**Mobile Application Development**

***Potato Leaf Disease Identification***

**Team-2**

**Nithees M - CB.EN.U4CSE21338**

**R Kaviswar – CB.EN.U4CSE21345**

**Problem Statement**

Potato crops are vulnerable to diseases like early blight and late blight, which can severely impact yield and quality. Traditional disease identification methods are time-consuming and often inaccurate, especially in resource-limited settings. There is a need for a fast, accurate, and accessible solution to diagnose these diseases in real-time, allowing farmers to take immediate action and reduce crop losses. To address this, we propose developing a CNN-based machine learning model integrated into a mobile app to classify potato leaves as early blight, late blight, or healthy, enabling timely and informed decision-making for sustainable agriculture.

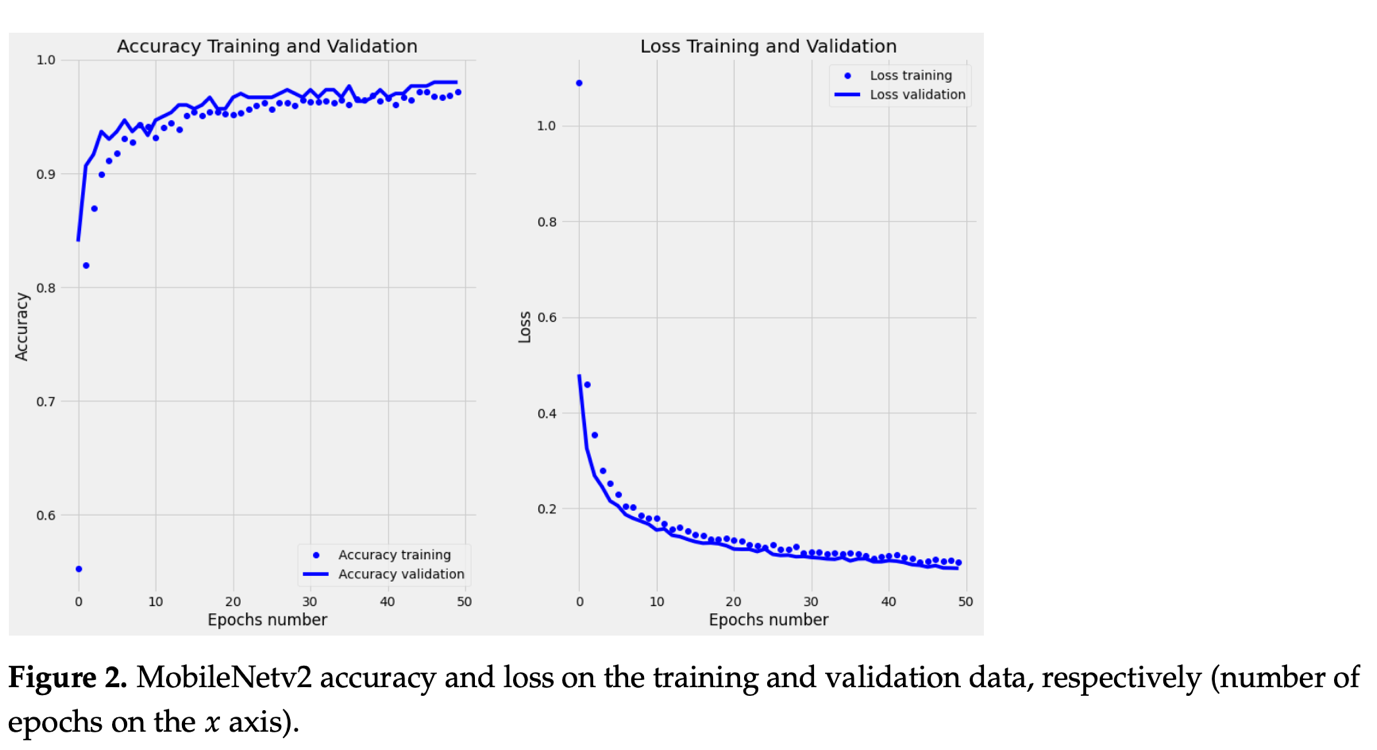
**Introduction**

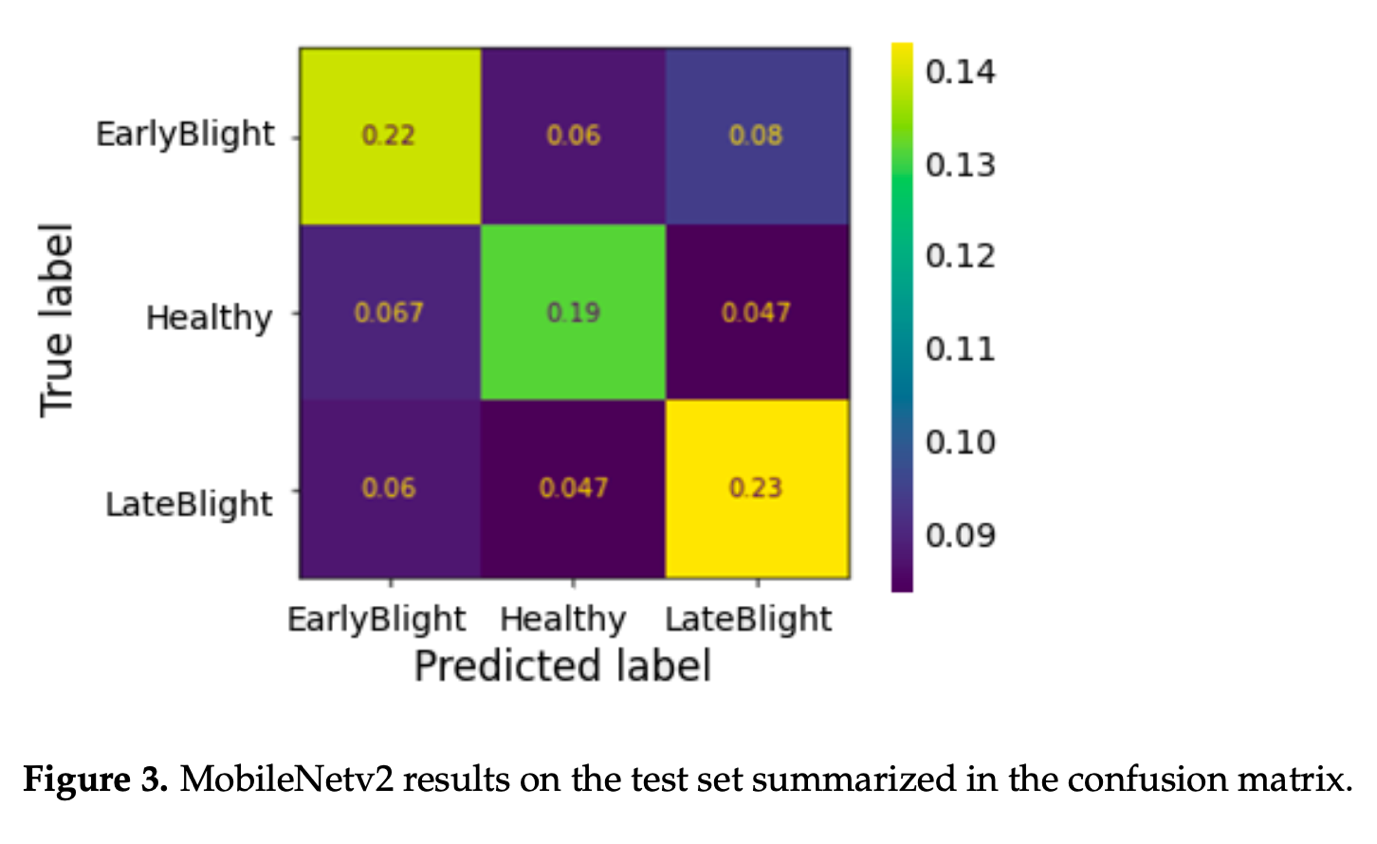
Potato leaf diseases, particularly early blight and late blight, pose significant threats to agriculture, affecting both crop yield and quality. Early blight, caused by Alternaria solani, results in dark spots on leaves and premature defoliation, while late blight, caused by Phytophthora infestans, can lead to rapid tissue decay and even complete crop failure. Timely and accurate identification of these diseases is crucial for effective treatment and minimizing economic losses.

