PROJECT DOCUMENTATION

DEEP LEARNING MODEL FOR DISEASE DETECTION IN TEA LEAVES

1. Introduction:

1.1 Project Overview:

This project is about developing a deep learning model for disease detection in tea leaves. The project aims to provide an automated and technological system for diagnosing tea leaf diseases as a preventative measure to help the farmers and the industry. The project uses various techniques, such as image processing, convolutional neural networks, transfer learning, and evaluation metrics, to acquire, process, classify, and evaluate the images of tea leaves that are affected by different types of diseases. The project also provides a user-friendly and intuitive interface that allows the users to interact with the system and obtain the results and feedback of the disease detection. The project has various benefits, such as improving the tea production, quality, and safety, as well as increasing the knowledge and innovation in the field of plant disease detection.

1.2 Purpose

The purpose of this project is to use deep learning to detect tea leaf diseases and help the tea farmers and the industry. This Provide a novel and innovative solution for tea leaf disease detection, which can improve the tea production, quality, and safety, as well as increase the knowledge and innovation in the field of plant disease detection.

we can Compare the performance and effectiveness of different deep learning models and algorithms for tea leaf disease detection, such as CNN.

2. Literature survey :

2.1 Existing Problem:

One of the main problem is the Tea leaf images taken in natural environments have problems, such as complex backgrounds, dense leaves, and large-scale changes, which affect the accuracy of the image processing and classification techniques

The other is visual symptoms of some tea leaf diseases are similar to each other, such as grey blight, brown blight, white blight, and bud blight, which leads to the disease detection models misclassifying the diseases

The existing convolutional neural networks (CNNs) have low accuracy in detecting and identifying tea leaf diseases, due to the insufficient or imbalanced data, the model complexity, and the generalization ability

2.2 References:

1.The Grey Blight Disease Detection on Tea Leaves Using Improved Deep Convolutional Neural Network" by J.Arun Pandian , paper proposes a novel DCNN using inverted residuals and linear bottleneck layers for diagnosing grey blight disease on tea leaves, and reports superior performance over existing techniques.

2.The "Image Analysis and Detection of Tea Leaf Disease using Deep Learning" by S. Sivaranjani , paper develops a deep CNN called LeNet to detect the tea plant diseases from leaf image set, and claims that LeNet is the perfect CNN model for improving the diagnostic measurement of tea leaves.

3. The paper "Machine Learning-Based Tea Leaf Disease Detection: A Comprehensive Review" by Md. Rashedul Islam provides a systematic review of the literature on machine learning methodologies applied to diagnose tea leaf disease via image classification, and discusses the challenges and future directions..

2.3 Problem Statement Definition:

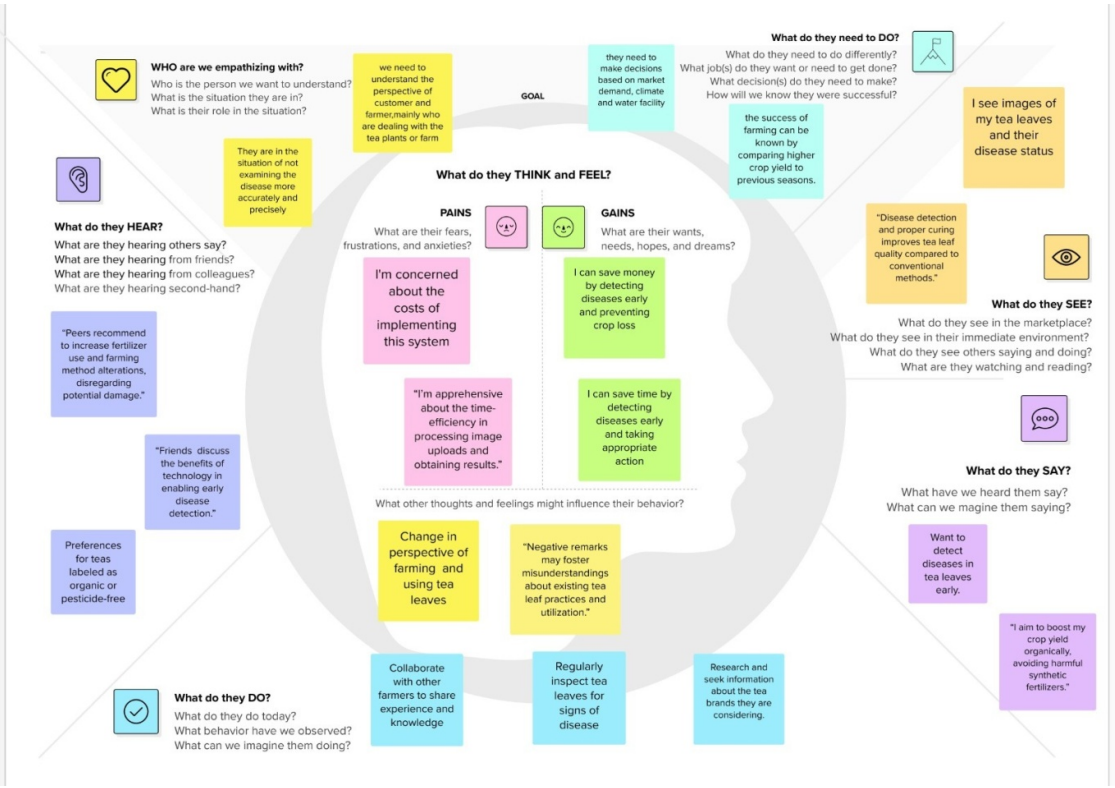
Problem Statement:The problem statement of this project is to develop a deep learning model that can accurately and efficiently detect and classify different types of tea leaf diseases from leaf images, and to provide a user-friendly and intuitive interface for the system.

