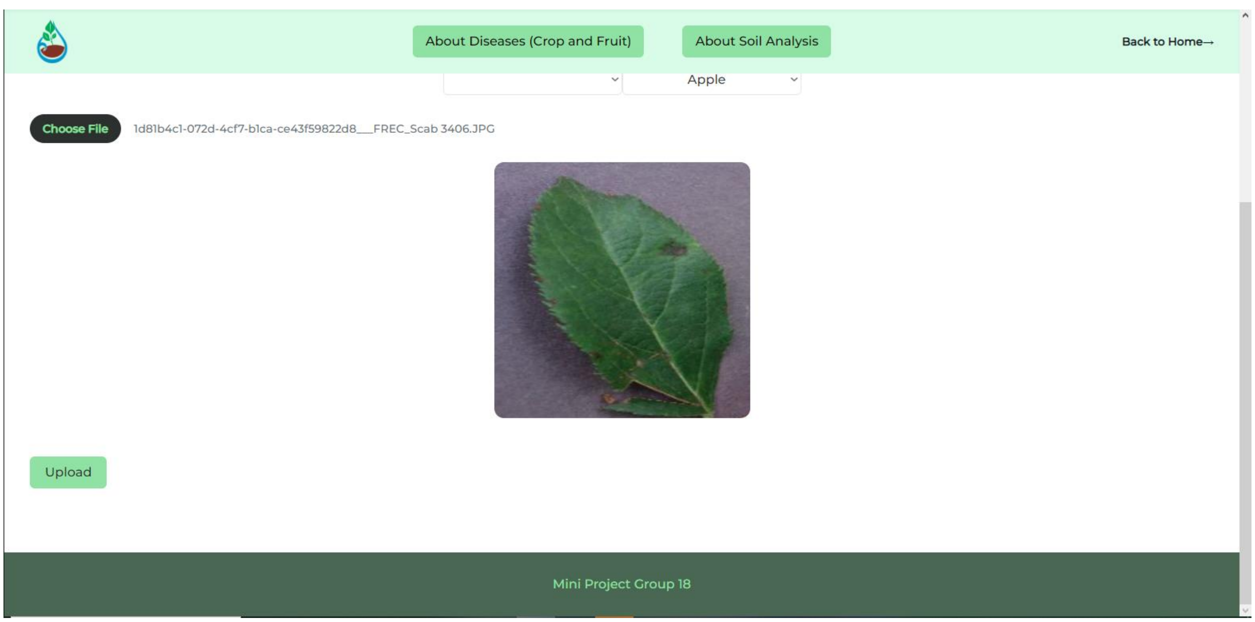


Fig – 4.9 Uploading an infected crop image for Diagnosis



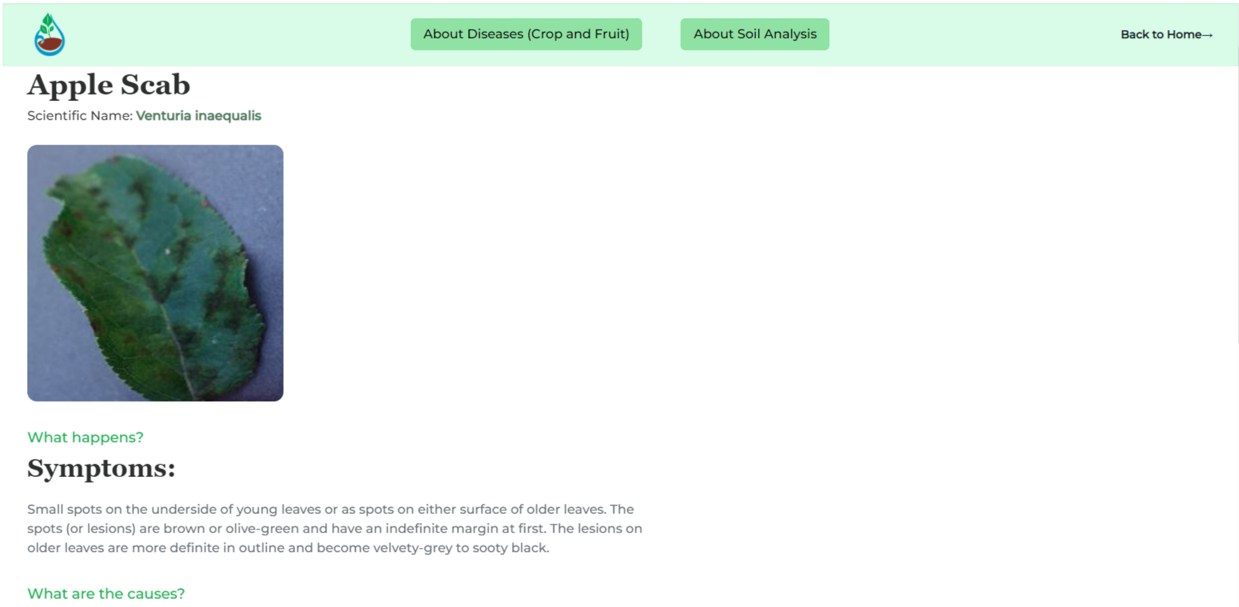


Fig – 4.10 Diagnosis result page Part-1

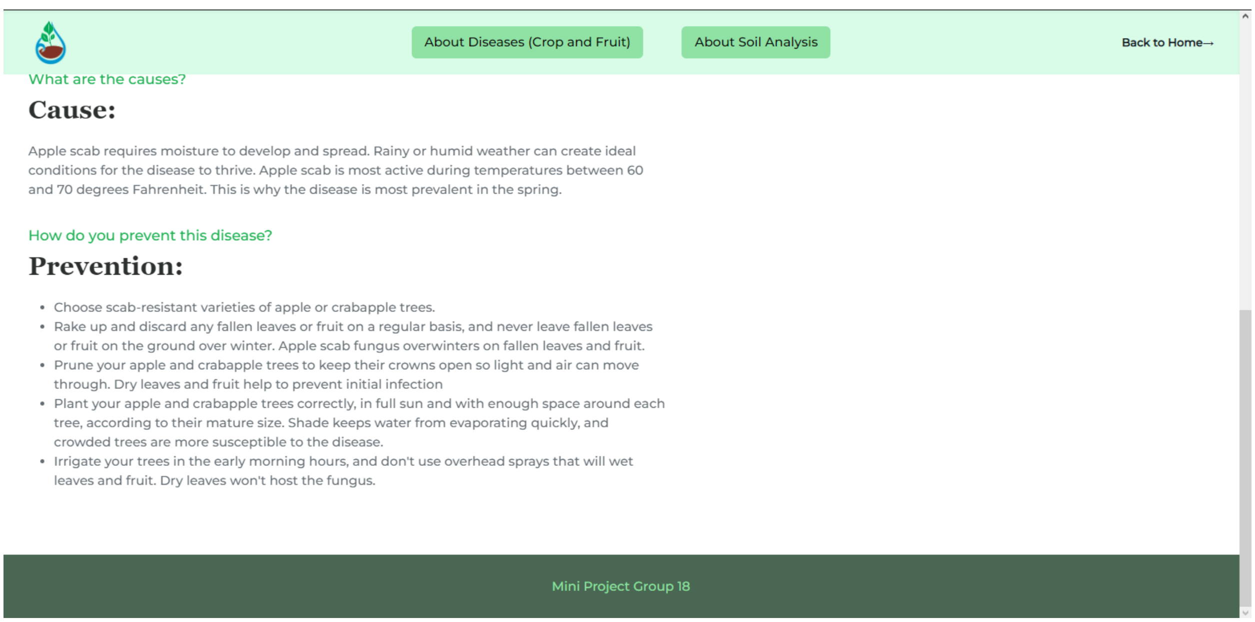


Fig – 4.11 Diagnosis result page Part-2

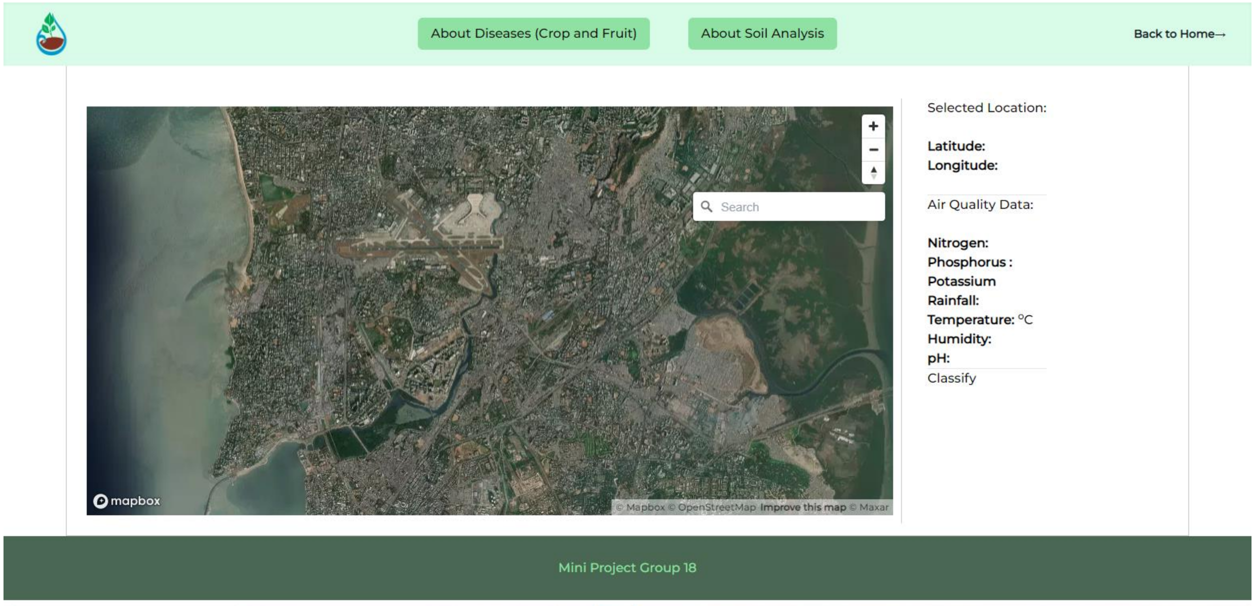


Fig – 4.12 Soil Analysis Page

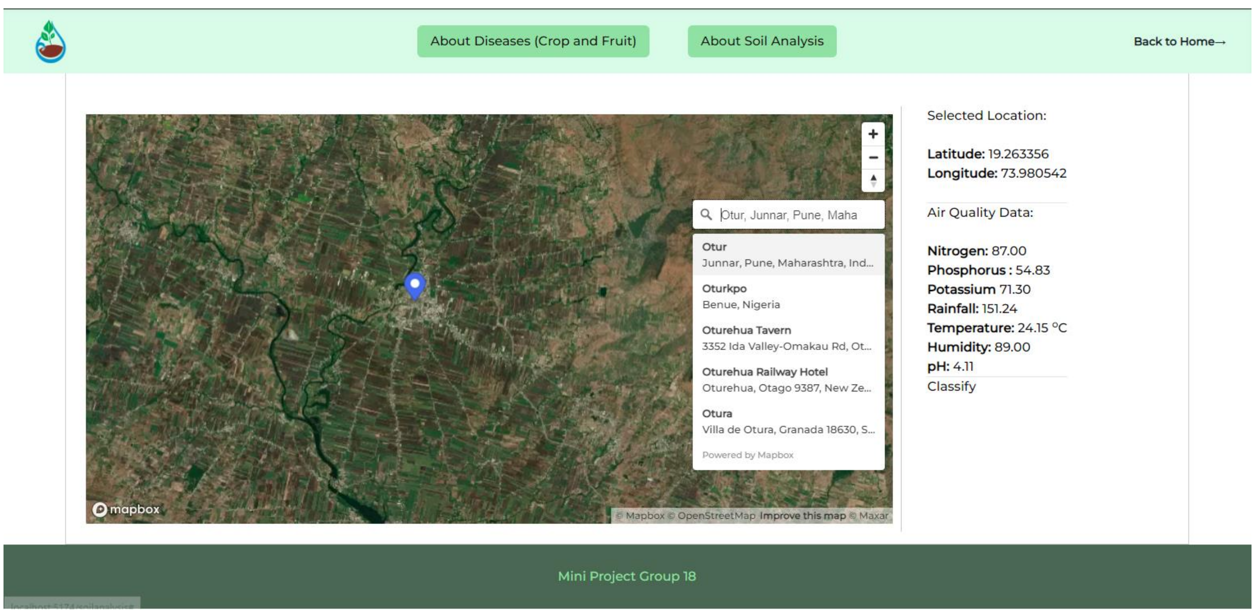


Fig – 4.13 Soil Analysis page after giving input location

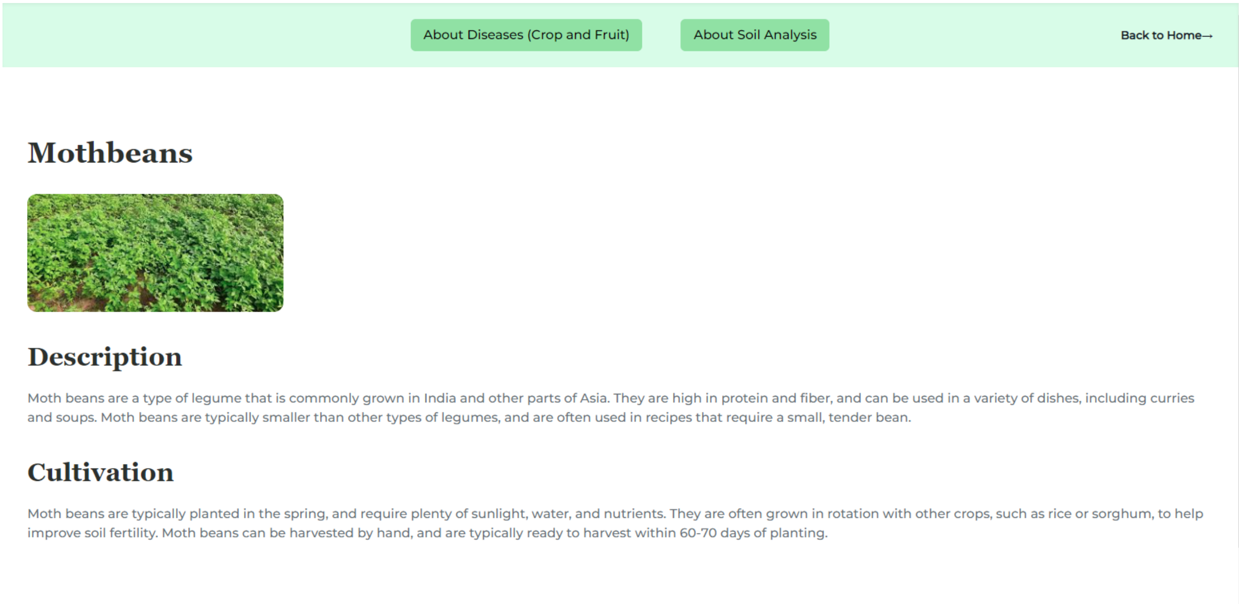


Fig – 4.14 Predicting the crop suitable according to the location

Chapter 5

Conclusions and Learnings

5.1 Conclusions:

The project effectively predicted crop illnesses for maize and a fruit crop using image processing techniques and databases. EfficientNetB2 was used to forecast diseases, and OpenCV was used to analyze digital images of plant leaves and find possible ailments. The use of these image processing techniques for disease identification has proven to be more effective and accurate than conventional approaches, allowing farmers to act quickly to limit additional damage and maximize crop yields.

