

WeatherJag

Recording real-time weather changes using MongoDB and OpenWeather API.

Shirish Nair

Computer and Information Science
Indiana University- Purdue University,
Indianapolis
Indianapolis **Indiana** United States
shinair@iu.edu

Maitreyi Muthya

Computer and Information Science
Indiana University- Purdue University,
Indianapolis
Indianapolis **Indiana** United States
mmuthya@iu.edu

ABSTRACT

WeatherJag - Real-time weather changes is a python application that uses OpenWeatherMap's REST services to fetch weather data and store it in MongoDB for analysis. This multithreaded application generates warnings during extreme weather conditions like snow or heat and shows the current weather data, a geographical map of the location, and weather statistics.

The application makes REST API calls to the OpenWeatherMap's API every time a thread is run and creates a collection in MongoDB. Once a collection is created in MongoDB, this data is then used to trigger weather alerts, fetch map coordinates and generate a line plot.

KEYWORDS

OpenWeatherMapAPI, Python, MongoDB, REST API, Multithreading, NoSQL, Pymongo, Real-time weather data, JSON.

1 Introduction

Weather forecasting is predicting weather conditions using scientific methods and tools, often done by the meteorological office, and uses different measures like barometric pressure, temperature, cloud cover, etc to determine the weather of a particular region or how it will change over a period of time. The techniques used for weather forecasting include using Persistence, Using a barometer, Observing the sky, Nowcasting, and Analog technique. Models have also been developed which use data from weather satellites, weather drones, and weather radars to accurately predict the weather forecast.

Weather forecasting has many applications and is used by government agencies, and the military, mainly by the navy and the airforce, and also by the general public. While the weather data consumed by agencies other than the general public has very specific applications, forecasting general weather data has become important to protect people from extreme weather conditions like thunderstorms, heat or cold waves, snow, cyclones, and other adverse weather conditions. All these weather conditions can disrupt the economy by making roads inaccessible in the case of snow or rain, cutting off electricity in the case of thunderstorms or cyclones, destroying vegetation, and causing a loss of human and animal life. Severe weather advisories in this can help in

preventing loss of life, both human and animal equally, and help in mobilising masses to a safer location. Weather forecasts are commonly published in newspapers and covered in news programs. Technological advancements and the introduction of the internet have made it possible for people to access weather information through web browsers and mobile phones. Many mobile phones these days have weather applications pre installed that show weather information.

WeatherJag is one such application written in Python which shows weather information. The targeted users of this application are students at IUPUI. Users can enter the name of the city whose weather information or alerts they wish to see. The application is also capable of generating a map of the city showing the current weather and a statistical graph showing temperature variation on various days. OpenWeatherMap API and MongoDB are the main components of this application. Figure 1 shows a high level overview of the working of the application.

1.1 MongoDB

WeatherJag uses MongoDB community edition, a NoSQL open source database. OpenWeatherMap APIs return JSON responses making MongoDB a suitable database to store JSON values directly. This reduces the overhead of parsing the JSON response string to fetch key-value pairs and then storing them in the database. MongoDB Compass allows effective data visualization and has different ways of representing data in the database.

1.2 OpenWeatherMap

OpenWeatherMap is an online REST service that provides global weather data via APIs. It provides a minute-by-minute forecast for any location. The REST APIs use an API key for authentication which is unique for every user. These APIs can fetch weather data as well as maps for any geographical location.

This application uses a 5 days/3 hour forecast API endpoint to fetch weather data for any geographic coordinates. All weather data can be obtained in JSON or XML formats.

1.3 Libraries

WeatherJag is written in Python 3.10.7 and uses different libraries to make API calls and to connect to MongoDB. Some of the important libraries used by the application include:

1. Pymongo: MongoDB driver for Python applications.
2. Httplib2: Http library to handle API requests..
3. Tkinter: Tools for the application interface.
4. Folium: Create maps in a separate HTML file.
5. Matplotlib and Seaborn: Plot lineplot.

2 Related Work

The Weather Channel is a popular weather forecasting channel that provides weather forecast and analysis. It also provides detailed weather data in the form of a web application and popular phone widgets. These widgets essentially make API calls and fetch the latest weather information and have been a motivation for developing this application [2].