3.Ideation & Proposed Solution

3.1 Empathy Map Canvas:

In the context of our project, "Data-Driven Insights on Olympic Sports Participation and Performance," an empathy map is a strategic tool used to gain a deeper understanding of the athletes and participants involved in Olympic sports. It helps you develop empathy by putting yourself in their shoes, allowing you to comprehend their needs,

desires, frustrations, and motivations. By understanding the perspective of these athletes, you can better design data driven solutions and insights tailored to their unique challenges and goals.

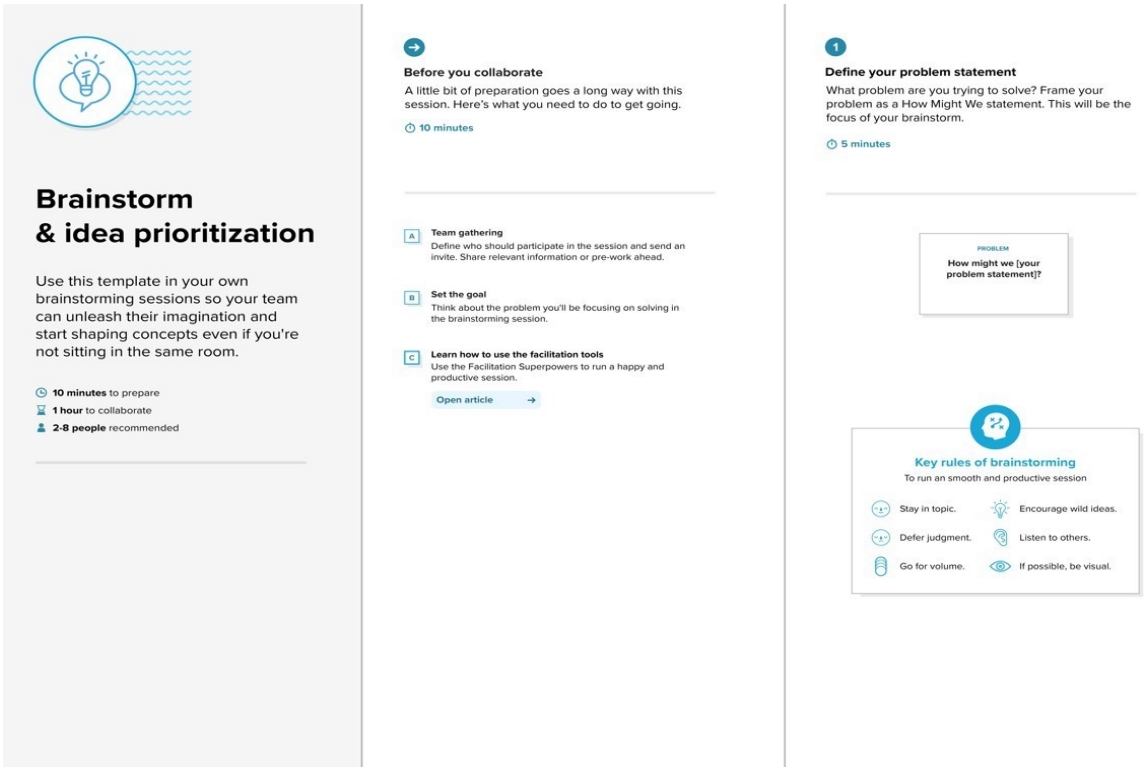


Empathy Map Link:
https://app.mural.co/t/detectingdiseasesintealeaves5889/m/detectingdiseasesintealeaves5889/1698127042953/efa681ebb785dd3fa05e0130f5b7e0c1e478cc7e?fromVisitorModal=true&sender=ub8a1433e3044d9dcf2338349

