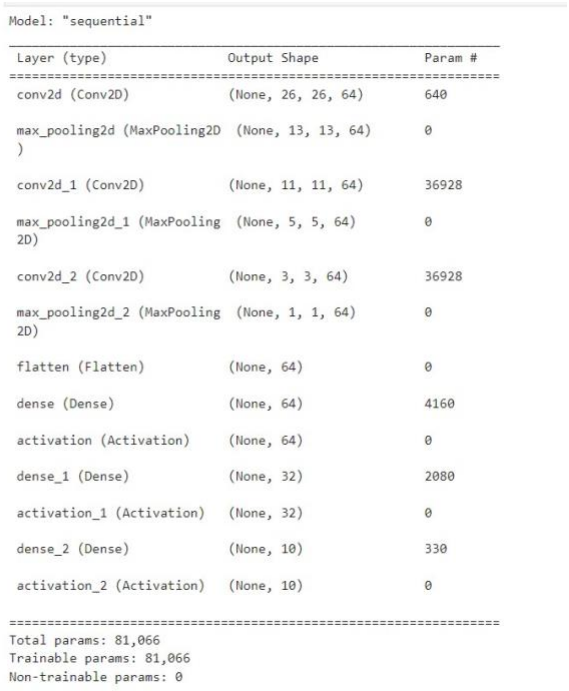
Project Development Phase
Model Performance Test

Date	19 NOvember 2022
Team ID	PNT2022TMID40505
Project Name	Project – A novel method for handwritten digit recognition system.
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.	Metrics	**Regression Model:** model summary	

2.	Accuracy	Accuracy of 99.21% is achieved.	Epoch 1/10 1125/1125 [==========] - 43s 37ms/step - loss: 0.3674 - accuracy: 0.8839 - val_loss: 0.1889 - val_accuracy: 0.9452 Epoch 2/10 1125/1125 [==========] - 44s 39ms/step - loss: 0.1151 - accuracy: 0.9643 - val_loss: 0.1069 - val_accuracy: 0.9679 Epoch 3/10 1125/1125 [==========] - 42s 38ms/step - loss: 0.0821 - accuracy: 0.9745 - val_loss: 0.0724 - val_accuracy: 0.9774 Epoch 4/10 1125/1125 [==========] - 43s 38ms/step - loss: 0.0639 - accuracy: 0.9803 - val_loss: 0.0748 - val_accuracy: 0.9774 Epoch 5/10 1125/1125 [==========] - 43s 38ms/step - loss: 0.0512 - accuracy: 0.9839 - val_loss: 0.0748 - val_accuracy: 0.9762 Epoch 6/10 1125/1125 [==========] - 46s 41ms/step - loss: 0.0431 - accuracy: 0.9860 - val_loss: 0.0760 - val_accuracy: 0.9782 Epoch 7/10 1125/1125 [==========] - 60s 53ms/step - loss: 0.0347 - accuracy: 0.9884 - val_loss: 0.0810 - val_accuracy: 0.9786 Epoch 8/10 1125/1125 [==========] - 64s 57ms/step - loss: 0.0282 - accuracy: 0.9909 - val_loss: 0.0697 - val_accuracy: 0.9809 Epoch 9/10 1125/1125 [==========] - 52s 46ms/step - loss: 0.0243 - accuracy: 0.9919 - val_loss: 0.0730 - val_accuracy: 0.9797 Epoch 10/10 1125/1125 [==========] - 42s 37ms/step - loss: 0.0229 - accuracy: 0.9921 - val_loss: 0.0620 - val_accuracy: 0.9837 Out[26]:

3.	Tune the model	Dataset is tested and digits are recognized.	In [101…
			`img = Image.open(streaming_body_1).convert("L")`
			`img = img.resize( (28,28) )`
			In [102…
			`img`
			Out[102…
			 
			In [103…
			`im2arr = np.array(img)`
			`im2arr = im2arr.reshape(1, 28, 28, 1)`
			In [104…
			`pred = model.predict(im2arr)`
			`print(pred)`
			`[[0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]]`
			In [105…
			`print(np.argmax(pred, axis=1))`
			`[8]`
			In [ ]: