Climate Visibility

PW Skills

Project Overview

- The objective of this project is to develop a machine learning model that can accurately predict the maximum visibility distance in a given location and weather condition.
- The model should take into account various weather parameters such as humidity, temperature, wind speed, and atmospheric pressure, as well as geographical features such as elevation, terrain, and land cover.
- The model should be trained on a large dataset of historical weather and visibility data, and validated using a separate test dataset.
- The ultimate goal is to provide a tool that can help improve safety and efficiency in various applications such as aviation, transportation, and outdoor activities.

Solution Proposed

Tech Stack Used

1. Python
2. Flask
3. Machine learning algorithms
4. Docker
5. MongoDB

Infrastructure required

1. AWS S3
2. Azure
3. Github Actions

GitHub Link: https://github.com/PWskills-DataScienceTeam/Climate-Visibility

Dataset used

https://github.com/PWskills-DataScienceTeam/Climate-Visibility/blob/main/notebooks/data.csv

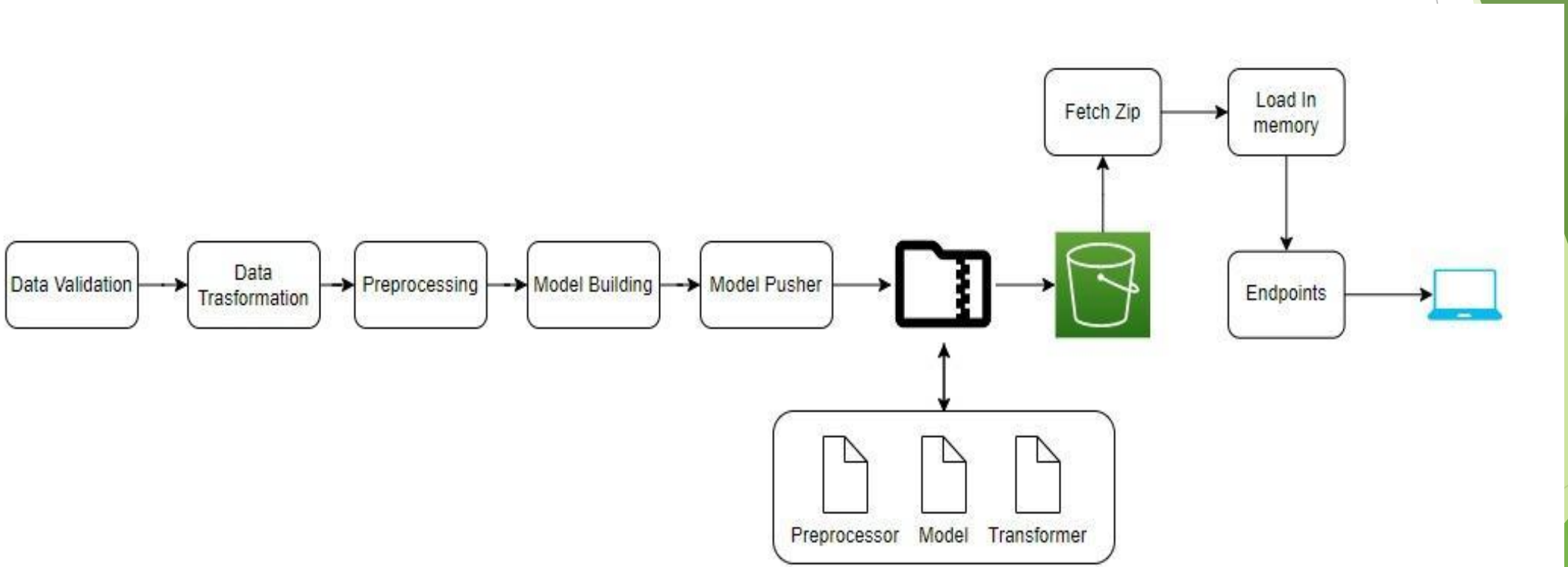
Flow of the Project Demo

Introduction
Environment Setup
Dataset Exploration
Data Preprocessing
Feature Engineering
Model Training & Evaluation
End to End Pipeline Run – Connecting the components of source folder (src)
Flask/FastAPI Deployment
Docker
Cloud Deployment
MLOps
Version Control Git GithHub
Final Wrap-Up

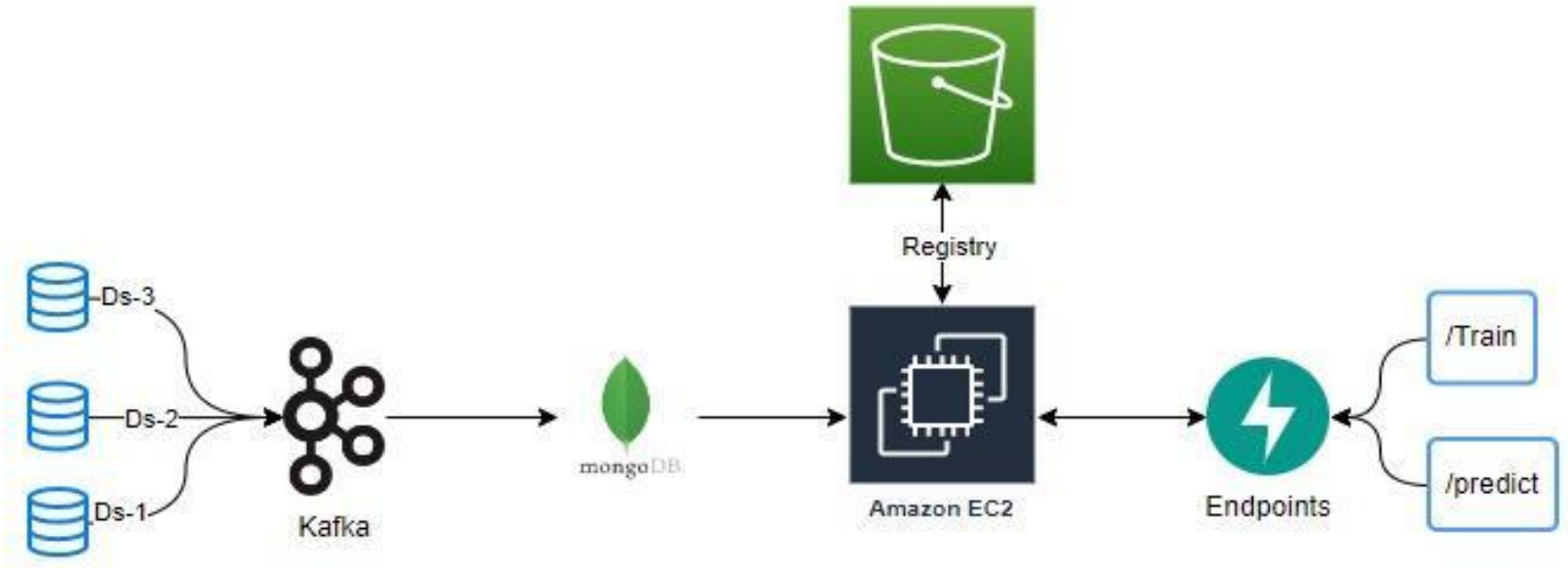
Libraries for the Project

Library	Description
boto3	AWS SDK for Python to interact with AWS services (S3, EC2, etc.).
botocore-stubs	Type hints and auto-completion support for botocore, the core of boto3.
dill	Enhanced serialization tool for Python objects, supporting more data types.
dnspython	DNS toolkit for performing DNS queries and operations.
evidently	Library for monitoring machine learning models and detecting data drift.
fastapi	Modern, high-performance web framework for APIs, based on Python type hints.
from-root	Utility to find the root directory of a project dynamically.
httptools	High-performance HTTP parser used by async web servers like uvicorn.
imblearn (imbalanced-learn)	Library for handling imbalanced datasets (e.g., SMOTE for oversampling).
mypy-boto3-s3	Type hints for boto3 S3 client, improving IDE autocompletion.
pip-chill	Simplifies viewing and freezing installed Python packages.
pymongo	MongoDB driver for Python, enabling interaction with MongoDB databases.
jinja2	Fast and powerful template engine used in Flask and FastAPI.
python-dotenv	Loads environment variables from a .env file for configuration.
types-s3transfer	Type hints for s3transfer, an Amazon S3 transfer utility in boto3.
uvicorn	ASGI web server for running FastAPI and other async frameworks.
watchfiles	Monitors file changes in real time, useful for auto-reloading services.
websockets	Enables real-time, two-way communication over WebSockets in Python.
wincertstore	Windows-only package for handling SSL/TLS certificates.
xgboost	High-performance machine learning library optimized for gradient boosting.
python-multipart	Parses multipart/form-data requests, often used for file uploads in APIs.
neuro_mf	A neural network-based meta-feature extractor for ML models.
kneed	Given a set of x and y values, **kneed will return the knee point of the function.** The knee point is the point of maximum curvature.

