Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	13 May 2023
Team ID	NM2023TMID15563
Project Name	Deep learning model for the detecting diseases in tea leaves

Components & Technologies:

1. Programming Language: Python is widely used in the field of deep learning due to its extensive libraries and frameworks support. It provides a rich ecosystem for data manipulation, image processing, and model development.

2. Deep Learning Framework: PyTorch or TensorFlow are popular frameworks for developing deep learning models. They provide high-level APIs, extensive support for neural networks, and efficient computation on GPUs, enabling faster training and inference.

3. Image Processing Libraries: OpenCV (Open Source Computer Vision Library) is commonly used for image preprocessing tasks, such as resizing, cropping, and color enhancement. It provides a wide range of functions for image manipulation and feature extraction.

4. Data Manipulation Libraries: NumPy and Pandas are essential libraries for data manipulation and preprocessing. They offer powerful data structures and functions for handling numerical data and performing operations like data cleaning, filtering, and transformation.

5. Convolutional Neural Network (CNN) Architectures: For developing the deep learning model, you can utilize pre-existing architectures such as VGGNet, ResNet, Inception, or EfficientNet. These architectures have proven to be effective in image classification tasks and can be implemented using the chosen deep learning framework.

6. Transfer Learning: Leveraging pre-trained models is crucial for transfer learning. Models pre-trained on large-scale datasets like ImageNet can be used as a starting point, and their weights can be fine-tuned on the tea leaf dataset. This can be achieved through the deep learning framework's APIs and utilities.

7. Model Training and Deployment: The deep learning model can be trained and deployed on frameworks like PyTorch Lightning or TensorFlow Serving. These frameworks provide convenient APIs and tools for managing the training process, optimizing hyperparameters, and deploying the trained model for inference.

8. Web-Based Interface: If a web-based interface is required for user interaction, frameworks like Flask or Django can be used to develop a web application. These frameworks enable you to build a user-friendly interface for uploading tea leaf images and displaying disease detection results.

9. Cloud Services: Cloud platforms like Amazon Web Services (AWS), Google Cloud Platform (GCP), or Microsoft Azure can be utilized for training the deep learning model on powerful GPUs or TPUs. These platforms also offer services for hosting web applications and deploying trained models in a scalable and cost-effective manner.

10. Version Control and Collaboration: Git and GitHub/GitLab can be used for version control and collaboration among team members. These tools allow for effective code management, tracking changes, and collaboration on the project.

This technology stack provides a robust set of tools and frameworks for developing, training, and deploying a deep learning model for detecting diseases in tea leaves. However, it's important to choose the specific components based on your familiarity, project requirements, and team preferences.