

Progress Report on AI-Driven Proactive Maintenance API Deployment

Introduction

This report's goal is to provide an overview of the developments made in the Flask deployment of the AI-Driven Proactive Maintenance API. This project aims to improve proactive maintenance techniques by offering a dependable way to forecast equipment failure based on IoT data.

Deployment Summary

I successfully deployed the Flask application on PythonAnywhere. Below are the key steps taken during the deployment process:

Account Creation: Created a free account on PythonAnywhere.

Web App Setup:

- Navigated to the "Web" tab and added a new web app.
- Selected the Flask framework and specified the Python version.

Code Upload:

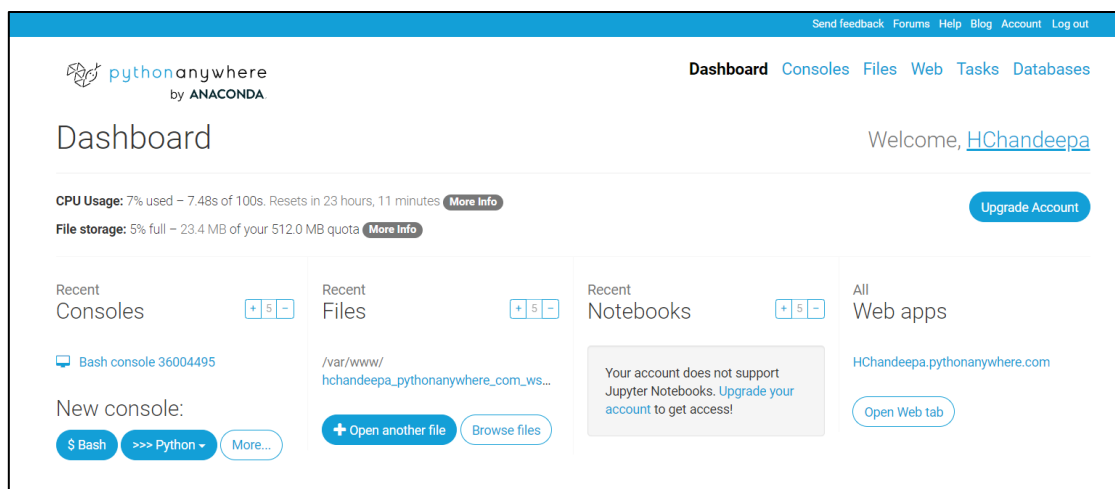
- Uploaded the app.py file and the random_forest_model.pkl model file.

Configuration:

- Ensured that the application was configured to run on the correct port and settings.

Testing:

- Verified the application was running by accessing the root URL, which displayed a welcome message.



pythonanywhere
by ANACONDA

HChandeepa.pythonanywhere.com

Add a new web app

Dashboard Consoles Files Web Tasks Databases

Configuration for HChandeepa.pythonanywhere.com

Reload:

Reload HChandeepa.pythonanywhere.com

Best before date:

We're happy to host your free website – and keep it free – for as long as you want to keep it running, but you'll need to log in at least once every three months and click the "Run until 3 months from today" button below. We'll send you an email a week before the site is disabled so that you don't forget to do that. [See here for more details.](#)

This site will be disabled on **Wednesday 08 January 2025**

Run until 3 months from today

Paying users' sites stay up forever without any need to log in to keep them running.

Traffic:

How busy is your site?

This month (previous month)	11	(0)
Today (yesterday)	11	(0)
Hour (previous hour)	9	(2)

← → ↺

hchandeepa.pythonanywhere.com

🗖️ | 📧 Gmail 📍 Maps 📺 New Tab 📺 (610) Image Detecti... 📖 Find your organizati... 🐙 Strok

Welcome to the AI-Driven Proactive Maintenance API

API Functionality

The API provides an endpoint for making predictions regarding equipment failure based on the following features: Temperature, Humidity, and HVAC_Status.