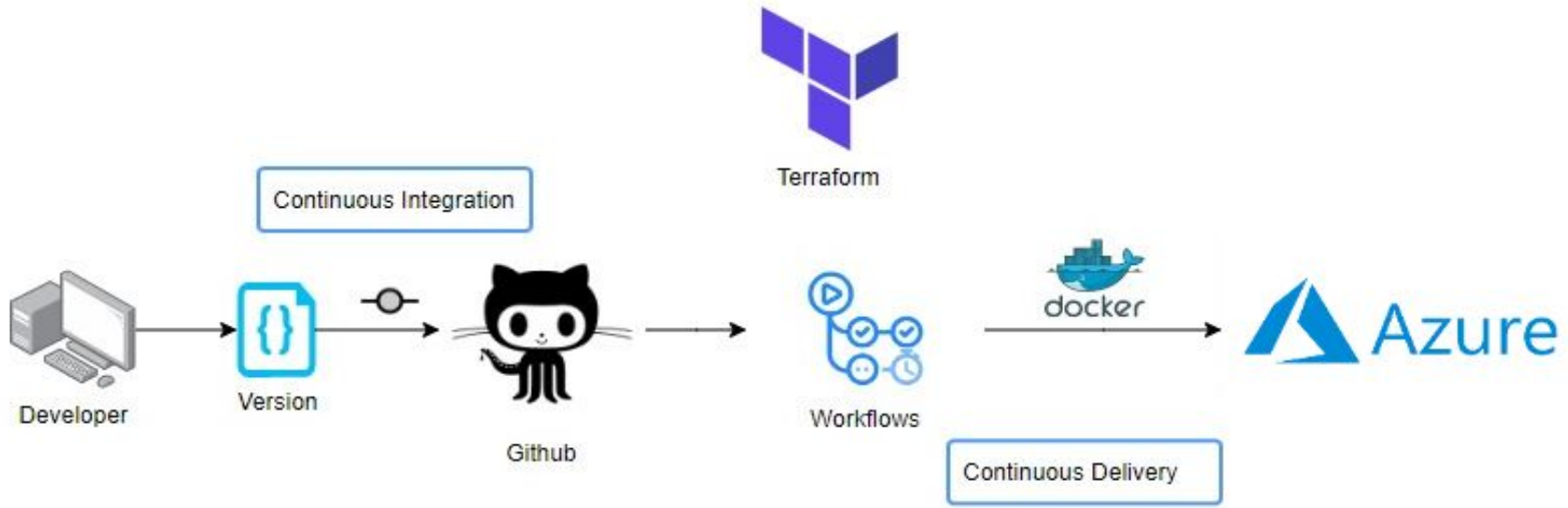
Project Architecture



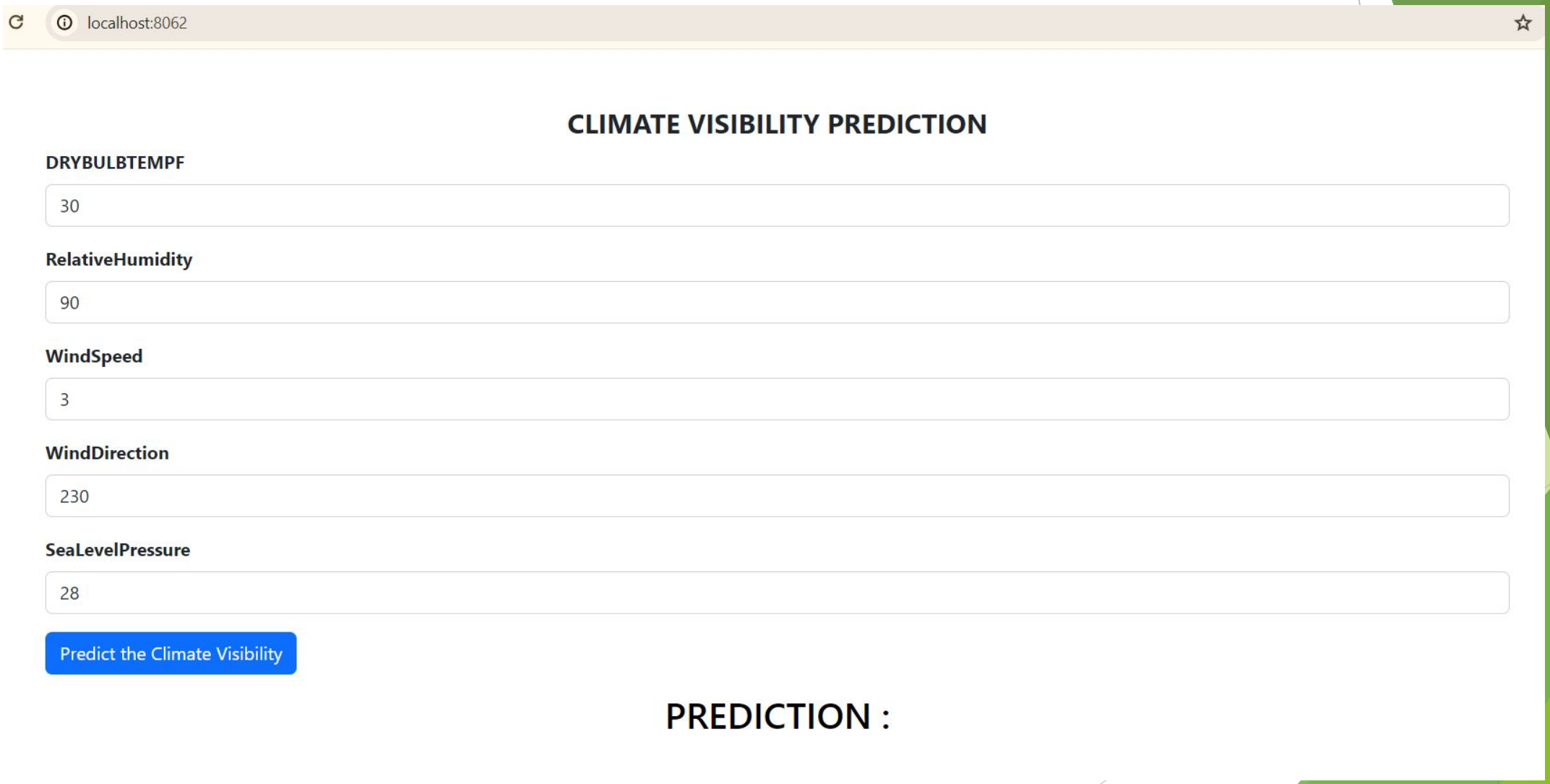
Data Collection Architecture



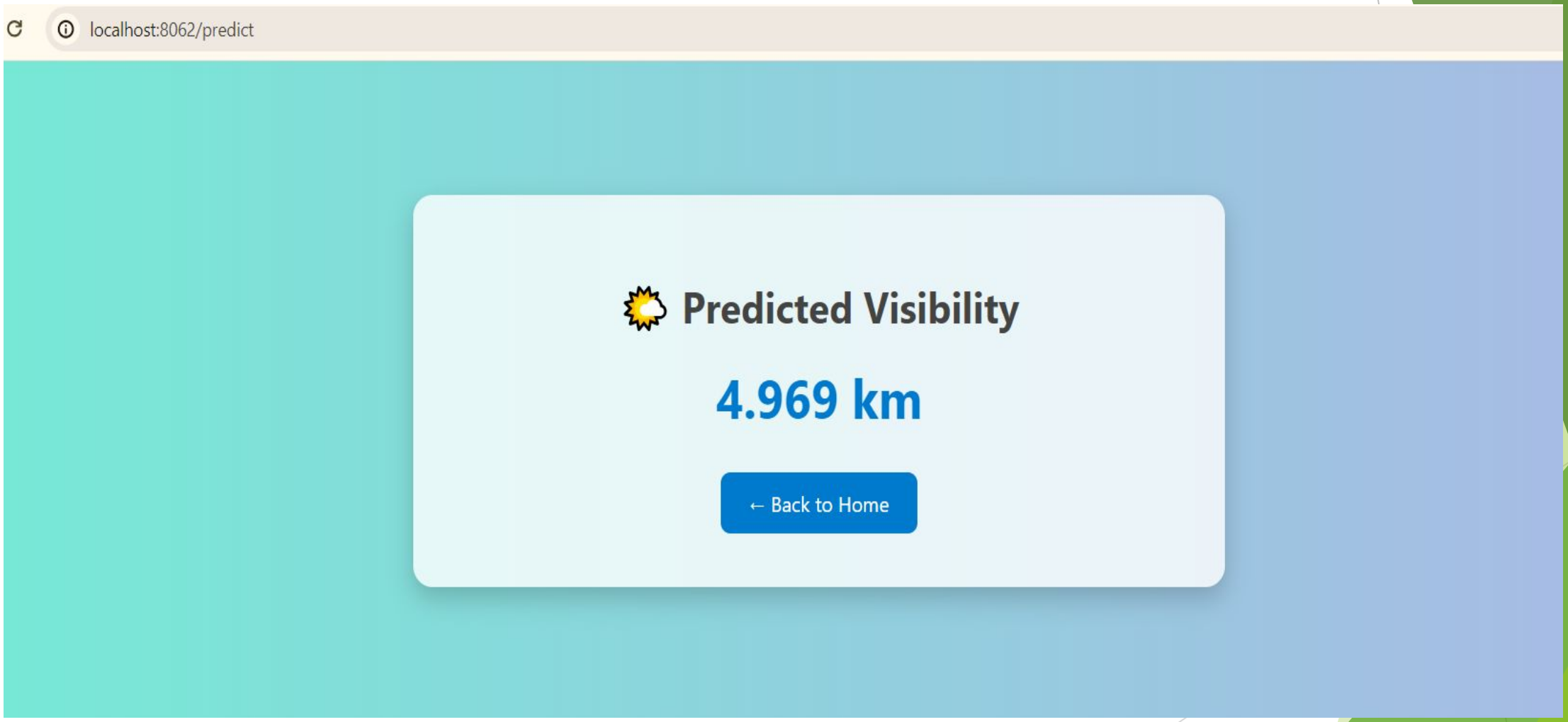
Deployment Architecture



Final Project Demo



Final Project Demo Output



Conclusion

✅ Conclusion: Visibility Distance Prediction from Climatic Conditions

This project focuses on predicting **visibility distance** based on various **climatic parameters** such as temperature, humidity, wind speed, pressure, and more. Visibility forecasting is critical for safe transportation, especially in aviation, maritime, and road networks. By leveraging machine learning models, the project aims to estimate how far one can see under given weather conditions, helping in proactive decision-making for traffic safety and logistics planning.

