

## PROJECT DEVELOPMENT PHASE

### SPRINT-4

Date	07 November 2022
Team ID	PNT2022TMID50358
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 local host through the generated link. By navigating the local host the webpage will be visible.

```
73         output = frame.copy()
74         #print("apple")
75         frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
76         frame = cv2.resize(frame, (64, 64))
77         #frame = frame.astype("float32")
78         x=np.expand_dims(frame, axis=0)
79         result = np.argmax(model.predict(x), axis=-1)
80         index=['Cyclone', 'Earthquake', 'Flood', 'Wildfire']
81         result=str(index[result[0]])
82         #print(result)
83         #result=result.tolist()
84
85         cv2.putText(output, "activity: {}".format(result), (10, 120), cv2.FONT_HERSHEY_PLAIN,
86                     1, (0,255,255), 1)
87         #playaudio("Emergency it is a disaster")
88         cv2.imshow("Output", output)
89         key = cv2.waitKey(1) & 0xFF
90
91         # if the `q` key was pressed, break from the loop
92         if key == ord("q"):
93             break
94
95         # release the file pointers
96         print("[INFO] cleaning up...")
97         vs.release()
98         cv2.destroyAllWindows()
99         return render_template("upload.html")
100
101 if __name__ == '__main__':
102     app.run(debug=False, threaded=True)
```

Output



IBM x 127.0.0.1:5000/home x +

← → X 127.0.0.1:5000/home ☆ □ ⋮

Gmail YouTube Maps AI-Based-Natural-D...

### Cyclone

activity: Cyclone

activity: Earthquake

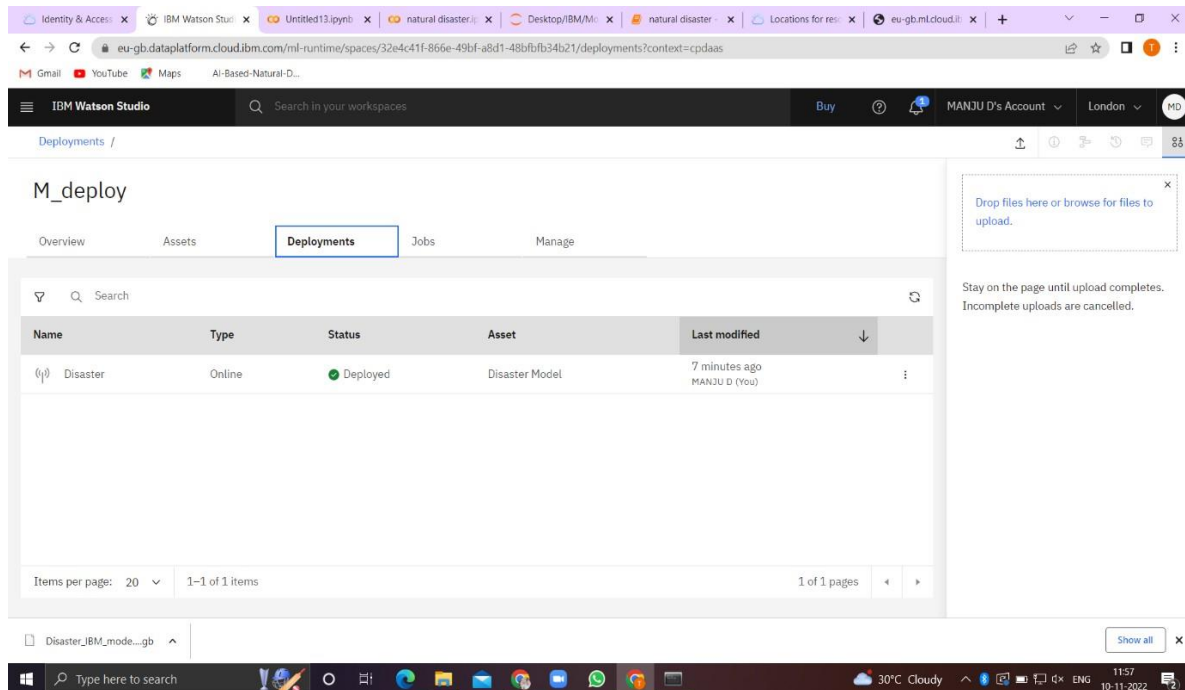
activity: WildFire

Uncontrolled fire in a forest, grassland, brushland

29°C 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.



The screenshot displays the IBM Watson Studio interface, specifically the 'Deployments' tab for a workspace named 'M\_deploy'. The interface shows a table with one deployment entry: 'Disaster' (Type: Online, Status: Deployed, Asset: Disaster Model, Last modified: 7 minutes ago by MANJU D (You)). The table has columns for Name, Type, Status, Asset, and Last modified. A search bar is located above the table. To the right of the table, there is a message: 'Drop files here or browse for files to upload.' and a note: 'Stay on the page until upload completes. Incomplete uploads are cancelled.' The bottom of the screen shows a Windows taskbar with various application icons and a system tray indicating 30°C, Cloudy, and the date 10-11-2022.

Name	Type	Status	Asset	Last modified
Disaster	Online	Deployed	Disaster Model	7 minutes ago MANJU D (You)