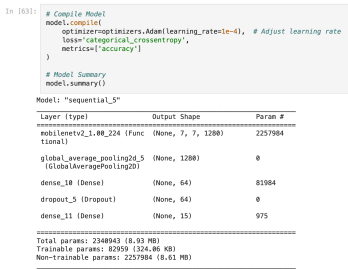
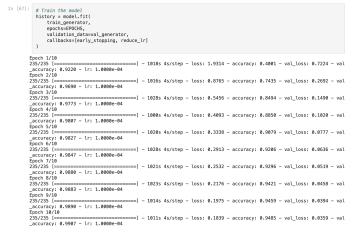


## Project Development Phase Model Performance Test

Date	10 November 2022
Team ID	PNT2022TMID592620
Project Name	GreenClassify: Deep Learning-based Approach for Vegetable Image Classification
Maximum Marks	10 Marks

### Model Performance Testing:

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

S.No.	Parameter	Values	Screenshot
1.	Model Summary	<p>This summary provides an overview of the layers in our model, along with the number of parameters in each layer. It looks like we have a MobileNetV2 base model followed by a global average pooling layer, a dense layer with 64 units and ReLU activation, a dropout layer with a dropout rate of 0.5, and a final dense layer with 15 units (assuming one for each class) and softmax activation. The model summary also shows the total number of parameters, trainable parameters, and non-trainable parameters.</p>	 <pre> In [1031]: # Compile Model model.compile(     optimizer=optimizers.Adam(learning_rate=0.1), # Adjust learning rate     loss='categorical_crossentropy',     metrics=['accuracy'])  # Model Summary model.summary()  Model: "sequential_5" Layer (type)                 Output Shape              Param # ----- mobile_net_v2_1_08_224 (Func  dense_11_1 (Dense)         975 tion) global_average_pooling2d_5   (None, 1280)              0 (global_average_pooling2d) dense_18 (Dense)              (None, 64)                 81984 dropout_5 (Dropout)           (None, 64)                 0 dense_11 (Dense)              (None, 15)                 975  Total params: 2348943 (8.93 MB) Trainable params: 82950 (324.40 KB) Non-trainable params: 226704 (8.92 MB) </pre>
2.	Accuracy	<p>Training Accuracy - (Epoch 10): 94.85%</p> <p>Validation Accuracy - (Epoch 10): 99.07%</p>	 <pre> In [1031]: # Train the model history = model.fit(     train_generator,     validation_data=train_generator,     callbacks=[EarlyStopping(monitor='val_loss', patience=10)],     epochs=10)  Epoch 1/10 20000/20000 [100%] 1.0000s - loss: 1.0000 - accuracy: 0.0000 - val_loss: 0.7000 - val_accuracy: 0.0000 Epoch 2/10 20000/20000 [100%] 1.0000s - loss: 0.5000 - accuracy: 0.5000 - val_loss: 0.3000 - val_accuracy: 0.5000 Epoch 3/10 20000/20000 [100%] 1.0000s - loss: 0.3000 - accuracy: 0.7000 - val_loss: 0.1000 - val_accuracy: 0.9000 Epoch 4/10 20000/20000 [100%] 1.0000s - loss: 0.1000 - accuracy: 0.9000 - val_loss: 0.0500 - val_accuracy: 0.9500 Epoch 5/10 20000/20000 [100%] 1.0000s - loss: 0.0500 - accuracy: 0.9500 - val_loss: 0.0200 - val_accuracy: 0.9800 Epoch 6/10 20000/20000 [100%] 1.0000s - loss: 0.0200 - accuracy: 0.9800 - val_loss: 0.0100 - val_accuracy: 0.9900 Epoch 7/10 20000/20000 [100%] 1.0000s - loss: 0.0100 - accuracy: 0.9900 - val_loss: 0.0050 - val_accuracy: 0.9950 Epoch 8/10 20000/20000 [100%] 1.0000s - loss: 0.0050 - accuracy: 0.9950 - val_loss: 0.0020 - val_accuracy: 0.9980 Epoch 9/10 20000/20000 [100%] 1.0000s - loss: 0.0020 - accuracy: 0.9980 - val_loss: 0.0010 - val_accuracy: 0.9990 Epoch 10/10 20000/20000 [100%] 1.0000s - loss: 0.0010 - accuracy: 0.9990 - val_loss: 0.0005 - val_accuracy: 0.9995 </pre>