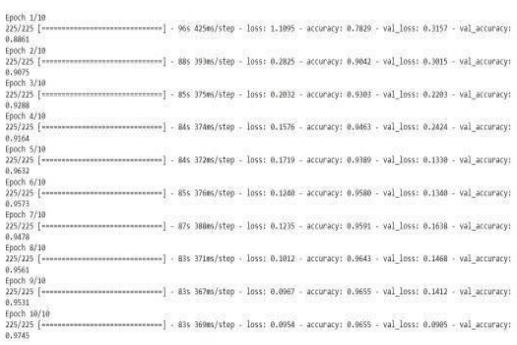
Project Development Phase
Model Performance Test

Date	18-11-2022
Team ID	PNT2022TMID42312
Project Name	Fertilizers Recommendation System for Disease Prediction
Maximum Marks	8mark

Model Performance Testing:

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

S.No.	Parameter	Values	Screenshot
1.	Model Summary	Total params: 5,082,202 Tainable params: 5,082,202 Non-trainable params: 0	`model.summary()` Model: "sequential_1" Layer (type) — Output Shape — Param # conv2d_6 (Conv2D) — (None, 126, 126, 32) — 896 max_pooling2d_2 (MaxPooling2D) — (None, 63, 63, 32) — 0 flatten_2 (Flatten) — (None, 127008) — 0 dense_6 (Dense) — (None, 40) — 5080360 dense_7 (Dense) — (None, 20) — 820 dense_8 (Dense) — (None, 6) — 126 Total params: 5,082,202 Trainable params: 5,082,202 Non-trainable params: 0
2.	Accuracy	Training Accuracy – 96.55 Validation Accuracy – 97.45	

Model Summary

```
model.summary()
```

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_6 (Conv2D)           (None, 126, 126, 32)      896

 max_pooling2d_2 (MaxPooling  (None, 63, 63, 32)       0
 2D)

 flatten_2 (Flatten)         (None, 127008)            0

 dense_6 (Dense)             (None, 40)                5080360

 dense_7 (Dense)             (None, 20)                820

 dense_8 (Dense)             (None, 6)                 126

=================================================================
Total params: 5,082,202
Trainable params: 5,082,202
Non-trainable params: 0
_________________________________________________________________
```

Accuracy

```
model.fit_generator(x_train,steps_per_epoch=len(x_train),validation_data=x_test,validation_steps=len(x_test),epochs=10)
```

```
C:\Users\Sree Ram\AppData\Local\Temp\ipykernel_13228\1582812018.py:1: UserWarning: `Model.fit_generator` is deprecated and will
be removed in a future version. Please use `Model.fit`, which supports generators.
  model.fit_generator(x_train,steps_per_epoch=len(x_train),validation_data=x_test,validation_steps=len(x_test),epochs=10)
Epoch 1/10
225/225 [==============================] - 96s 425ms/step - loss: 1.1095 - accuracy: 0.7829 - val_loss: 0.3157 - val_accuracy:
0.8861
Epoch 2/10
225/225 [==============================] - 88s 393ms/step - loss: 0.2825 - accuracy: 0.9042 - val_loss: 0.3015 - val_accuracy:
0.9075
Epoch 3/10
225/225 [==============================] - 85s 375ms/step - loss: 0.2032 - accuracy: 0.9303 - val_loss: 0.2203 - val_accuracy:
0.9288
Epoch 4/10
225/225 [==============================] - 84s 374ms/step - loss: 0.1576 - accuracy: 0.9463 - val_loss: 0.2424 - val_accuracy:
0.9164
Epoch 5/10
225/225 [==============================] - 84s 372ms/step - loss: 0.1719 - accuracy: 0.9389 - val_loss: 0.1330 - val_accuracy:
0.9632
Epoch 6/10
225/225 [==============================] - 85s 376ms/step - loss: 0.1240 - accuracy: 0.9580 - val_loss: 0.1340 - val_accuracy:
0.9573
Epoch 7/10
225/225 [==============================] - 87s 388ms/step - loss: 0.1235 - accuracy: 0.9591 - val_loss: 0.1638 - val_accuracy:
0.9478
Epoch 8/10
225/225 [==============================] - 83s 371ms/step - loss: 0.1012 - accuracy: 0.9643 - val_loss: 0.1468 - val_accuracy:
0.9561
Epoch 9/10
225/225 [==============================] - 83s 367ms/step - loss: 0.0967 - accuracy: 0.9655 - val_loss: 0.1412 - val_accuracy:
0.9531
Epoch 10/10
225/225 [==============================] - 83s 369ms/step - loss: 0.0954 - accuracy: 0.9655 - val_loss: 0.0905 - val_accuracy:
0.9745
```