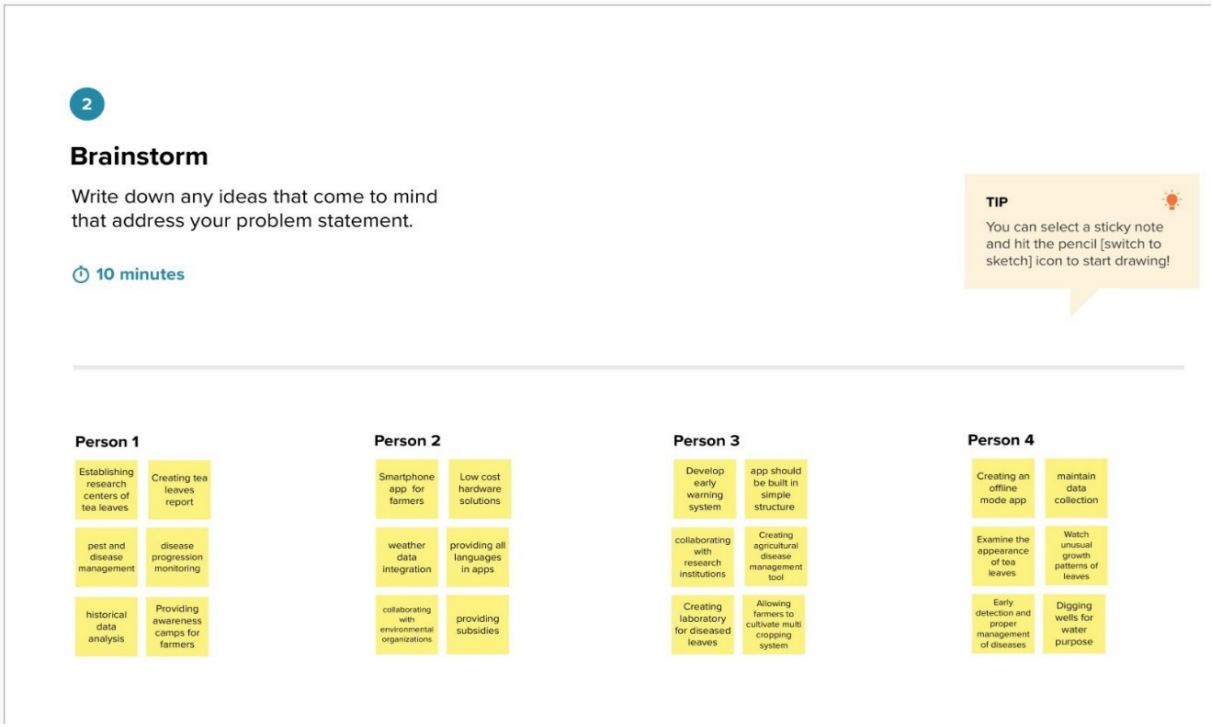
3.2 Ideation & Brainstorming:

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.

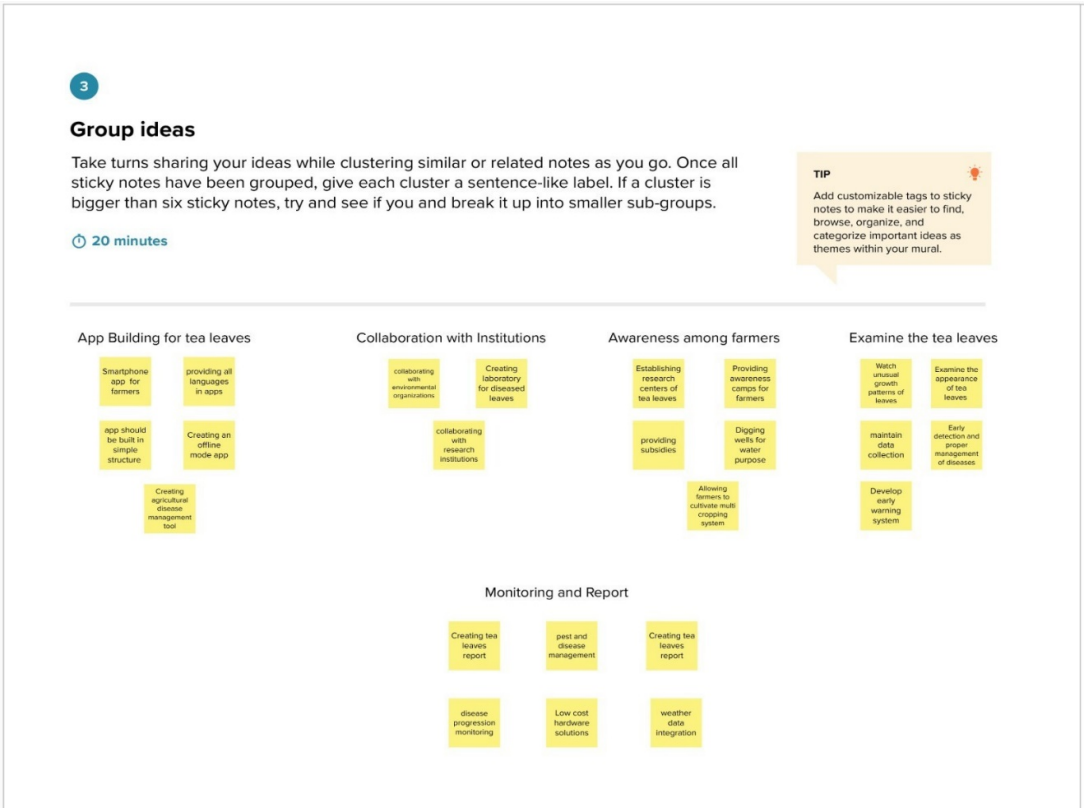
Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Idea Prioritization:



Template link :
https://app.mural.co/t/detectingdiseaseoftealeaves6738/m/detectingdiseaseoftealeaves6738/1698244764744/ce8c29e038c3255cb0b3a70c81ead9f47b7e17fa?sender=ub8a1433e3044d9dcf2338349

4.Required Analysis :

Data collection: Gathering information from a data source relevant to artificial intelligence is the first prerequisite.
Data preparation and cleaning: To make sure the gathered data is appropriate for analysis, it needs to be processed and cleaned. This could entail purging unneeded information, fixing errors and missing values, and formatting the data so that it works with the analysis software.
Data analysis: To find important insights, the data must be analyzed. To better understand the data, this may entail applying methods like regression analysis, data visualization, and descriptive statistics.