Endpoint Details

- **Root Route:** GET /
 - Returns a welcome message.
- **Prediction Route:** POST /predict

- **Input:** JSON format containing the features:

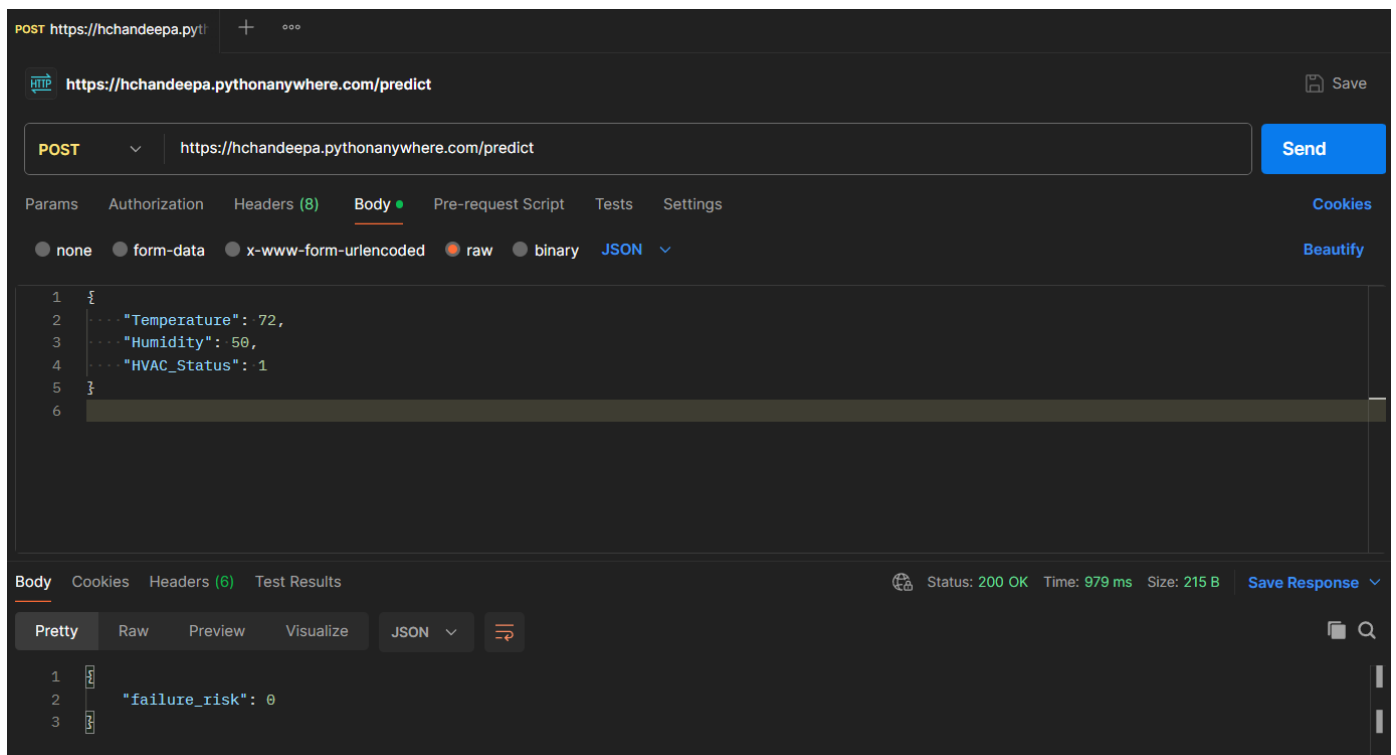
```
{  
  "Temperature": 72,  
  "Humidity": 50,  
  "HVAC_Status": 1  
}
```

- **Response:** JSON indicating the risk of failure:

```
{  
  "failure_risk": 0  
}
```

Testing

The API has been tested successfully using Postman, which confirmed that it responds correctly to valid input.



Connecting to IoT

The Flask API can be integrated with IoT devices to receive real-time data for making predictions. For example, a sample code snippet for sending data from an IoT device might look like this:

```
import requests

# Sample IoT data
data = {
    "Temperature": 72,
    "Humidity": 50,
    "HVAC_Status": 1
}

response = requests.post('https://hchandeepa.pythonanywhere.com/predict', json=data)
print(response.json())
```

This connection allows for immediate prediction based on live data, facilitating proactive maintenance actions.

Next

- Test the API with actual IoT data to ensure robustness and accuracy.
- Implement additional features such as data logging and user authentication for enhanced security.
- Monitor the API's performance and make necessary adjustments to improve efficiency.