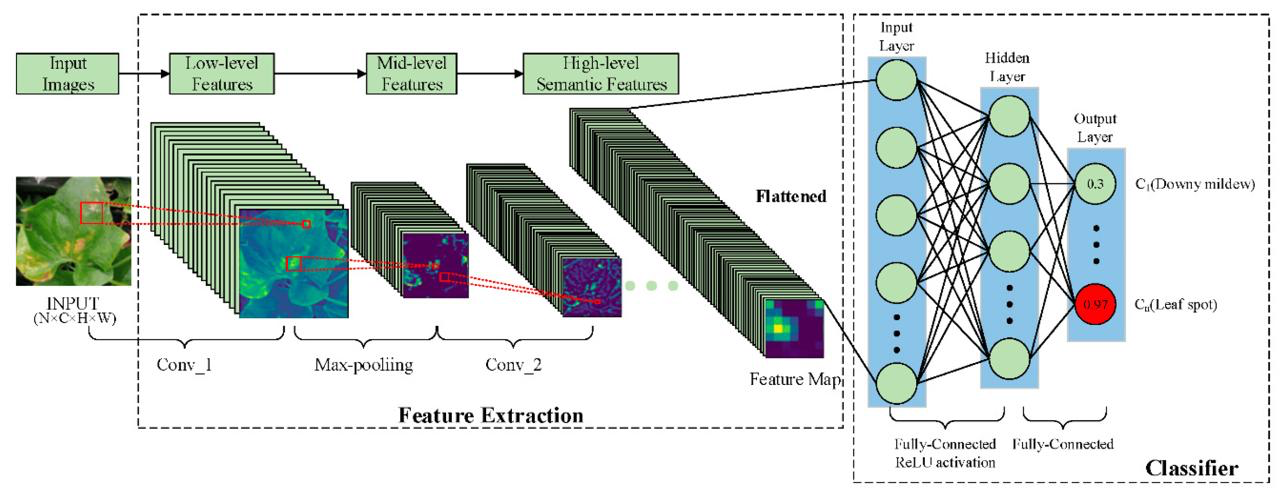
To address this challenge, we are developing a convolutional neural network (CNN)-based machine learning model aimed at classifying potato leaves into three categories: early blight, late blight, and healthy. CNNs are highly effective for image classification tasks, making them ideal for detecting subtle differences in leaf texture and coloration. Our goal is to integrate this model into a mobile application, providing farmers and agronomists with an accessible tool for real-time disease diagnosis, ultimately contributing to more sustainable agricultural practices.

**Base paper:** A Mobile App for Detecting Potato Crop Diseases

|  |  |  |
| --- | --- | --- |
| Technology/Tool | Purpose | Remarks |
| CNNs | Used for the automatic detection of potato crop diseases via leaf images | Includes architectures like MobileNetv2, VGG16, VGG19, InceptionV3, Xception. MobileNetv2 was found to be the most effective for mobile application deployment. |
| MobileNetv2 | Lightweight CNN architecture tailored for mobile applications | Selected for its high accuracy (99.43%) and low computational cost, making it ideal for low-performance mobile devices. |
| |  | | --- | | **Android Studio** |  |  | | --- | |  | | |  | | --- | | Development environment used to build the mobile application |  |  | | --- | |  | | |  | | --- | | Supports integration with TensorFlow-lite for deploying CNN models on Android devices. |  |  | | --- | |  | |
| Kotlin | |  | | --- | | Programming language used for developing the mobile app |  |  | | --- | |  | | |  | | --- | | Facilitates modern Android app development with compatibility across various Android versions. |  |  | | --- | |  | |
| |  | | --- | | **TensorFlow-lite** |  |  | | --- | |  | | |  | | --- | | Framework used for deploying machine learning models on mobile devices |  |  | | --- | |  | | |  | | --- | | Optimizes deep learning models like MobileNetv2 for mobile platforms, ensuring efficient performance. |  |  | | --- | |  | |
| |  | | --- | | **PlantVillage Dataset** |  |  | | --- | |  | | |  | | --- | | Dataset containing labeled images of potato leaves, used for training and testing the CNN models |  |  | | --- | |  | | |  | | --- | | Crucial for developing and validating the disease detection models, focusing on diseases like late blight and early blight. |  |  | | --- | |  | |
| |  | | --- | | **Transfer Learning** |  |  | | --- | |  | | |  | | --- | | Technique to fine-tune pre-trained models on new datasets |  |  | | --- | |  | | |  | | --- | | Helps in adapting the CNN models for potato disease classification with limited computational resources. |  |  | | --- | |  | |

