**INFERA DATA SCIENCE INTERNSHIP**

**Daily Progress Report**

**Date:** 18th June 2025  
**Intern Name:** Aniruddh Vijayvargia  
**Project:** Climate Data Analysis - Refined Single Dataset Focus  
**Report Day:** Day 6

**1. Executive Summary**

Day 6 was dedicated to integrating the trained climate model into a fully functional and interactive **Streamlit dashboard**. The goal was to operationalize the machine learning model into a user-facing interface, enabling real-time max temperature predictions based on meteorological inputs. Significant debugging efforts were undertaken to resolve input feature mismatches and ensure consistent prediction accuracy. The resulting dashboard provides a clean, intuitive interface for stakeholders to interact with climate data and understand temperature dynamics.

**2. Tasks Completed Today**

* Streamlit App Setup: Initialized app.py with clean UI layout and page configuration
* Model Integration: Loaded trained Random Forest model and deployed using joblib
* Feature Alignment: Resolved input feature mismatches using model\_features.pkl to match training schema
* Input Interface Design: Added all relevant weather parameters (humidity, pressure, sunshine, wind, etc.) with interactive widgets
* Prediction Logic: Enabled real-time temperature prediction on button click with user feedback
* Dashboard Finalization: Ensured app renders properly, error-free, and is deployment-ready

**3. Technical Work Details**

* Built app.py with:
  + Clean UI layout using st.form, columns, and sliders
  + Real-time prediction with dynamic user inputs
  + Display of model outputs with confidence formatting
* Loaded best\_model.pkl and model\_features.pkl to ensure consistent feature schema
* Addressed and resolved:
  + NameError from JSON remnants in corrupted app files
  + ValueError from mismatched input features
* Refactored model input pipeline to support one-hot encoded features and default values
* Successfully tested the complete dashboard pipeline locally using streamlit run app.py

**4. Key Learnings & Insights**

**Technical Insights**

* Model prediction pipelines must retain exact column structures at inference time
* Joblib-based serialization is effective for both models and associated metadata (feature list)
* Streamlit provides rapid interface prototyping but requires careful layout handling to ensure user clarity

**Deployment Readiness**

* Dashboard is fully operational for local usage
* Model is accurate, stable, and fast — suitable for near real-time forecasting
* Framework is modular enough to later include additional outputs like min temperature or composite indices

**5. Challenges Encountered**

* Feature Mismatch Errors: Needed to re-align prediction-time inputs with training-time features using saved schema
* Notebook JSON Corruption: Initial app.py file was accidentally saved as notebook JSON, causing execution failures
* User Input Completeness: Missing predictors had to be approximated with fallback defaults or sliders to retain performance

**7. Screenshots & Evidence**