🎯 Educational Scope & Learning Outcomes

1. **Real-World Problem Solving**:
    - Addresses visibility forecasting — a practical and safety-critical task.
    - Helps understand meteorological variables and their impact on visibility.
2. **Hands-On ML Pipeline Development**:
    - Teaches dataset preprocessing, outlier handling, and normalization.
    - Includes model selection (e.g., Linear Regression, Random Forest, XGBoost) and evaluation (RMSE, MAE, R^2).
3. **Model Deployment**:
    - Builds an end-to-end solution: from data ingestion to model serving.
    - Includes a web-based prediction interface using FastAPI.
4. **Feature Engineering**:
    - Explores derived features like dew point, fog likelihood, and air density.
    - Introduces domain-specific feature interpretation.

Wrap Up!

🌍 Real-World Applications & Future Scope

- **Transportation Safety**: Real-time prediction can help reduce accidents in foggy, rainy, or dusty conditions.
- **Aviation & Marine Navigation**: Supports flight takeoff/landing protocols and ship routing decisions.
- **Smart Cities**: Can be integrated into intelligent traffic systems to manage flow and alerts.
- **Environmental Monitoring**: Useful for assessing visibility trends with climate change.

🔚 Final Summary

This project empowers learners to apply machine learning to a meaningful and high-impact problem in environmental and transportation domains. It combines data science, meteorology, and software development to create a fully functional predictive system. The hands-on experience gained from this project builds confidence in tackling real-world challenges using AI.