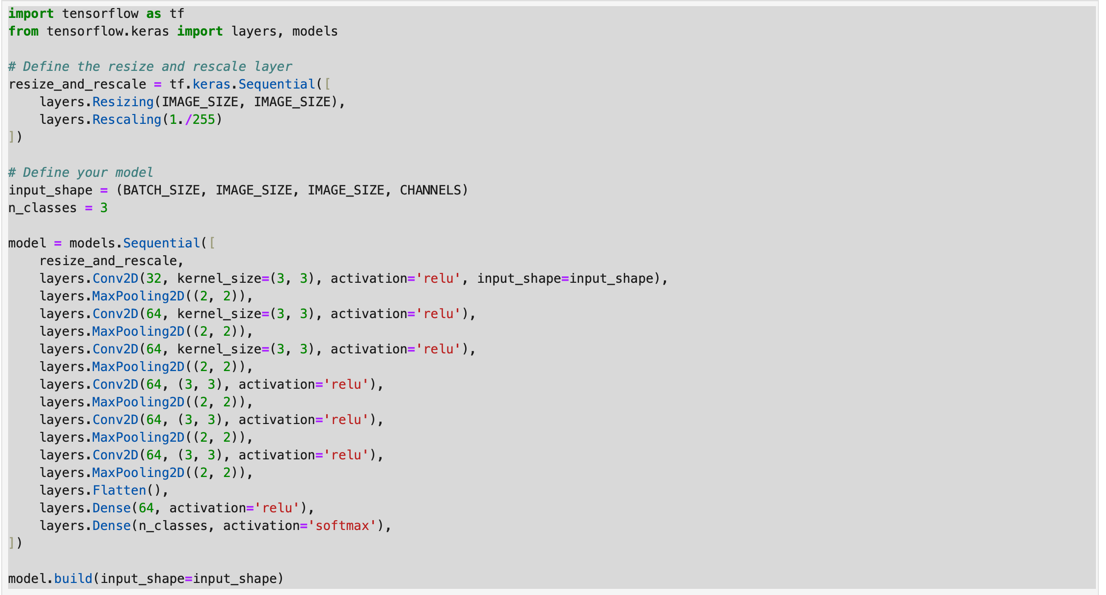
****

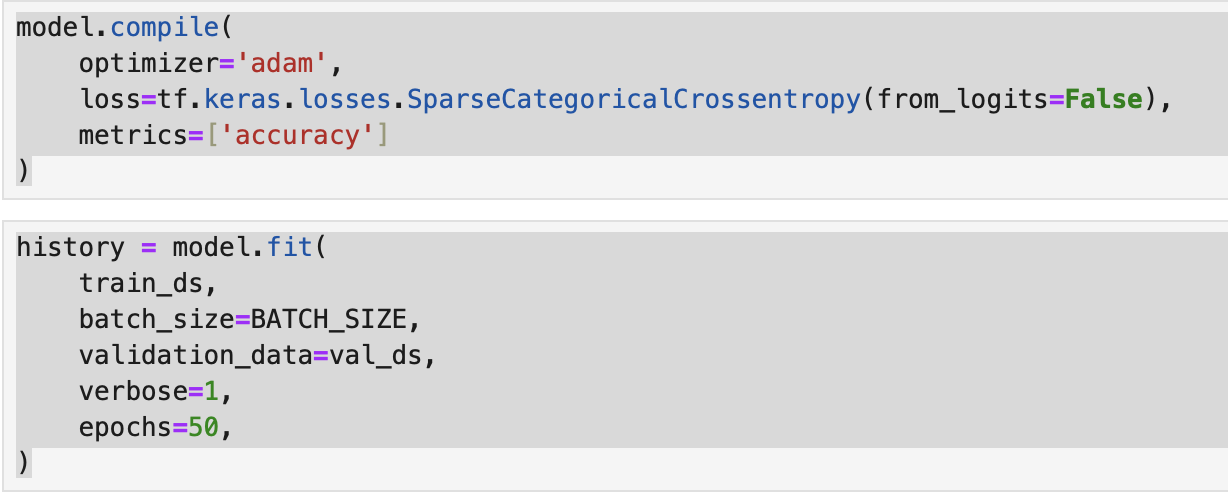
****

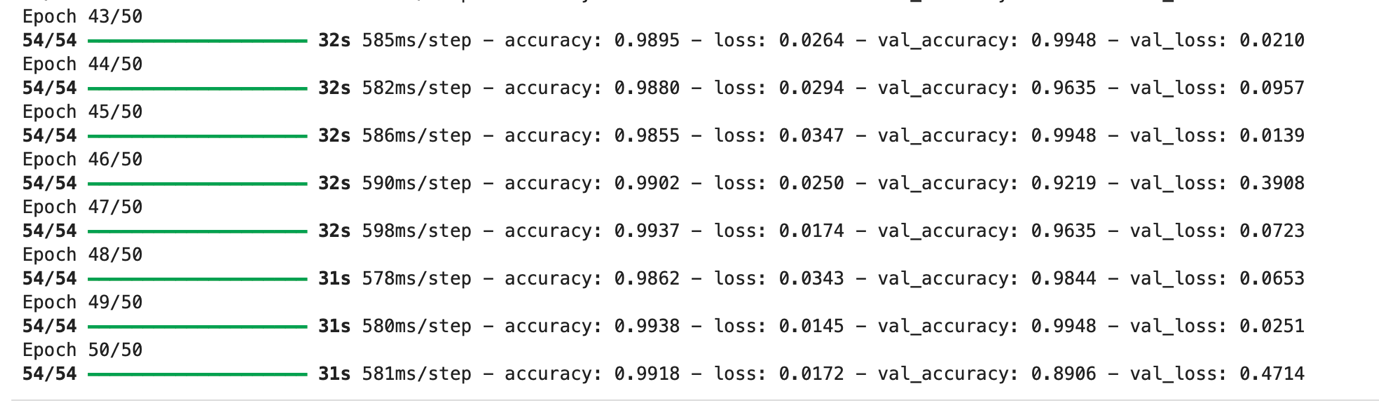
