

PROJECT DEVELOPMENT PHASE

SPRINT-4

Date	07 November 2022
Team ID	PNT2022TMID50388
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 127.0.0.1:5000/home

127.0.0.1:5000/home

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

Cyclone

Output

activity: Cyclone

rain, high waves and, very

that creates seismic waves.

ground

WildFire

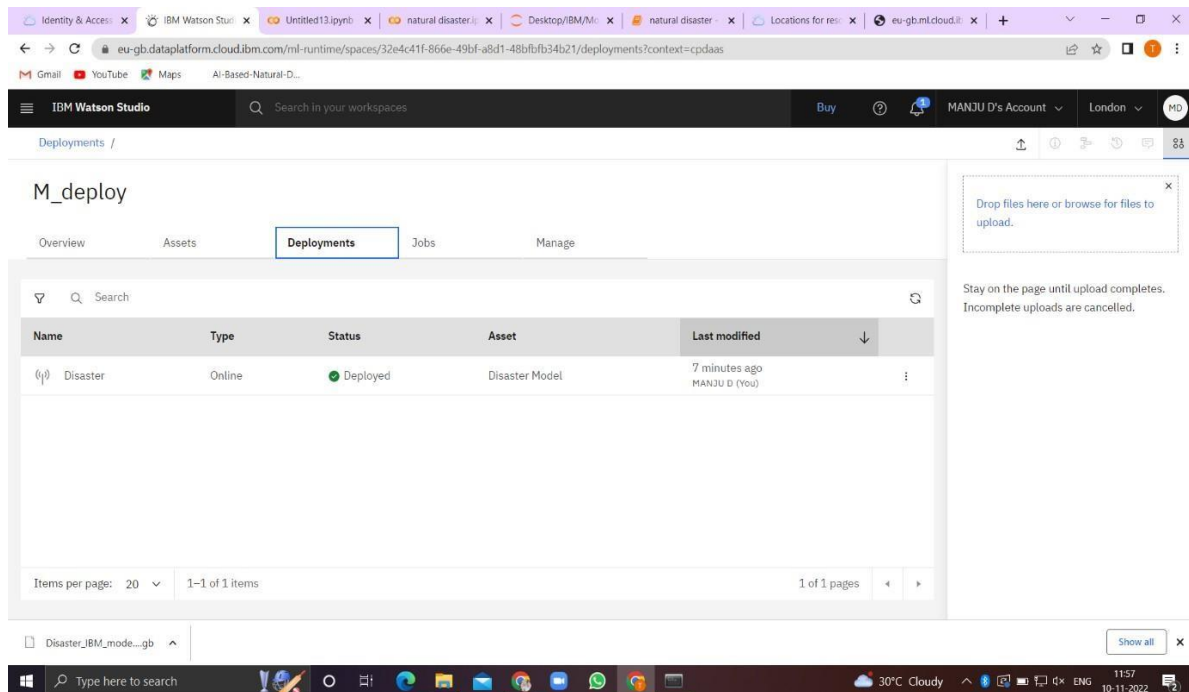
Uncontrolled fire in a forest, grassland, brushland

Type here to search 29°C 10:17 08-11-2022

The image shows a web application interface for natural disasters. It features a browser window with the URL "127.0.0.1:5000/home". The main content area displays three disaster-related images: a satellite view of a cyclone, a ground-level view of a cyclone's impact, and a photograph of a wildfire. Each image has a title and a brief description. The interface also includes a search bar and a system tray at the bottom showing the date and time.

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 shows the IBM Watson Studio interface in a web browser. The browser's address bar displays the URL: `eu-gb.dataplatform.cloud.ibm.com/ml-runtime/spaces/32e4c41f-866e-49bf-a8d1-48bfufb34b21/deployments?context=cpdaas`. The page title is "M_deploy". Below the title, there are tabs for "Overview", "Assets", "Deployments" (which is active), "Jobs", and "Manage". A search bar is located above a table of deployments. The table has columns for "Name", "Type", "Status", "Asset", and "Last modified". One deployment is listed: "Disaster" with Type "Online", Status "Deployed" (indicated by a green checkmark), Asset "Disaster Model", and Last modified "7 minutes ago MANJU D (You)". To the right of the table, there is a message box that says "Drop files here or browse for files to upload." and a note: "Stay on the page until upload completes. Incomplete uploads are cancelled." At the bottom of the page, there is a "Show all" button. The Windows taskbar at the bottom shows the time as 11:57 on 10-11-2022, with a temperature of 30°C and a cloudy weather forecast.

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