PROJECT DEVELOPMENT PHASE

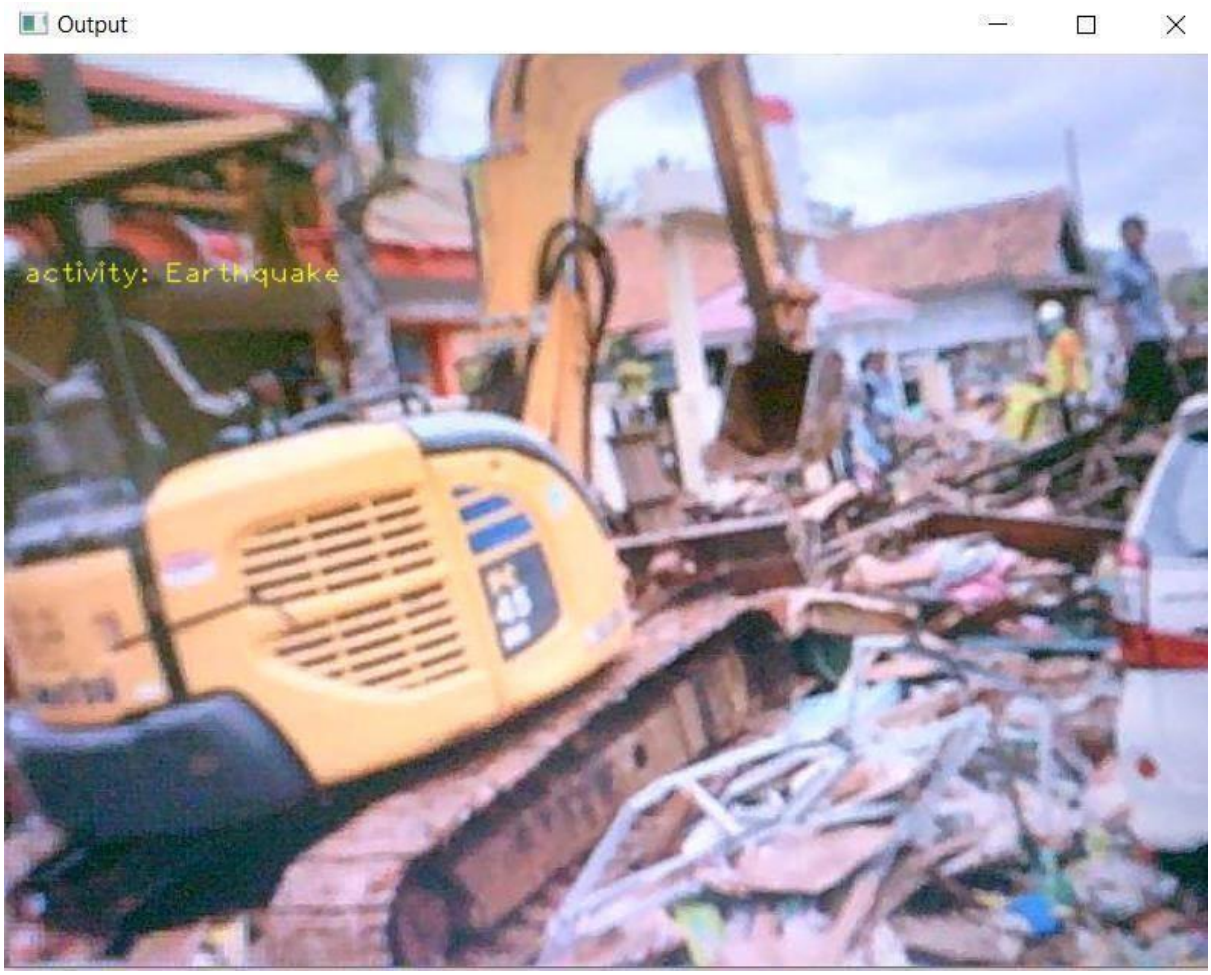
SPRINT-4

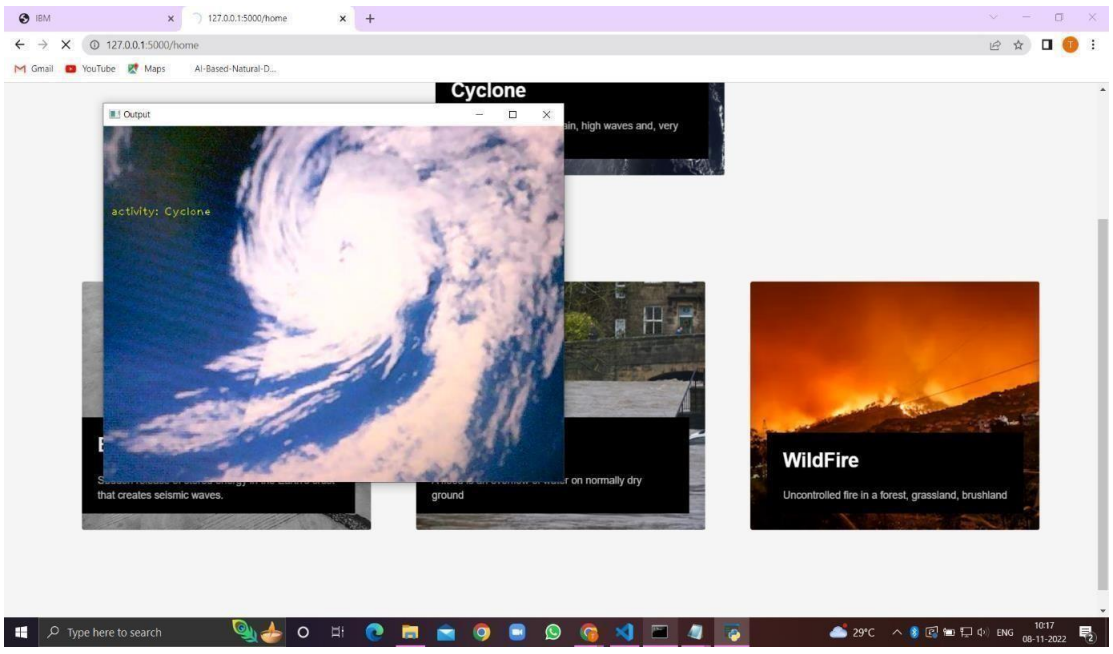
Date	15 November 2022
Team ID	PNT2022TMID12341
Project Name	Natural Disaster Intensity Analysis and Classification using Artificial Intelligence

INTEGRATE THE WEB APP WITH AI MODEL:

After creating the Model, the Model should be integrated with the web app using the Flask application. The coding part is named as app.py and it will be running in the localhost through the generated link. By navigating the localhost the webpage will be visible.

```
            output = frame.copy()
            #print("apple")
            frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
            frame = cv2.resize(frame, (64, 64))
            #frame = frame.astype("float32")
            x=np.expand_dims(frame, axis=0)
            result = np.argmax(model.predict(x), axis=-1)
            index=['Cyclone','Earthquake','Flood','Wildfire']
            result=str(index[result[0]])
            #print(result)
            #result=result.tolist()

            cv2.putText(output, "activity: {}".format(result), (10, 120), cv2.FONT_HERSHEY_PLAIN,
                        1, (0,255,255), 1)
            #playaudio("Emergency it is a disaster")
            cv2.imshow("Output", output)
            key = cv2.waitKey(1) & 0xFF

        # if the `q` key was pressed, break from the loop
            if key == ord("q"):
                break

        # release the file pointers
        print("[INFO] cleaning up...")
        vs.release()
        cv2.destroyAllWindows()
        return render_template("upload.html")


if __name__ == '__main__':
      app.run(debug=False,threaded=True)
```


Output
activity: Earthquake


IBM
127.0.0.1:5000/home
127.0.0.1:5000/home
Gmail
YouTube
Maps
AI-Based-Natural-D...
Cyclone
Output
activity: Cyclone
that creates seismic waves.
WildFire
Uncontrolled fire in a forest, grassland, brushland
Type here to search
29°C
ENG
10:17
08-11-2022

MODEL DEPLOYMENT:

The trained model which is running in the localhost without any error is deployed in the IBM Cloud for making available for the users to predict the Disaster's type and its intensity. It is integrated with the Flask application.