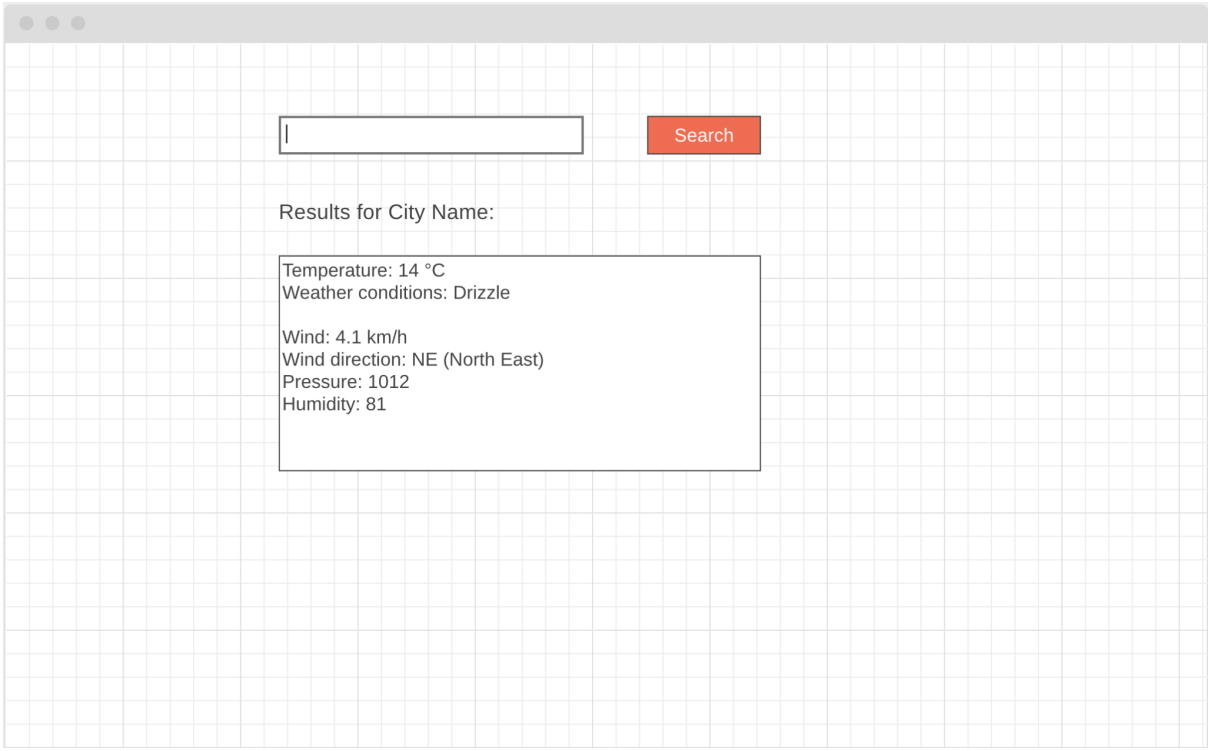
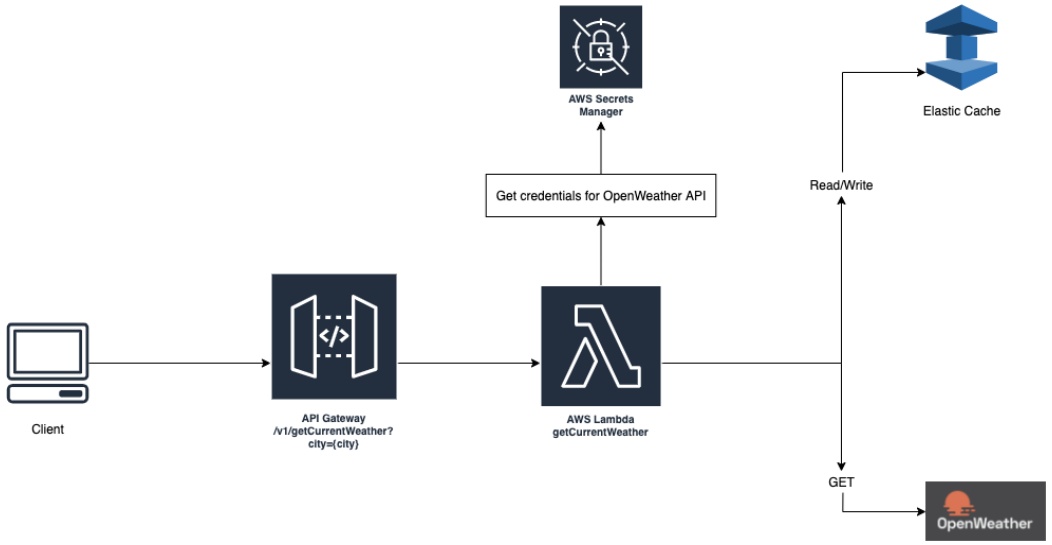
Technical Task

