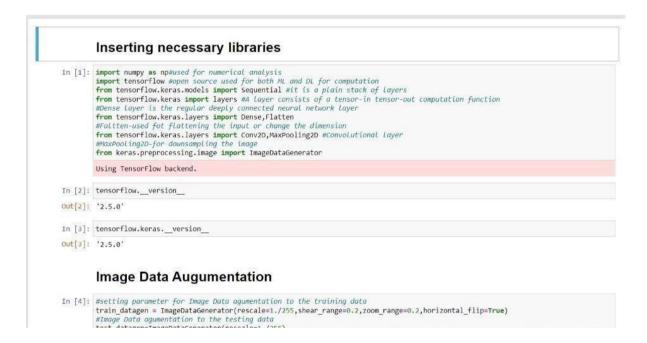
PROJECT DEVELOPMENT PHASE

SPRINT-III

Date	10 November 2022
Team ID	PNT2022TMID37914
Project Name	Natural Disaster Intensity Analysis and
	Classification using Artificial
	Intelligence

DETECTION AND ANALYSIS OF DATA:

After Testing and Training the model, data which given in dataset are analysed and visualised effectively to detect the Disaster Type. Using webcam, it can capture image or video stream of Disaster, to detect and analyse the type of Disaster.



MODEL BUILDING:

Building a Model with web application named "FLASK", model building process consist several steps like,

Import the model building Libraries

- Initializing the model
- Adding CNN Layers
- Adding Hidden Layer
- · Adding Output Layer
- · Configure the Learning Process
- · Training and testing the model

all the above processes are done and saved in a model.

```
In [1]: import numpy as np#used for numerical analysis
import tensorflow #open source used for both ML and DL for computation
from tensorflow keras.models import Sequential #it is a plain stack of layers
from tensorflow.keras.models import sequential #it is a plain stack of layers
from tensorflow.keras.models import Sequential #it is a plain stack of layers
from tensorflow.keras.import Dense,Flatten
#Entten-used for Italtening the input or change the dimension
from tensorflow.keras.layers import ConvDn,MaxPooling2D #Convolutional layer
#MaxPooling2D-for downsompling the image
from keras.preprocessing.image import TmageDataGenerator
Using Tensorflow backend.

In [2]: tensorflow.version_
out[2]: '2.5.0'

Image Data Augumentation

In [4]: #setting parameter for Image Data agumentation to the training data
train_datagen = ImageDataGenerator(rescale=1./255, shear_range=0.2, horizontal_flip=True)
#Image Data agumentation to the testing data
train_datagen = ImageDataGenerator(rescale=1./255, shear_range=0.2, zoom_range=0.2, horizontal_flip=True)
#Image Data agumentation to the testing data
test_datagen=ImageDataGenerator(rescale=1./255, shear_range=0.2, zoom_range=0.2, horizontal_flip=True)
#Image Data agumentation to the testing data
test_datagen=ImageDataGenerator(rescale=1./255)
```