In addition to disease prediction, the project included soil analysis using maps to identify plots of land and associated soil contents. Using a map interface, farmers may choose a piece of land and get crop recommendations based on soil analysis. The soil study gave farmers important knowledge about the soil's pH level, nitrogen content, and other characteristics. The Ambee API soil content information was used by the crop recommendation system to determine the best crops to plant on the chosen piece of land.

5.2 Future Scope:

The project has future scope to improve its usability. One such option is the creation of a mobile application that would give farmers a simple way to access features for crop disease detection and soil analysis. The app may have functions like image capture, soil analysis based on location, crop recommendations, and other useful data for crop management. The addition of a crop price prediction model is another potential direction for research. This will assist farmers in selecting the best crops to cultivate based on projected market prices. In the end, the project can be expanded to suggest nearby labs for additional testing and analysis of plant samples, assisting farmers in obtaining a more precise diagnosis of the crop disease and putting in place the necessary controls to stop its spread. Based on the crop and location, the project can identify the closest and most suitable lab using the databases of labs that already exist and their experience in diagnosing crop diseases.

5.3 Learning:

Through the development of a crop disease detection system, the project highlights how computer vision algorithms can analyze images of crops and accurately identify diseases or abnormalities. By training a deep learning model on a large dataset of crop images, the system can learn patterns and features indicative of various diseases. This process involves preprocessing the data, augmenting the images, and training the model using techniques like transfer learning.

Furthermore, the project incorporates soil analysis for crop recommendation, illustrating how machine learning algorithms can analyze soil properties and make informed recommendations for suitable crops. By collecting and analyzing soil data such as pH levels, nutrient content, and moisture levels, the system can provide personalized recommendations for crop selection, optimizing agricultural practices, and maximizing yield.

The project emphasizes the significance of user-centered design, ensuring that the crop disease detection system and crop recommendation interface are intuitive, user-friendly, and accessible to farmers and agricultural stakeholders. Usability testing, feedback gathering, and iterative design methodologies contribute to creating an interface that meets the needs and expectations of the target users.

Additionally, the project addresses crucial aspects such as security, scalability, cross-platform compatibility, speed, and easy maintenance. Data security measures are implemented to protect sensitive information, while scalable architecture allows the system to handle growing data volumes and user traffic. Cross-platform compatibility ensures that the system can be accessed from various devices, catering to the diverse technological landscape. Emphasizing speed and efficiency, the project aims to deliver timely results for crop disease detection and provide instant crop recommendations. Lastly, considerations for easy maintenance facilitate the long-term sustainability and upgradability of the system.

