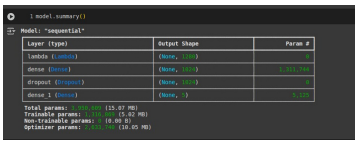
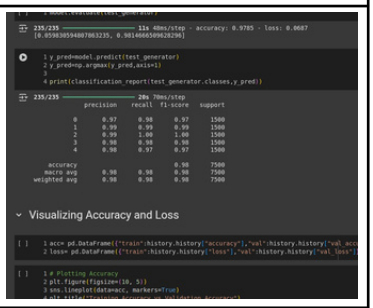
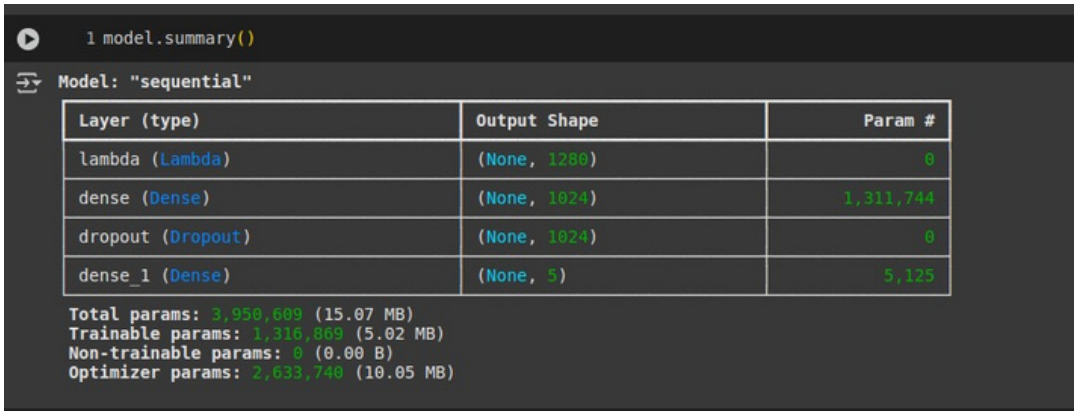
Project Development Phase
Model Performance Test

Date	26 june 2025
Team ID	
Project Name	GrainPalette A Deep Learning Odyssey In Rice Type Classification Through Transfer Learning
Maximum Marks	

Model Performance Testing:

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

S.No.	Parameter	Values	Screenshot
1.	Model Summary	Sequential Model Architecture: Lambda Layer, Dense Layer, Dropout Layer, Dense Layer.	
2.	Accuracy	Training Accuracy - 97.85% Validation Accuracy -98%	
3.	Fine Tunning Result(if Done)	Validation Accuracy - N/A	N/A



```
1 model.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Param #
lambda (Lambda)	(None, 1280)	0
dense (Dense)	(None, 1024)	1,311,744
dropout (Dropout)	(None, 1024)	0
dense_1 (Dense)	(None, 5)	5,125

Total params: 3,950,609 (15.07 MB)
Trainable params: 1,316,869 (5.02 MB)
Non-trainable params: 0 (0.00 B)
Optimizer params: 2,633,740 (10.05 MB)

```
[ ]  1 model.evaluate(test_generator)