4.1 Functional Requirements:

These are the features or functionalities that the system should provide to the users. Some of the functional requirements for this project are:

- The system should be able to acquire the data from the tea fields in real time, using cameras or other devices.
- The system should be able to segment or extract the disease areas from the background using image processing techniques.
- The system should be able to train and test a deep learning model that can accurately classify the tea leaf diseases, such as Convolutional Neural Networks (CNNs) .
- The system should be able to forecast the accuracy of the disease detection based on the input images and the trained model.
- The system should have a user-friendly and intuitive interface that allows the users to interact with the system easily and effectively.

4.2 Non-Functional Requirements:

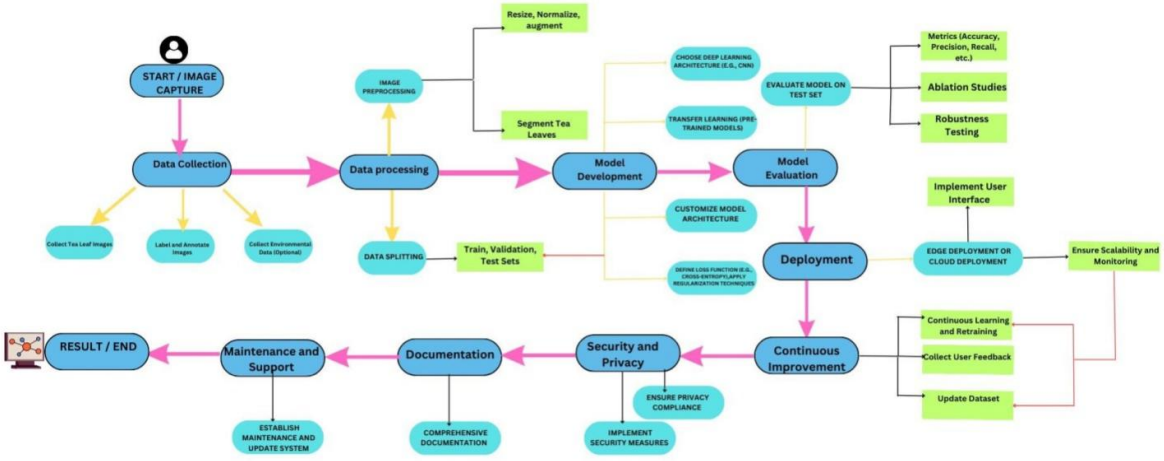
These are the qualities or attributes that the system should have or meet. Some of the non-functional requirements for this project are:

- The system should be reliable and robust, meaning that it should be able to recover from errors or failures.
- The system should be secure and private,so that the data and the model from unauthorized access, modification, or disclosure, and it should respect the users' privacy and consent.
- The system should be efficient and scalable,to process the data and the model in a fast and optimal way, and need to handle large amounts of data and users.
- The system should be compatible and interoperable, to work with different devices, platforms, and networks, and it should be able to communicate and exchange data with other systems or applications.

5. Project Design

5.1 Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



https://www.canva.com/design/DAFy73EfAw/rllhHCtJOpWMaCnX1eTtAA/edit?utm_content=DAFy73EfA-w&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton

5.2 User Stories:

Use the below template to list all the user stories for the product.

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 need registration form that asks for the user's email, password, and other optional information, such as name, location, etc.	The app should be able to create and validate the user's account and profile, and allow the user to access and edit their information, preferences, and history. The app should also have a secure and user-friendly interface and a help section..	High	Sprint-1
		USN-2	As a user, I will receive confirmation emailonce I have registered for the application	I can receive confirmationemail & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the applicationthrough Social Media ads/promotion, Gmail/Outlook,Phone Number,	I can register & access thedashboard with Social media credentials or registered phone number, or continue with gmail/outlook.	Low	Sprint-2
	Login	USN-5	As a user, I can log into the application byentering email & password	I can login using the saved password with the help of chrome or microsoft edge or anyother software available	High	Sprint-1
	Dashboard	USN-6	I want to access in a easy way without much effort At any time	The app should be able to retrieve and analyze the user's diagnosis results, and display the summary and statistics of the images, diseases, treatments, and preventive measures that the user has obtained from the app. The app should also have a user-friendly interface and a help section.	Medium	Sprint-1
Customer (Web user)	Registration	USN-7	I want to be able to register and login to the website using my email and password, so that I can create and access my personal account and profile.	The website should be able to create and validate the user's account and profile, and allow the user to access and edit their information, preferences, and history. .	High	Sprint-1
Researcher,	Report	USN-8	I want to be able to evaluate the performance and accuracy of the deep learning model using various metrics and benchmarks, such as precision, recall, F1-score, ROC curve, etc.	The deep learning model should be able to demonstrate a high performance and accuracy in detecting and diagnosing the diseases of the tea leaves, and be able to compare and contrast with other existing methods and models. The deep learning model should also have a clear report and a presentation section.	Low	Sprint-1
Developer	Support	USN-9	I want to be able to access the deep learning model through an API and integrate it with various applications and platforms, such as mobile apps, web apps, chatbots, etc.	he API should be able to receive and process the image of the tea leaf, and return the diagnosis result with a confidence score, as well as some suggestions for treatment and prevention. The API should also have a clear documentation and a support section.	Medium	Sprint-2
Programmer	Training	USN-10	I want to be able to train, test, and fine-tune the deep learning model using a large and diverse dataset of tea leaf images, along with their labels and annotations.	The deep learning model should be able to achieve a high accuracy and reliability in detecting and diagnosing the diseases of the tea leaves, and be able to handle various scenarios and conditions, such as different lighting, angles, backgrounds, etc. The deep learning model should also have a clear code and a comment section.	Medium	Sprint-1
Project manager	Communication	USN-11	I want to be able to monitor and manage the progress and quality of the project, such as setting goals, assigning tasks, tracking issues, reviewing code, etc.	The project should be able to achieve the desired outcomes and objectives, and meet the expectations and requirements of the stakeholders. The project should also have a clear plan and a communication section.	Low	Sprint-2
Tea consumer	Diagnosis	USN-12	I want to be able to use the deep learning model to check and verify the freshness and safety of the tea leaves that I buy or consume, and avoid any health risks or frauds due to diseases.	The deep learning model should be able to detect and diagnose any diseases that affect the tea leaves, and inform me of the severity and impact of the diseases on the quality and health of the tea. The deep learning model should also have a user-friendly interface and a help section.	Low	Sprint-2

5.3 Solution Architecture

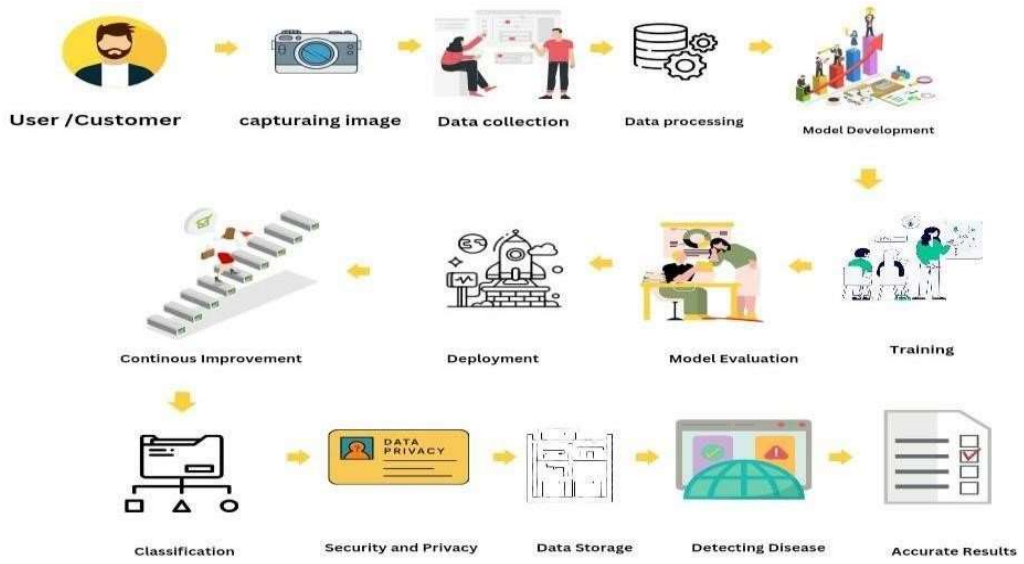


Figure 1: Architecture and data flow of the voice patient diary sample application

6. Project planning and scheduling :

6.1 Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

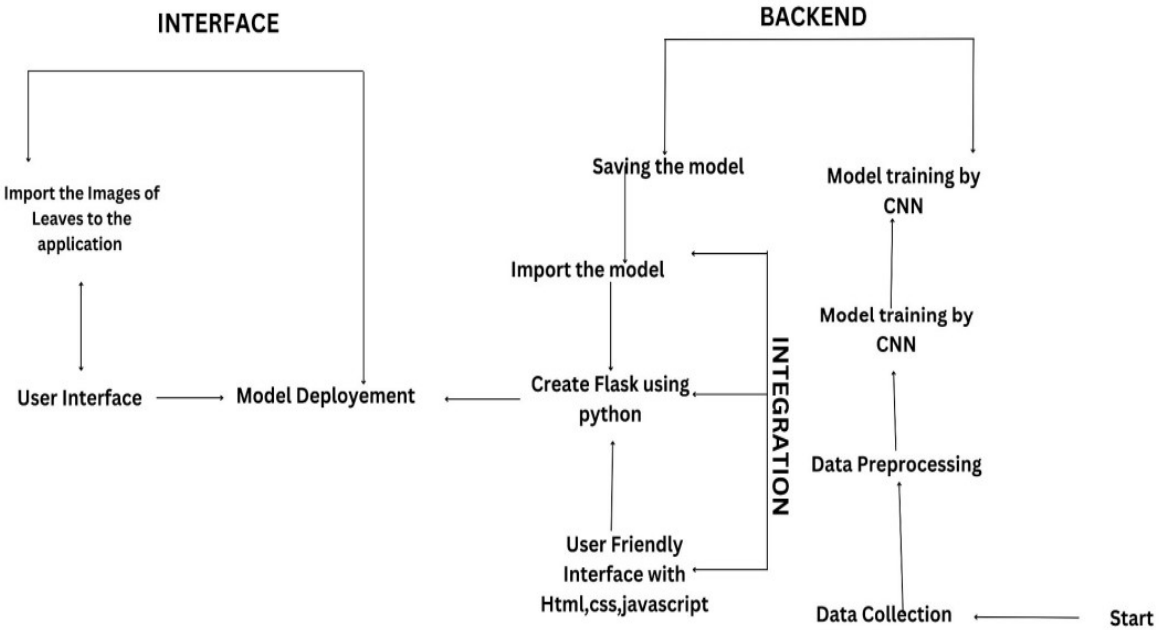
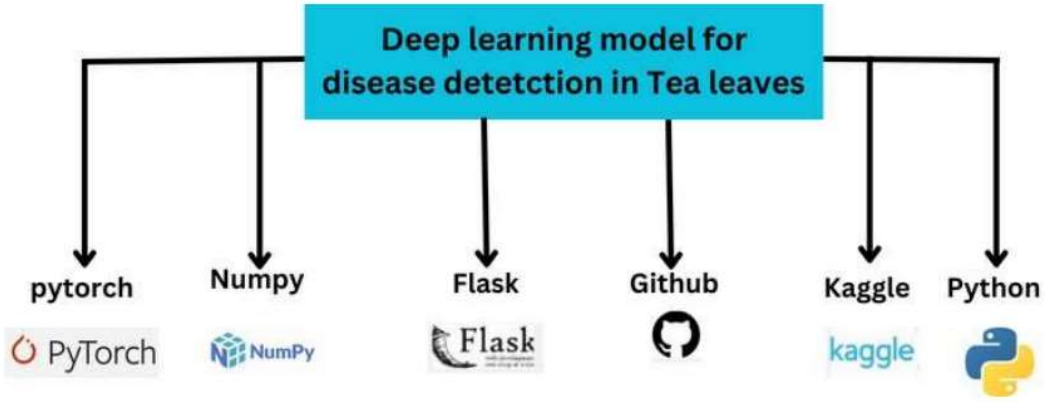