****

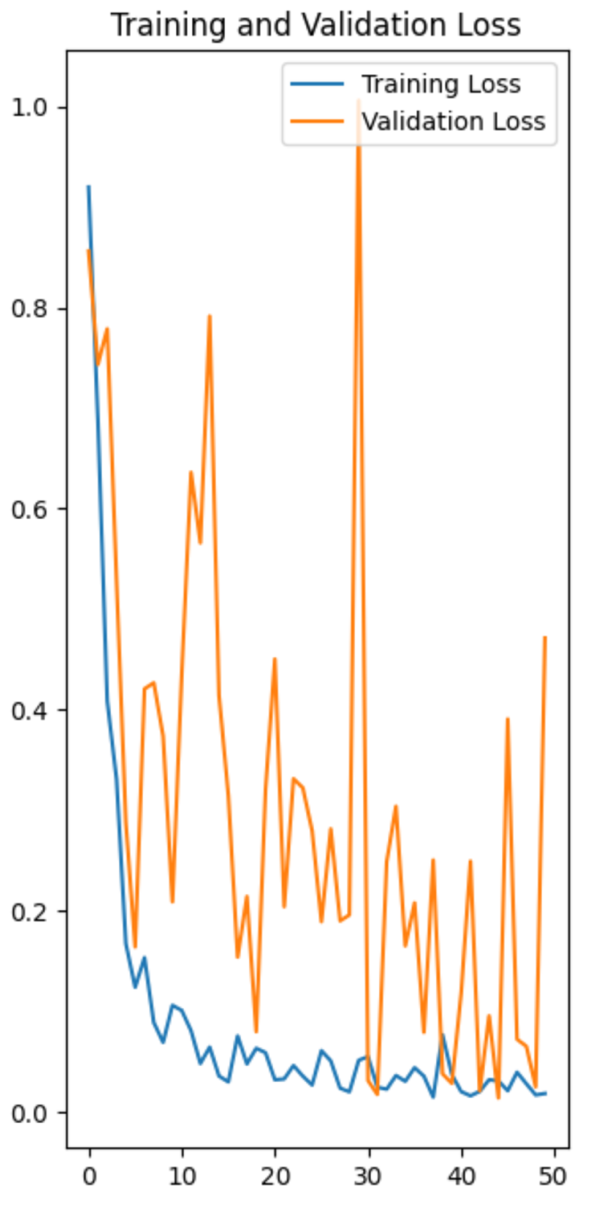
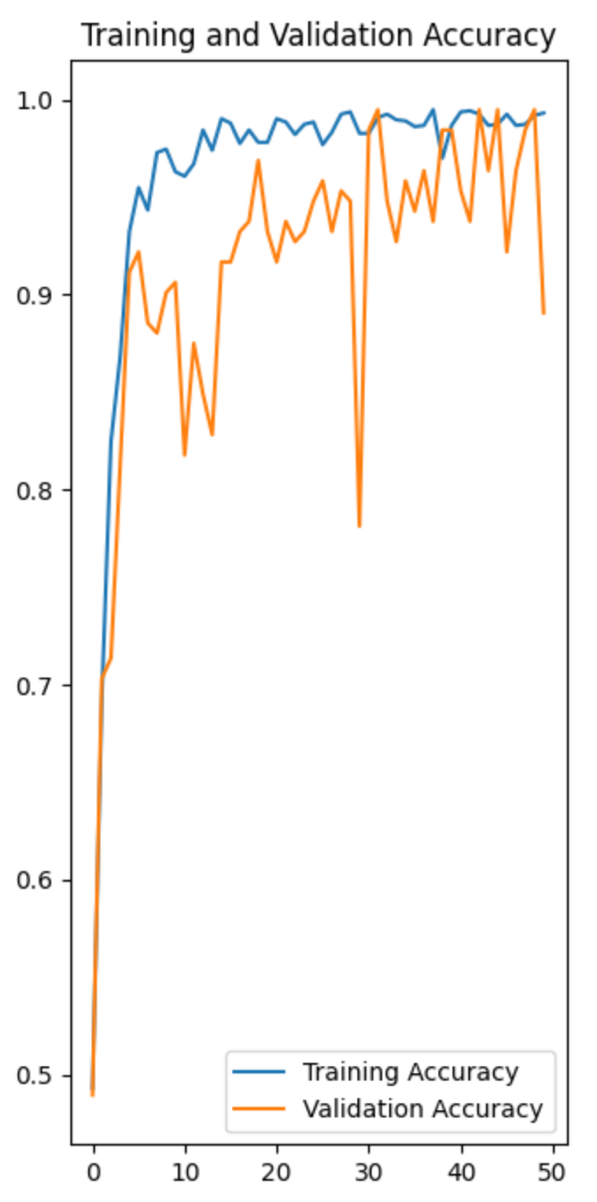
**Our Model:**

