

TEAM ID	PNT2022TMID51228
PROJECT NAME	REAL TIME COMMUNICATION SYSTEM FOR SPECIALLY AIDED PEOPLE
WORK	PERFORMANCE TESTING
DATE	18.11.2022

## Locust Test Report

During: 11/17/2022, 5:24:47 PM - 11/17/2022, 5:34:15 PM

Target Host: http://127.0.0.1:5000

Script: locustfile.py

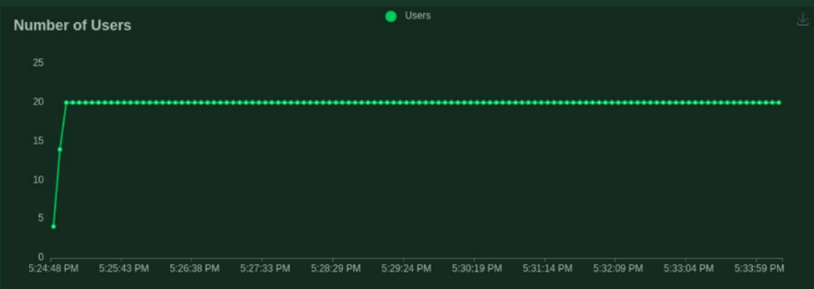
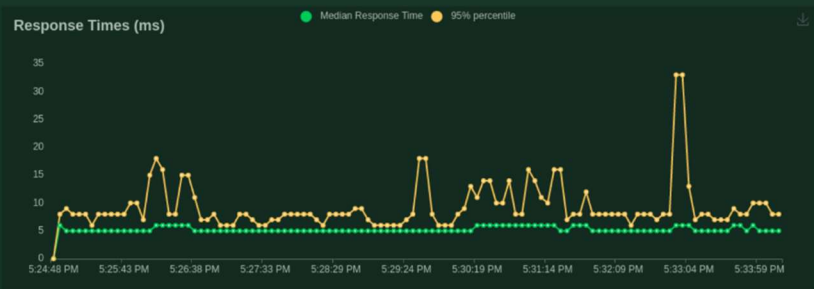
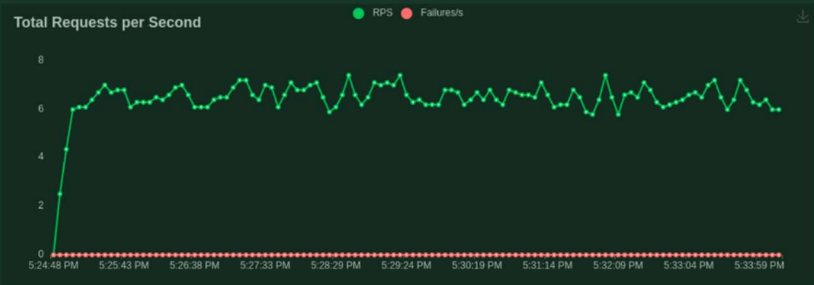
### Request Statistics

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/	1890	0	5	4	41	6381	3.3	0.0
GET	/prediction	1028	0	5	4	34	4404	3.2	0.0
Aggregated		3718	0	5	4	41	5448	6.5	0.0

### Response Time Statistics

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
GET	/	5	5	6	6	7	9	19	41
GET	/prediction	5	5	6	6	7	9	19	34
Aggregated		5	5	6	6	7	9	19	41

### Charts



### Final ratio

#### Ratio per User class

- 100.0% AppUser
  - 50.0% home
  - 50.0% prediction

#### Total ratio

- 100.0% AppUser
  - 50.0% home
  - 50.0% prediction

## PERFORMANCE ACCURACY:

The screenshot shows a Jupyter Notebook interface with a code cell containing a `model.fit` call. The output displays training progress for epochs 1 through 9. Each epoch's output includes the number of steps (158/158), time per step, loss, accuracy, validation loss, and validation accuracy.

```
In [17]: model.fit(
          xtrain, steps_per_epoch=len(xtrain),
          epochs=20, validation_data=xtest, validation_steps=len(xtest))
```

