Weather Forecasting API Project

# 1. Introduction

This project is a Weather Forecasting API built using FastAPI. It integrates external weather data fetching services, JSON-based storage, SQL databases (via SQLAlchemy), and MongoDB for handling weather forecasts. The system also includes benchmarking to compare SQL and MongoDB query speeds.

# 2. System Architecture

The project uses the following components:  
- FastAPI: For building REST APIs.  
- SQLAlchemy: ORM for interacting with SQL databases.  
- MongoDB: NoSQL database for fast storage and retrieval.  
- JSON Reports: Simple file-based persistence.  
- External Weather API: Data source for weather forecasts.

# 3. Features

The API provides multiple endpoints for fetching, saving, deleting, and benchmarking weather forecasts:  
  
1. Root Endpoint: Health check for the API.  
2. Get Weather: Fetch weather for a single city.  
3. Get Multiple Weather: Fetch weather for multiple cities at once.  
4. Save Weather (JSON): Save weather forecast into a JSON file.  
5. Get Reports: Retrieve all saved reports from JSON file.  
6. Delete Report: Delete a specific city's report from JSON file.  
7. Save Weather (DB): Save weather forecasts into SQL + MongoDB databases.  
8. Benchmark: Compare SQL and MongoDB query performance for a given city.

# 4. Data Flow

1. The API fetches weather data using an external service.  
2. The data is structured into forecasts with time, temperature, and description.  
3. Data is then stored either in a JSON file (simple reports) or databases (SQL + MongoDB).  
4. SQL is useful for structured relational queries, while MongoDB is optimized for fast document-based retrieval.  
5. The benchmark endpoint allows comparing query times and record counts between SQL and MongoDB.