**Model 1:** Deep Convolutional Neural Network

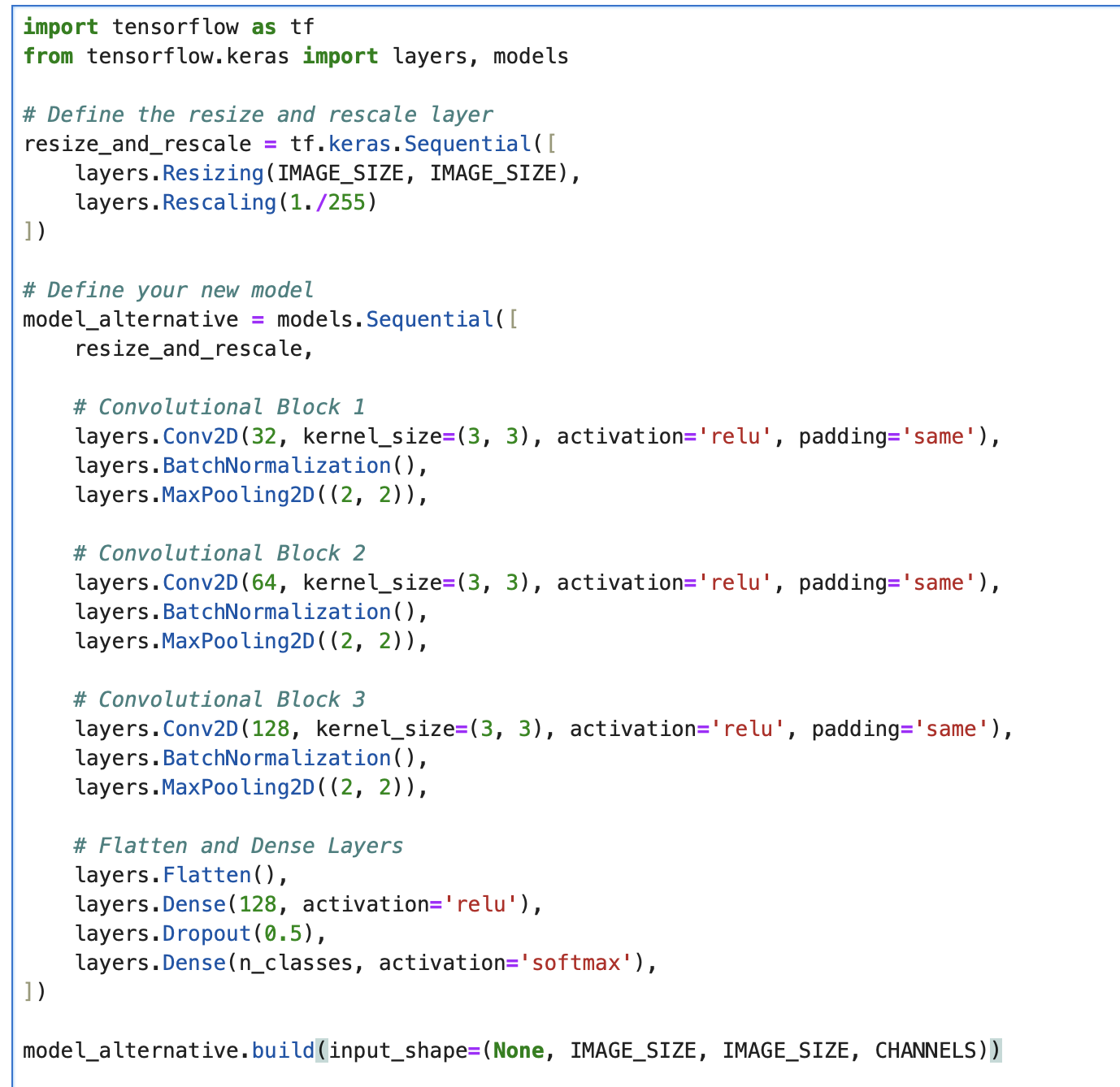
****

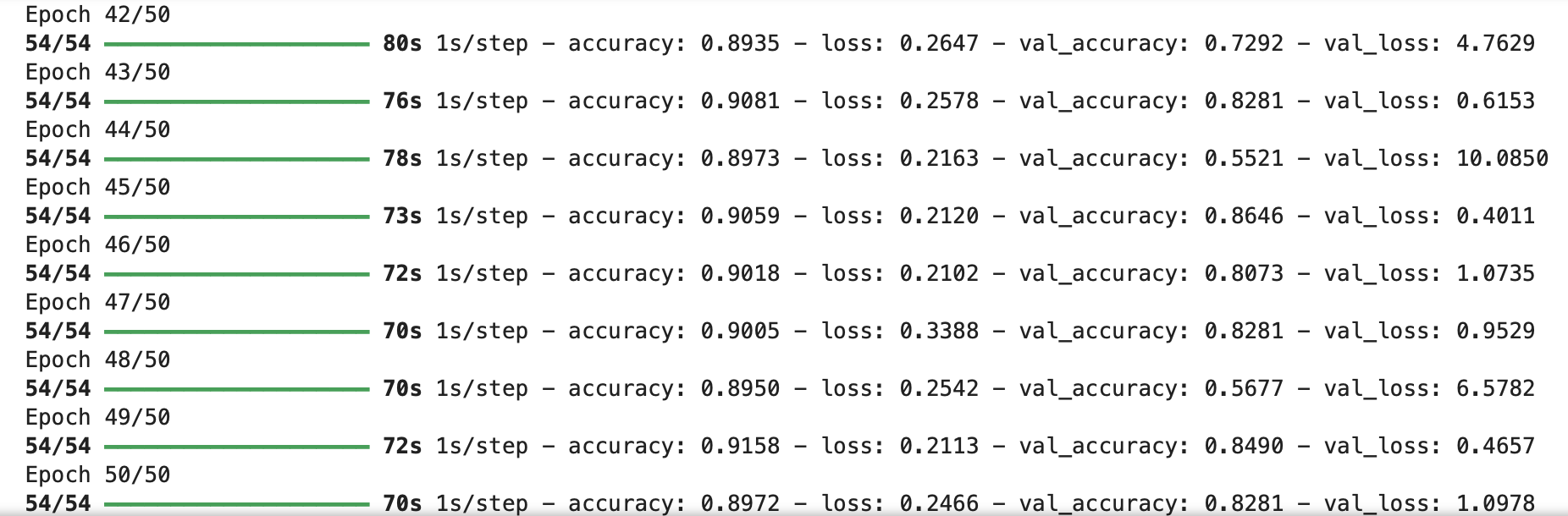