235/235 ━━━━━━━━━━━━━━━━━━━━ 11s 48ms/step - accuracy: 0.9785 - loss: 0.0687
[0.059830594807863235, 0.9814666509628296]
```

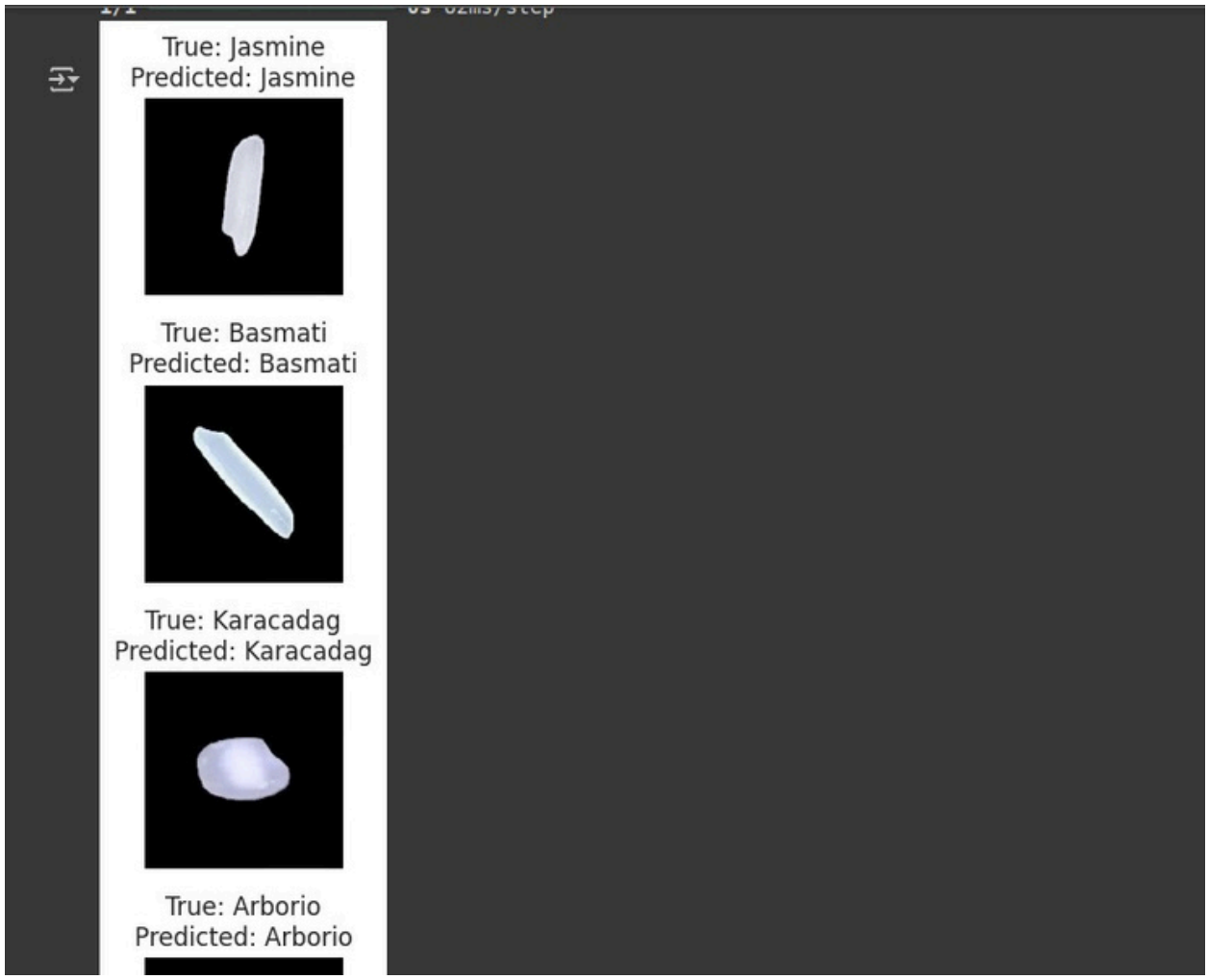
```
1 y_pred=model.predict(test_generator)
2 y_pred=np.argmax(y_pred,axis=1)
3
4 print(classification_report(test_generator.classes,y_pred))
```

```
235/235 ━━━━━━━━━━━━━━━━━━━━ 20s 70ms/step
              precision    recall  f1-score   support

           0       0.97      0.98      0.97      1500
           1       0.99      0.99      0.99      1500
           2       0.99      1.00      1.00      1500
           3       0.98      0.98      0.98      1500
           4       0.98      0.97      0.97      1500

    accuracy                           0.98      7500
   macro avg       0.98      0.98      0.98      7500
weighted avg       0.98      0.98      0.98      7500
```

Visualizing Accuracy and Loss

```
[ ]  1 acc= pd.DataFrame({"train":history.history["accuracy"],"val":history.history["val_accu
     2 loss= pd.DataFrame({"train":history.history["loss"],"val":history.history["val_loss"]}
```

```
[ ]  1 # Plotting Accuracy
     2 plt.figure(figsize=(10, 5))
     3 sns.lineplot(data=acc, markers=True)
     4 plt.title("Training Accuracy vs Validation Accuracy")
```


True: Jasmine
Predicted: Jasmine
True: Basmati
Predicted: Basmati
True: Karacadag
Predicted: Karacadag
True: Arborio
Predicted: Arborio