Overall, this project serves as a valuable learning experience in leveraging computer vision and machine learning for addressing real-world challenges in agriculture. It underscores the importance of considering user needs, security, scalability, compatibility, speed, and maintenance in the development of such systems. By documenting the methodologies, challenges, and outcomes, this project provides insights and guidance for future endeavors in similar domains, highlighting the transformative potential of technology in improving agricultural practices and positively impacting the lives of farmers and communities.

References

1. Akanksha, E., Sharma, N. and Gulati, K. (2021) ‘OPNN: Optimized probabilistic neural network based automatic detection of maize plant disease detection’, *2021 6th International Conference on Inventive Computation Technologies (ICICT)* [Preprint]. doi:10.1109/icict50816.2021.9358763.

2. Tichkule, S.K. and Gawali, Dhanashri.H. (2016) ‘Plant diseases detection using image processing techniques’, *2016 Online International Conference on Green Engineering and Technologies (IC-GET)* [Preprint]. doi:10.1109/get.2016.7916653.

3. Zhang, X. *et al.* (2018) ‘Identification of maize leaf diseases using improved deep convolutional neural networks’, *IEEE Access*, 6, pp. 30370–30377. doi:10.1109/access.2018.2844405.

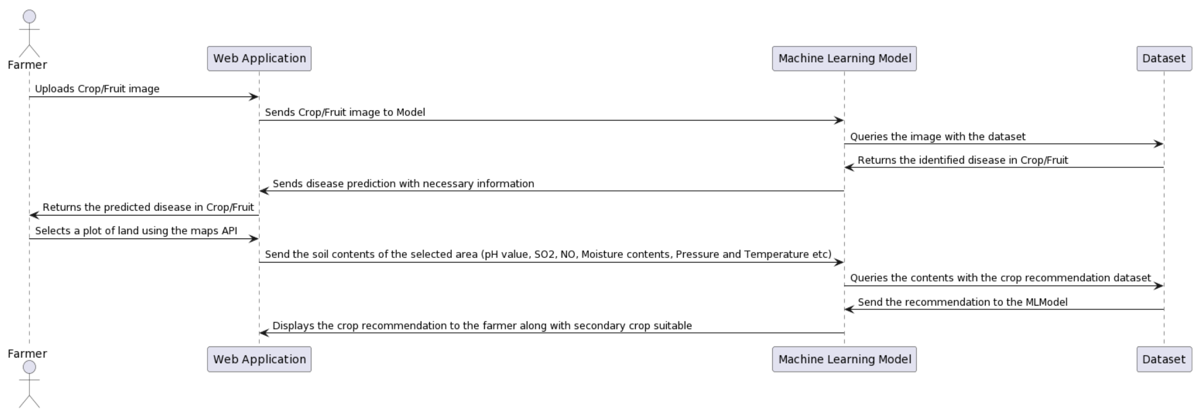
4. Wetterlind, J., Stenberg, B. and Rossel, R.A. (2012) ‘Soil analysis using visible and near infrared spectroscopy’, *Plant Mineral Nutrients*, pp. 95–107. doi:10.1007/978-1-62703-152-3_6.

5. Yadav, J., Chopra, S. and M, V. (2021) ‘Soil analysis and crop fertility prediction using machine learning’, *International Journal of Innovative Research in Advanced Engineering*, 8(3), pp. 41–49. doi:10.26562/ijirae.2021.v0803.003.

6. Prakash, S., Sharma, A. and Sahu, S.S. (2018) ‘Soil moisture prediction using machine learning’, *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)* [Preprint]. doi:10.1109/icicct.2018.8473260.

List of Tables

3.1 Sequence Diagram of the implementation



ACKNOWLEDGEMENT

The successful completion of this project was made possible by the invaluable guidance and assistance provided by the Department of Information Technology, KJSCE, for which we are deeply grateful. We acknowledge that without the supervision and support, this achievement would not have been possible.

Our sincere gratitude goes to our project guide and mentor, Prof. Chirag Desai, for her keen interest in our work and for guiding us every step of the way by providing us with the necessary information to develop a high-quality system. We extend our appreciation to our college for providing us with the opportunity to undertake this project and for the unwavering support and guidance throughout its execution.

We acknowledge the invaluable feedback and suggestions provided by our colleagues and peers, which helped shape and improve our project.

Finally, we acknowledge the contributions of previous researchers and scholars whose work has laid the foundation for this project. Their insights and discoveries have been instrumental in shaping our understanding and approach to this topic.