3 Methods/Solution

OpenWeatherMap API is the main component of this application along with MongoDB. The OpenWeatherMap API services need an API key, which is used for authentication of the API calls. WeatherJag is a multithreaded application. Each thread performs a specific function. As shown in Figure 1, clicking on each button executes a thread which then makes API calls to the OpenWeatherMap API service or queries the database. If in case the database is empty, making an API call to fetch the weather details would create a new collection inside the database.

There are two main API endpoints that fetch the data in this application:

1. `https://api.openweathermap.org/data/2.5/weather?lat={lat}&lon={lon}&appid={API key}`
2. `api.openweathermap.org/data/2.5/forecast?lat={lat}&lon={lon}&appid={API key}`

The first API endpoint fetches the current weather information for any given city name [3]. The JSON response contains a number of key-value pairs out of which temperature, pressure, humidity, sunrise, sunset, cloud and fog information is displayed to the user.

The second API endpoint fetches a 5 day-3 hour forecast and is stored in MongoDB [4]. The response from this API endpoint creates a collection for every city that is searched for. As shown in Figure 7, every collection has 40 documents, which can be calculated as: 5 days * 8 intervals of 3 hours (in 24 hours). These details are then queried and weather alerts are generated in python depending on the weather values in the database.

Figure 1 shows a simplified flowchart of how data is fetched and stored in the database.

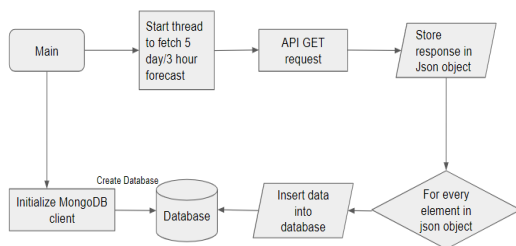


Figure 1: “Fetching and storing the data”

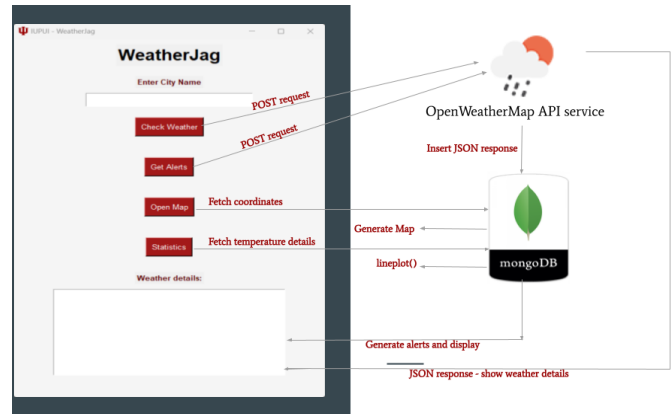


Figure 2: “WeatherJag user interface and working of the features”

4 Experimental Results/Evaluation

WeatherJag provides 4 features - Check weather, Get alerts, Open map and Get statistics in the form of a lineplot. A user just has to enter the name of any city. Clicking on Check Weather makes a POST request to the OpenWeatherMap API service. From the JSON response, weather details such as temperature, pressure, humidity, sunrise and sunset, clouds and fog for the entered city name are displayed in the textbox below the buttons as seen in Figure 3.

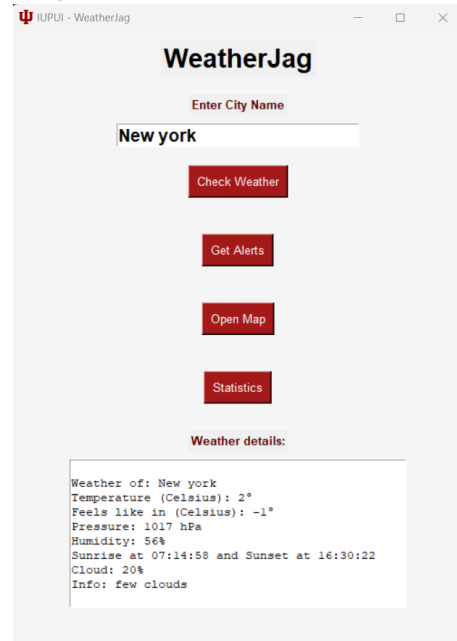


Figure 3: “Check Weather functionality of the application”

Get Alerts button fetches a forecast for the next 5 days and stores it in the database. Generated alerts show how the weather will be for the next 5 days with a 3 hour interval (Figure 4).

