**Weather Forecast**

This Weather App is a web-based application built using Flask for the backend and HTML, JavaScript, and Chart.js for the frontend. The application allows users to check weather forecasts by searching for a city, selecting a date, or using their current location. The app provides data visualizations in the form of tables and line charts showing temperature over time.

**Backend: Flask Application-**

* **Flask Routes and Functions**

1. **/weather API -** Receives the city name, fetches the forecast data for that city, and returns the data to the frontend.
2. **/location API -** Receives latitude and longitude, uses a geolocation API to get the city name, and returns it to populate the search field
3. **/data API –** Read the data.json file which inherit country, state and city name and provides all country name
4. **/state and /city API -** Retrieves a list of states and cities based on the selected country or state to populate dropdowns dynamically.
5. **/get\_Date API-** Takes a city and date as inputs and retrieves detailed forecast data at 3-hour intervals for the specified date.

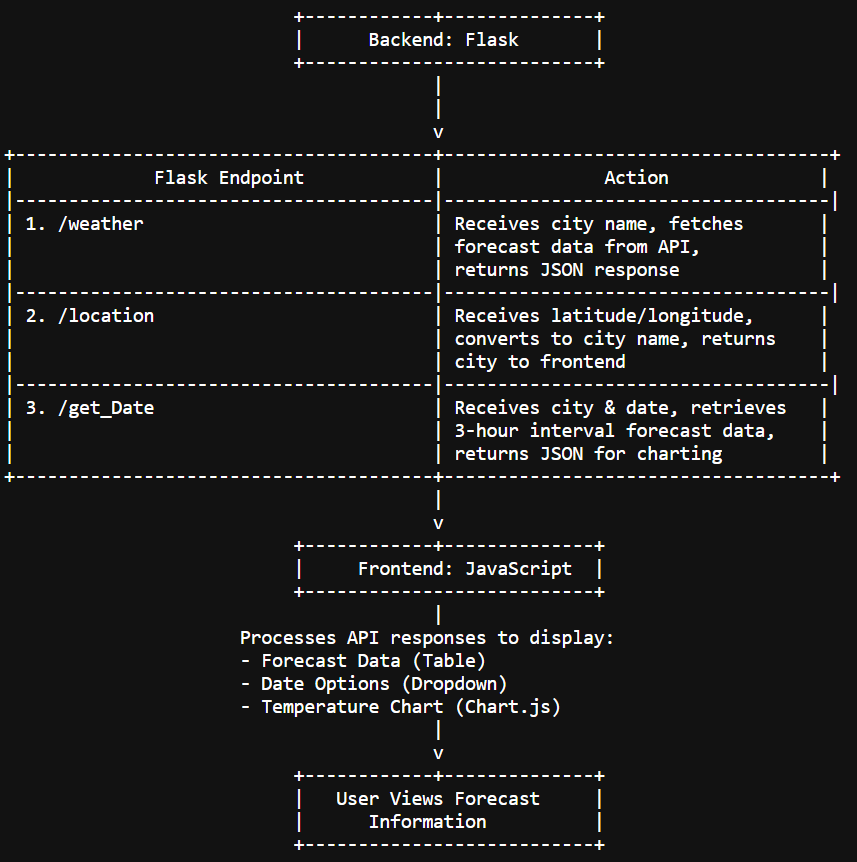
**Frontend: HTML, CSS, BootStrap and JavaScript-**

* **Header and Libraries-** Imports jQuery, Chart.js, Selectize.js, and Bootstrap for layout and functionality.
* **Main Sections-**
* **Header-** Contains the app title.
* **City Search Input**- Allows users to search for a city or use their current location.
* **Country-State-City Dropdowns -** Provides dropdowns to select a country, state, and city.
* **Forecast Table-** Display available forecast dates and allow users to select a date.
* **Weather Chart-** A line chart that displays temperature changes at 3-hour intervals.
* **JavaScript Functions-**
* **getLocation()** -Retrieves the user’s current geolocation and fetches the corresponding city name.
* **getWeather() –** provide 7 days forecasting table data.
* **getCountry(), getState() and getCity() –** provide country, state and city name
* **getDate()-** Populates the date dropdown with available forecast dates. Sends an initial request to the `/get\_Date` API with a default date and calls `weatherChart()` with the response.
* **weatherChart()-** Uses Chart.js to render a line chart with temperature data at 3-hour intervals. Labels the x-axis with time and the y-axis with temperature (°C).

**Public API-**

* **visualcrossing.com** -- provide 7 days forecast
* **openweathermap.org** -- provide 3hr. interval weather report
* **countrystatecity.in** -- provide whole country name, state name and city name
* **opencagedata.com** -- provide city name based on Lan and Lon (like gps)

****

****