****

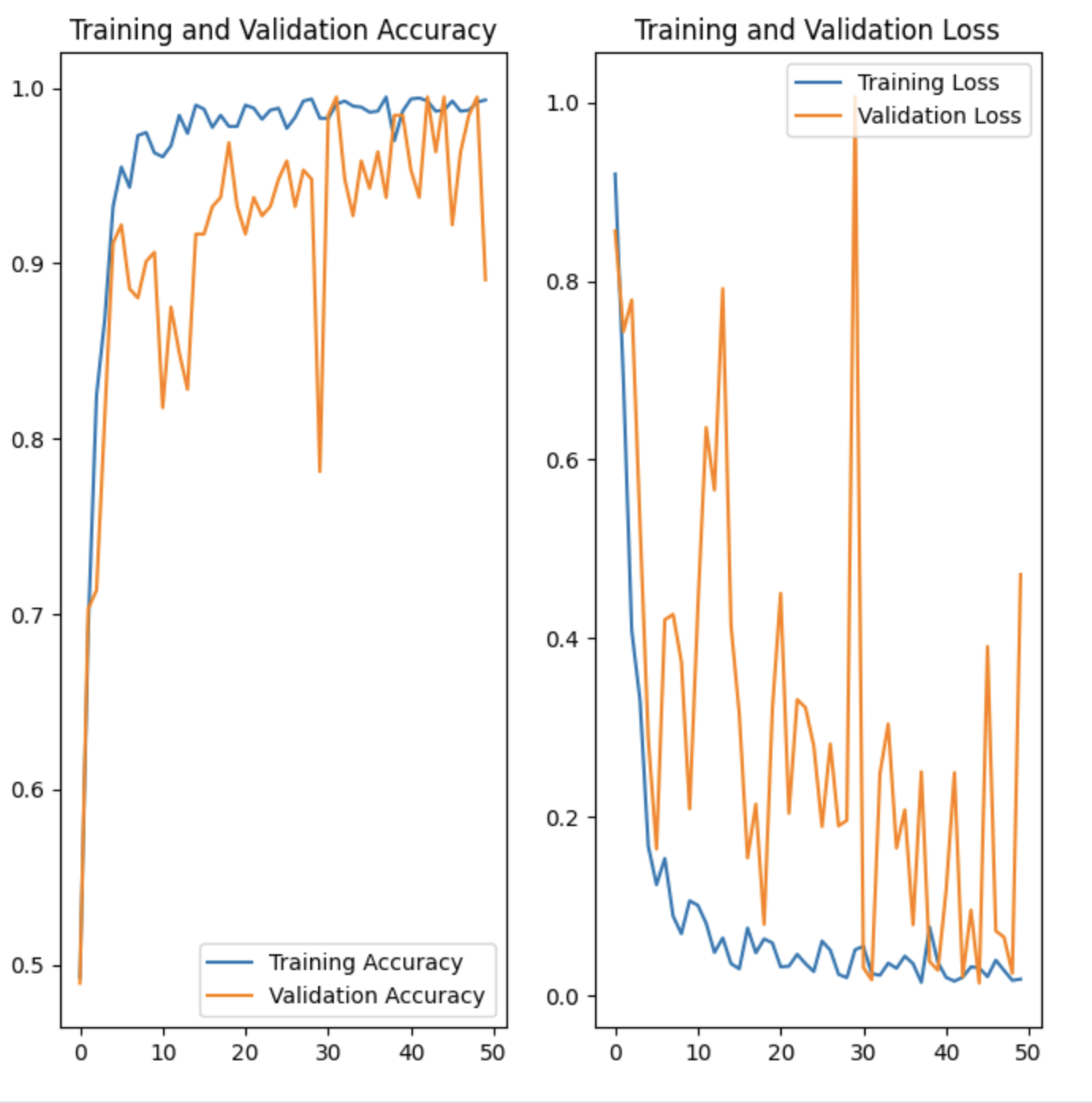
****

****

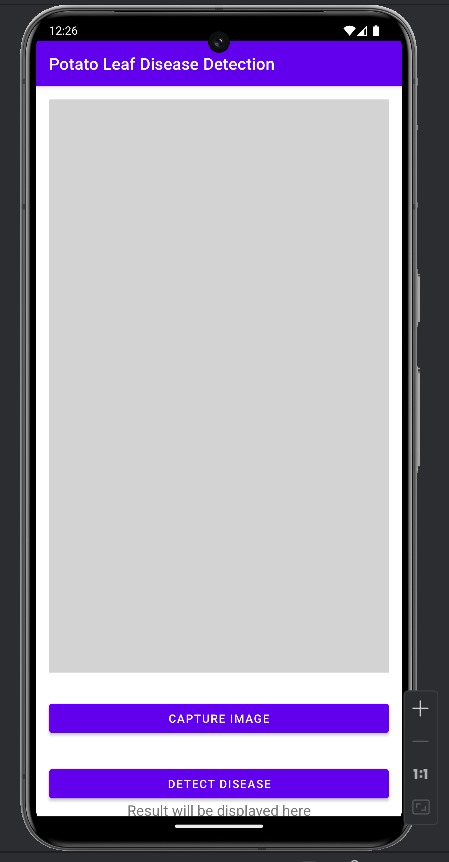
**Model 2**: Convolutional Neural Network with Batch Normalization and Dropout

****

****

****

**Android studio project design**

****