SKY FUSION – Weather & Space Dashboard

Project Submission Report

Internship Program: Agnirva AI Internship (Summer 2025)

Participant: Priya Kumari

Duration: 8 weeks

Domain: Agnirva Software Intenship

# 1. Project Overview

SKY FUSION is an interactive, multi-feature web application that integrates real-time weather, space weather alerts, natural disaster monitoring, and NASA's Astronomy Picture of the Day (APOD) — all in one place. It offers users global insights through a visually dynamic, informative, and user-friendly dashboard.

# 2. Objectives

* Create a unified dashboard to monitor Earth & space activity
* Integrate real-time APIs for weather, disasters, solar storms, and APOD
* Use visual AI tools for dynamic background and icon generation
* Provide interactive UI/UX with dropdowns, charts, and tooltips
* Offer users both simplified and scientific information layers

# 3. Tools & Technologies Used

* HTML, CSS, JavaScript (Frontend)
* Flask, Python (Backend server)
* OpenWeather API, NASA APOD API, NOAA SWPC Alerts, Global Disaster Alerts
* GitHub for version control and project hosting

# 4. Features Implemented

* Weather Info : Get current weather by entering your city
* Space Weather : View solar events, flares, and magnetosphere data with visual alerts
* Natural Disasters : See real-time earthquakes, volcanic activity, wildfires, and more
* NASA APOD : Daily space photo with a scientific description
* Clean UI/UX : Each module has a themed page for clarity and engagement
* Modular Navigation : Easily switch between features with one click

# 5. Project Description

This project was built with the aim of creating a centralized platform where users can explore:

* Local weather forecasts
* Active space weather phenomena (like solar flares or geomagnetic storms)
* Ongoing natural disasters across the globe
* Daily featured space imagery and data from NASA

# 6. DEMO VIDEO LINK

CLICK HERE TO VIEW :

<https://drive.google.com/drive/folders/1qNv-Faae4kaQ12CW6E_r2_rcrQyor6g0>

# 7. Challenges & Solutions

**Challenge**

* Integrating multiple APIs with different response formats
* Real-time data rendering in JS & Flask
* Cross-origin data fetch errors
* Making deployment-ready Flask app setup

**Solutions which applied**

* Wrote custom parsers and error handling logic
* Used asynchronous fetch and Flask routes properly
* Configured CORS and structured APIs to respond correctly
* Used proper folder structure and Render-compatible

# 7. Key Learnings

* API integration and data handling using Flask
* Frontend-backend coordination using JavaScript and Python
* Working with real-time scientific APIs
* Designing for both scientific and casual users

# 8. SETUP AND USAGE INFORMATION

Follow the steps below to set up and run the SKY FUSION application on your local system.

**Requirements**

* Python 3.7 or higher
* Python Flask from library
* Git installed
* Internet connection (for live API calls)
* A code editor (like VS Code) is recommended

**Clone the Repository**

Open your terminal or command prompt and run:

```bash

git clone https://github.com/priyakumari-hub/skyfusion

**Run the Flask Application**

Start the application with: python app.py

Your terminal will show output like:

\* Running on http://127.0.0

# 8. Conclusion

SKY FUSION demonstrates how real-time data integration can simplify access to complex space and climate information. This project not only sharpened my technical skills but also showed the power of building accessible tools for awareness and action.

# 9. Links & Repository

* GitHub Repo: <https://github.com/priyakumari-hub/skyfusion>
* Live Demo : [**https://skyfusion.onrender.com**](https://skyfusion.onrender.com/)