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	Web Application / user interface	An user friendly interface to interact with the application	Html, css, javascript
2.	Data processing	The images need to be preprocessed to enhance their quality, such as resizing, rescaling, cropping, rotating, flipping, and applying principal component analysis (PCA) to reduce noise and redundancy	numpy
3.	Application Logic-1	Logic for a process in the application	Python
4	Database	Collect the Dataset Based on the Problem Statement	File Manager, MySQL etc.
5.	File Storage/ Data	File storage requirements for Storing the dataset	Local System, Google Drive Etc
6	Training	To make it more advanced to detect we ae are improving the CNN model	PyTorch
7	Frame Work	Used to Create a web Application, Integrating Frontend and Back End	Python Flask etc
8.	Deep learning model	Purpose of model	CNN model
9	Data collection	This phase involves collecting images of tea leaves with different types of diseases,	kaggle

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1	Open-Source Frameworks	List the open-source frameworks used	Python's Flask
2	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	Controls
3	Scalable Architecture	Justify the scalability of architecture (3 – tier, Microservices)	
4	Availability	Justify the availability of applications (e.g. use of load balancers, distributed servers etc.)	Web service
5	Performance	Design consideration for the performance of the application	



Brief summary of what we did

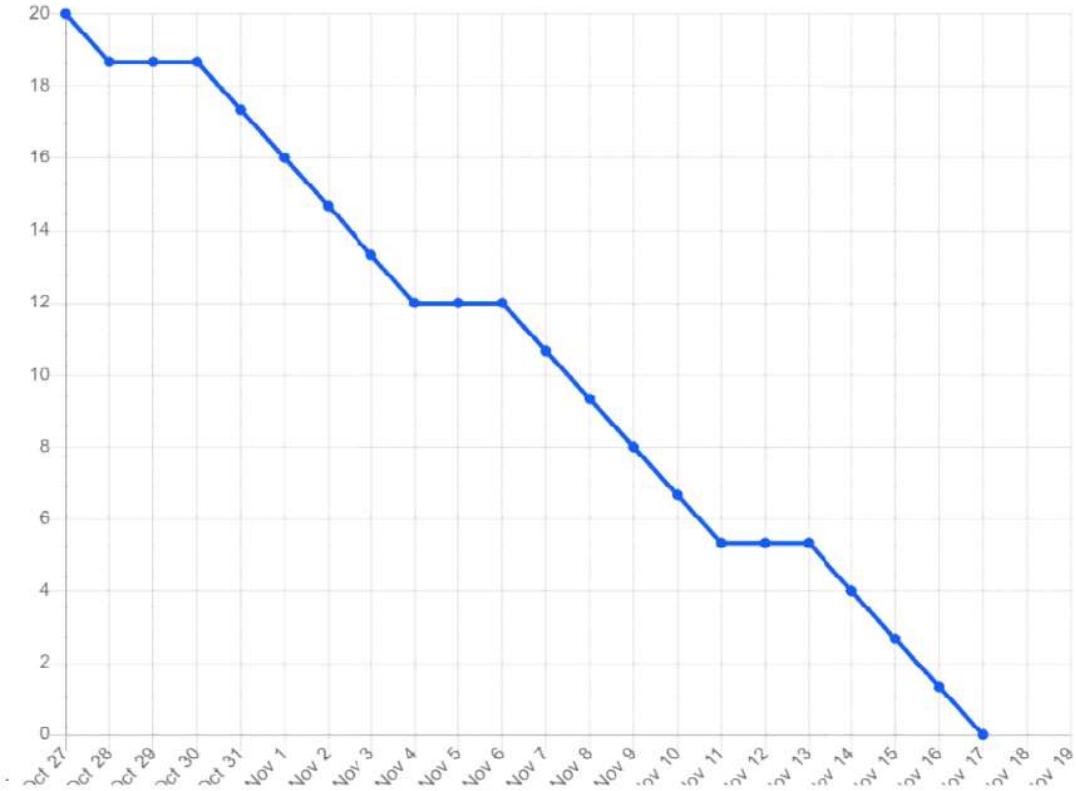
- We collected and preprocessed images of tea leaves with different diseases using tools like numpy.
- We developed and trained a deep CNN model using frameworks like PyTorch.
- We deployed and tested the CNN model using tools like Flask.

