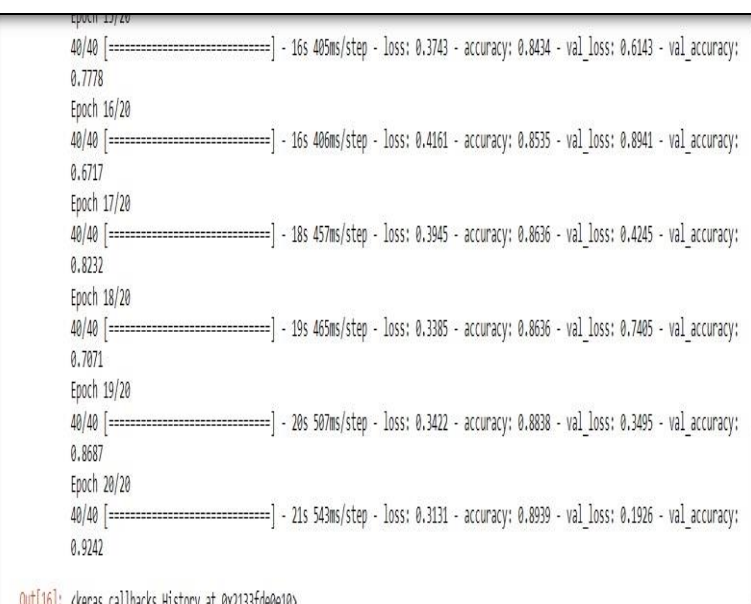


## Project Development Phase Model Performance Test

Date	20 November 2022
Team ID	PNT2022TMID25918
Project Name	Natural Disasters Intensity Analysis and Classification Using Artificial Intelligence
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	<b>Deep Learning Model – CNN</b> Accuracy through epoch	 <pre> Epoch 15/20 40/40 [=====] - 16s 405ms/step - loss: 0.3743 - accuracy: 0.8434 - val_loss: 0.6143 - val_accuracy: 0.7778 Epoch 16/20 40/40 [=====] - 16s 406ms/step - loss: 0.4161 - accuracy: 0.8535 - val_loss: 0.8941 - val_accuracy: 0.6717 Epoch 17/20 40/40 [=====] - 18s 457ms/step - loss: 0.3945 - accuracy: 0.8636 - val_loss: 0.4245 - val_accuracy: 0.8232 Epoch 18/20 40/40 [=====] - 19s 465ms/step - loss: 0.3385 - accuracy: 0.8636 - val_loss: 0.7405 - val_accuracy: 0.7071 Epoch 19/20 40/40 [=====] - 20s 507ms/step - loss: 0.3422 - accuracy: 0.8838 - val_loss: 0.3495 - val_accuracy: 0.8687 Epoch 20/20 40/40 [=====] - 21s 543ms/step - loss: 0.3131 - accuracy: 0.8939 - val_loss: 0.1926 - val_accuracy: 0.9242  Out[16]: &lt;keras.callbacks.History at 0x2133fde0e10&gt; </pre>

2.	Preprocess and Train the model	Using data augmentation and train test split method	<pre> In [ ]: #performing data agumentation to train data x_train = train_datagen.flow_from_directory(r'C:\Users\hp\Desktop\IBM\dataset\test_set',target_size=(64, 64),  color_mode='rgb',class_mode='categorical')  #performing data agumentation to test data x_test = test_datagen.flow_from_directory(r'C:\Users\hp\Desktop\IBM\dataset\test_set',target_size=(64, 64),   color_mode='rgb',class_mode='categorical')  Found 198 images belonging to 4 classes. Found 198 images belonging to 4 classes.  In [ ]: print(x_train.class_indices)#checking the number of classes  {'Cyclone': 0, 'Earthquake': 1, 'Flood': 2, 'Wildfire': 3}  In [ ]: print(x_test.class_indices)#checking the number of classes  {'Cyclone': 0, 'Earthquake': 1, 'Flood': 2, 'Wildfire': 3}  In [ ]: from collections import Counter as c c(x_train .labels)  Out[11]: Counter({0: 64, 1: 29, 2: 61, 3: 44}) </pre>
3.	Test the model	By loading the pretrained model and predicting the results	<pre> In [17]: from tensorflow.keras.models import load_model from tensorflow.keras.utils import load_img,img_to_array model = load_model("/content/disaster.h5") #Loading the model for testing  In [18]: img = load_img(r"/content/e2.jpg",grayscale=False,target_size= (64,64))#Loading of the image x = img_to_array(img)#image to array x = np.expand_dims(x,axis = 0)#changing the shape #pred = classifier.predict_classes(x)#predicting the classes predict=model.predict(x) classes_x=np.argmax(predict,axis=1) classes_x  1/1 [=====] - 0s 394ms/step  Out[18]: array([1])  In [ ]:  In [19]: index=['Cyclone','Earthquake','Flood','Wildfire'] result=str(index[classes_x[0]]) result  Out[19]: 'Earthquake' </pre>