Epoch 1/20  
158/158 [=====] - 449s 3s/step - loss: 0.0439 - accuracy: 0.7561 - val\_loss: 0.0066 - val\_accuracy: 0.9676  
Epoch 2/20  
158/158 [=====] - 167s 1s/step - loss: 0.0023 - accuracy: 0.9872 - val\_loss: 0.0059 - val\_accuracy: 0.9707  
Epoch 3/20  
158/158 [=====] - 165s 1s/step - loss: 9.7036e-04 - accuracy: 0.9950 - val\_loss: 0.0057 - val\_accuracy: 0.9702  
Epoch 4/20  
158/158 [=====] - 175s 1s/step - loss: 5.8128e-04 - accuracy: 0.9968 - val\_loss: 0.0051 - val\_accuracy: 0.9760  
Epoch 5/20  
158/158 [=====] - 278s 2s/step - loss: 4.2848e-04 - accuracy: 0.9977 - val\_loss: 0.0052 - val\_accuracy: 0.9764  
Epoch 6/20  
158/158 [=====] - 236s 1s/step - loss: 2.5589e-04 - accuracy: 0.9987 - val\_loss: 0.0050 - val\_accuracy: 0.9760  
Epoch 7/20  
158/158 [=====] - 223s 1s/step - loss: 2.8742e-04 - accuracy: 0.9987 - val\_loss: 0.0046 - val\_accuracy: 0.9787  
Epoch 8/20  
158/158 [=====] - 1184s 8s/step - loss: 3.3162e-04 - accuracy: 0.9982 - val\_loss: 0.0039 - val\_accuracy: 0.9818  
Epoch 9/20  
158/158 [=====] - 170s 1s/step - loss: 3.4941e-04 - accuracy: 0.9982 - val\_loss: 0.0054 - val\_accuracy: 0.9822

The screenshot shows the continuation of the Jupyter Notebook output, displaying training progress for epochs 10 through 20. The format of the output remains consistent with the previous epochs.

Epoch 10/20  
158/158 [=====] - 170s 1s/step - loss: 3.4941e-04 - accuracy: 0.9982 - val\_loss: 0.0054 - val\_accuracy: 0.9822  
Epoch 11/20  
158/158 [=====] - 175s 1s/step - loss: 2.4112e-04 - accuracy: 0.9987 - val\_loss: 0.0047 - val\_accuracy: 0.9773  
Epoch 12/20  
158/158 [=====] - 205s 1s/step - loss: 2.6106e-04 - accuracy: 0.9987 - val\_loss: 0.0050 - val\_accuracy: 0.9773  
Epoch 13/20  
158/158 [=====] - 224s 1s/step - loss: 2.0397e-04 - accuracy: 0.9990 - val\_loss: 0.0064 - val\_accuracy: 0.9689  
Epoch 14/20  
158/158 [=====] - 222s 1s/step - loss: 1.5883e-04 - accuracy: 0.9991 - val\_loss: 0.0050 - val\_accuracy: 0.9769  
Epoch 15/20  
158/158 [=====] - 217s 1s/step - loss: 2.2415e-04 - accuracy: 0.9988 - val\_loss: 0.0050 - val\_accuracy: 0.9773  
Epoch 16/20  
158/158 [=====] - 262s 2s/step - loss: 1.7302e-04 - accuracy: 0.9989 - val\_loss: 0.0050 - val\_accuracy: 0.9769  
Epoch 17/20  
158/158 [=====] - 230s 1s/step - loss: 2.5752e-04 - accuracy: 0.9984 - val\_loss: 0.0050 - val\_accuracy: 0.9773  
Epoch 18/20  
158/158 [=====] - 176s 1s/step - loss: 1.5972e-04 - accuracy: 0.9990 - val\_loss: 0.0050 - val\_accuracy: 0.9764  
Epoch 19/20  
158/158 [=====] - 172s 1s/step - loss: 2.2151e-04 - accuracy: 0.9987 - val\_loss: 0.0050 - val\_accuracy: 0.9773  
Epoch 20/20  
158/158 [=====] - 167s 1s/step - loss: 1.2325e-04 - accuracy: 0.9992 - val\_loss: 0.0049 - val\_accuracy: 0.9778

## Model Summary

```
In [41]: model.summary()
```

```
Model: "sequential_2"
```

Layer (type)	Output Shape	Param #
=====		
conv2d_5 (Conv2D)	(None, 126, 126, 32)	896
max_pooling2d_2 (MaxPooling 2D)	(None, 63, 63, 32)	0
flatten_2 (Flatten)	(None, 127008)	0
dense_6 (Dense)	(None, 40)	5080360
dense_7 (Dense)	(None, 70)	2870
dense_8 (Dense)	(None, 6)	426
=====		
Total params: 5,084,552		
Trainable params: 5,084,552		
Non-trainable params: 0		