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 Release Date (Actual)
Sprint-1	3	2 Days	27 Oct 2023	29 Oct 2023	20	29 Oct 2022
Sprint-2	3	2 Days	30 Oct 2023	01 Nov 2023	20	01 Nov 2023
Sprint-3	4	7 Days	02 Nov 2023	09 Nov 2023	20	09 Nov 2023
Sprint-4	4	3 Days	10 Nov 2023	09 Nov 2023	20	09 Nov 2023
Sprint-5	6	6 Days	14 Nov 2023	19 Nov 2023	20	19 Nov 2023

BURNDOWN CHART:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time



7. Coding & Solutioning (Explain the features added in the project along with code)

app.py home.html

new > templates > home.html > ...

```html
<!DOCTYPE html>
<html>

<head>
    <title>Tea Leaf Disease Detection</title>
    <link rel="stylesheet" href="/static/style.css">
    <link rel="icon" href="data:,">
</head>

<body>
    <h1>Tea Leaf Disease Detection</h1>
    <p>Upload an image of a tea leaf to get the disease diagnosis.</p>
    <form id="upload-form">
        <label for="image">Upload Here</label>
        <input class=".custom-file-input" type="file" id="image" accept="image/*"><br>
        <button type="submit" id="button">Predict</button>
    </form>
    <div id="prediction-container" style="display: none;">
        <img id="predicted-image" width="224" height="224">
        <p id="predicted-result"></p>
    </div>
    <!-- Update the script tag to use a direct URL -->
    <script src="/static/script.js"></script>
    <!-- In your HTML file -->
    <script>
        var baseUrl = "{{ url_for('static', filename='') }}";
    </script>

</body>

</html>
```

Drive link :- https://drive.google.com/drive/folders/1PG37Zq-w9EI44UCTCtUyA41TA_F4fXfk?usp=sharing

8. PERFORMANCE TESTING

Project team shall fill the following information in model performance testing template.

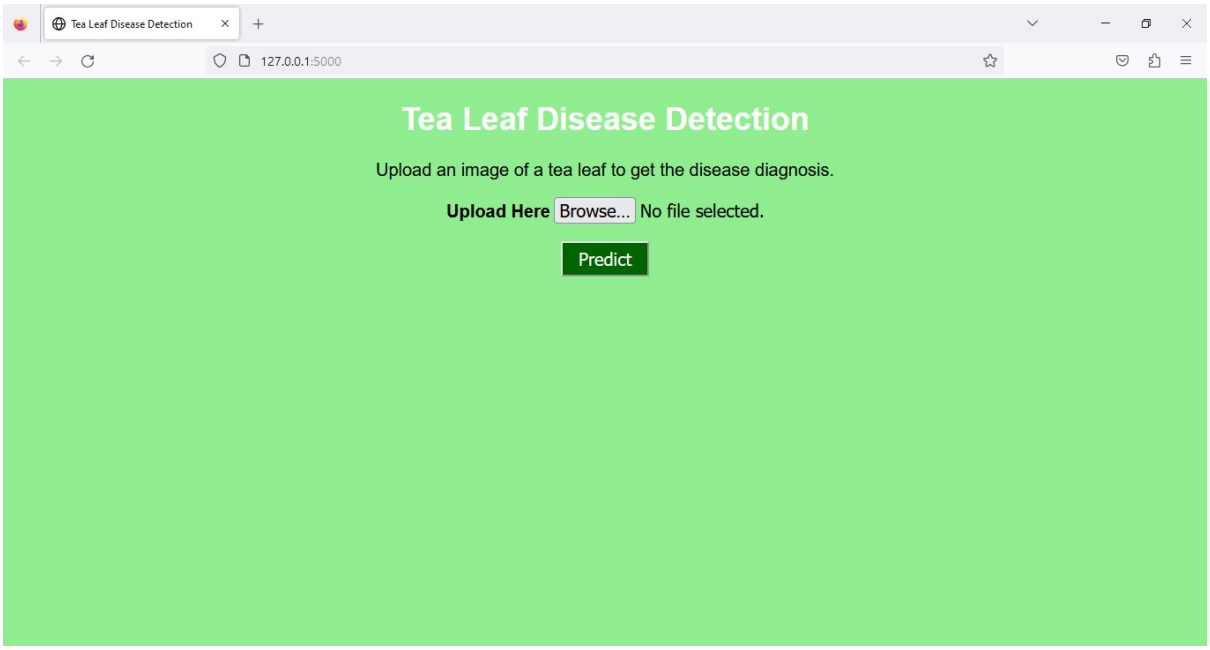
S.No.	Parameter	Values	Screenshot
1.	Model Summary	Total params: 811832 (3.10 MB) Trainable params: 811704 (3.10 MB) Non-trainable params: 128 (512.00 Byte)	''' Total params: 811832 (3.10 MB) Trainable params: 811704 (3.10 MB) Non-trainable params: 128 (512.00 Byte)
2.	Accuracy	Training Accuracy – 80% Validation Accuracy -78%	loss accuracy val_loss val_accuracy 4 0.839342 0.679379 0.779666 0.785311 5 0.851154 0.648305 0.754400 0.774011 6 0.780705 0.682203 0.771604 0.717514 7 0.778723 0.690678 0.844311 0.644068 8 0.778590 0.689266 0.835845 0.672316
			Training accuracy: accuracy 0.80 128 macro avg 0.82 0.82 0.81 128 weighted avg 0.81 0.80 0.79 128

9. Results

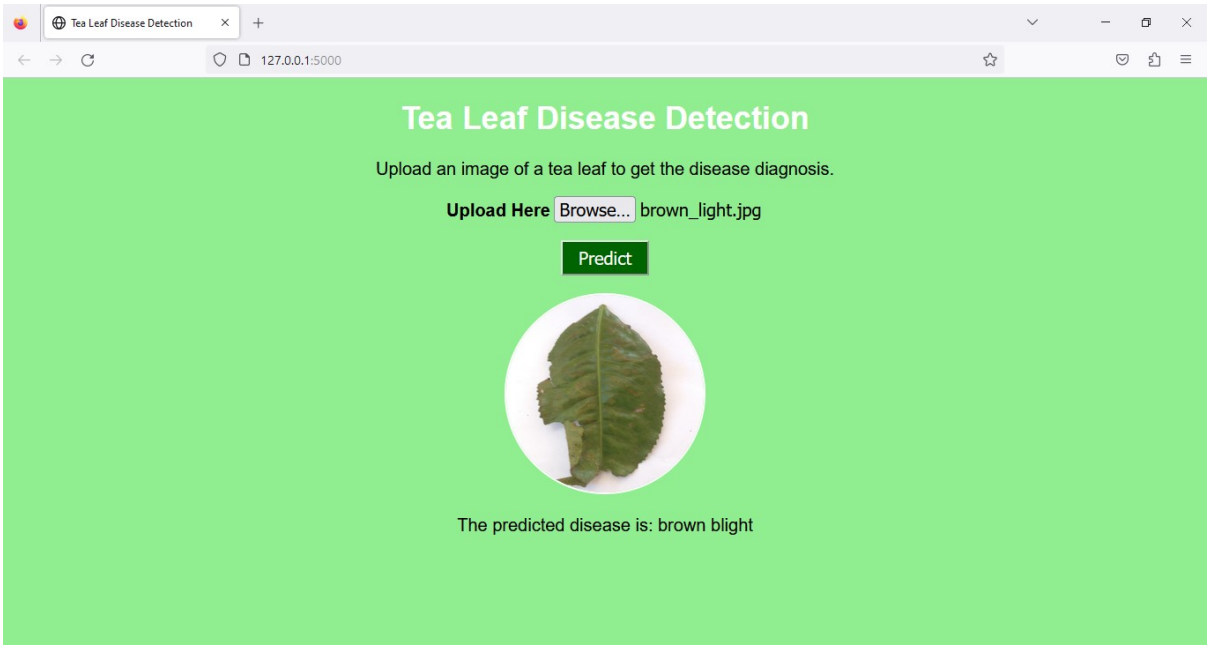
9.1 Output Screenshots

When you run the app.py file and click on the server url in terminal, you will redirected to home page. The

home page will looks like:



When we upload the image and select predict and then it looks like:



10. Advantages &Disadvantages :

Advantages:

1. The project can provide a beneficial and innovative solution for the tea farmers and the tea industry, by using deep learning to detect tea leaf diseases and improve the tea production, quality, and safety.
2. The project can compare the performance and effectiveness of different deep learning models and algorithms for tea leaf disease detection, such as CNN.
3. The project can provide a user-friendly and intuitive interface that allows the users to interact with the system and obtain the results of the disease detection, as well as to generate reports and forecasts.
4. The project can contribute to the scientific and technological advancement of the field of plant disease detection and diagnosis using deep learning methods, and generate new insights and opportunities for the tea farmers and the industry.

Disadvantages:

1. The project can face some challenges and limitations, such as data quality, model complexity, and generalization, which can affect the accuracy and efficiency of the disease detection.
2. The project can require a lot of resources and expertise, such as cameras,networks, servers, storage, and skilled workers, to acquire, process, and classify the images of tea leaves.
3. The project can encounter some ethical and social issues, such as privacy, security, consent, and trust, regarding the data and the model of the disease detection, and the impact of the system on the tea farmers and the consumers.

11. Conclusion

12. Future Scope

Some of the future scope for this project are:

1. To extend the dataset and the model to include more types and classes of tea leaf diseases, as well as other plant diseases, to increase the diversity and coverage of the disease detection.
2. To improve the data quality and the model performance by using advanced techniques, such as data augmentation, transfer learning, ensemble learning, and attention mechanisms,

to overcome the problems of insufficient or imbalanced data, model complexity, and generalization.

3. To integrate the system with other applications or platforms, such as mobile apps, web services, or cloud computing, to provide more accessibility and convenience for the users, as well as to leverage the advantages of distributed and parallel computing.
4. To explore the social and ethical implications of the system, such as the impact of the system on the tea farmers' livelihood, the consumers' health, and the environment, and to ensure the privacy, security, consent, and trust of the data and the model.

13. APPENDIX (Source Code GitHub & Project Demo Link completion.)

Project demo link : https://drive.google.com/drive/folders/1PG37Zq-w9EI44UCTCtUyA41TA_F4fXfk?usp=sharing

Github link:- https://github.com/smartinternz02/SI-GuidedProject-600653-1698242742