Integrate OpenWeather API (https://openweathermap.org/current) which will be consumed by following frontend:

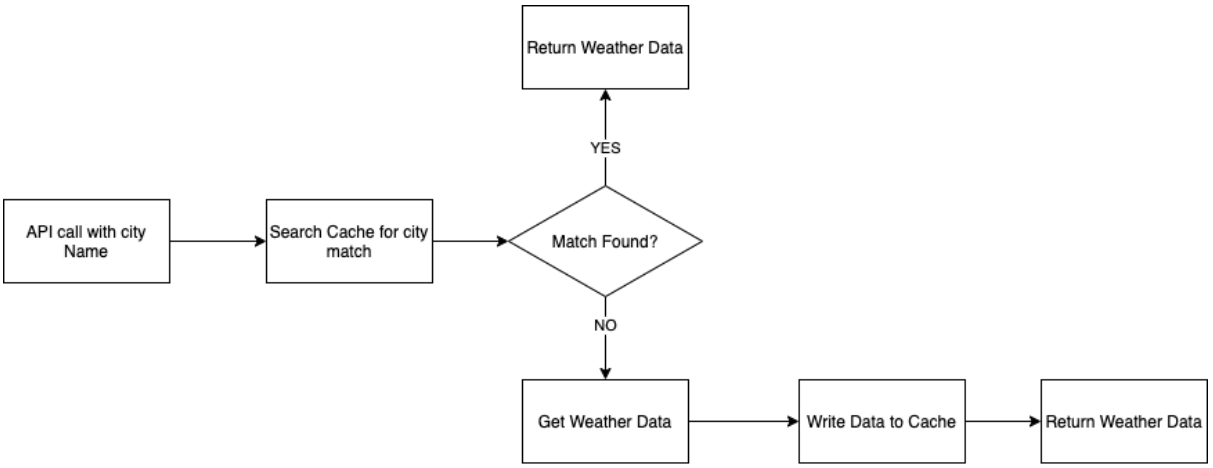


Which has text input for the city, button to trigger a request to API, and a text box below, to show information about current weather in the specified city.

AWS Diagram:



Application Logic Diagram:



Cache Policy time should be set to 60 seconds. Meaning that weather data should be cached only for a minute, and then data should be requested from the OpenWeather API.

API Mechanics

Request Parameters

There are only one endpoint. It accepts a city name as a string.

/v1/getCurrentWeather/?city={city}

Parameter	Data Type	Description	Required	Param Type	Notes
city	string	Name of the city	Y	Query string	Example: Oslo

Example Request

/v1/getCurrentWeather/?city=Oslo

Example Response

200 OK
Successful Response example


```
{
  "city": "Oslo",
  "temperature": 14,
  "weatherCondition": {
    "type": "Drizzle",
    "pressure": 1012
    "humidity": 81
  },
  "wind": {
    "speed": 4.1,
    "direction": "NE"
  }
}
```

Implementation Recommendations:

Frontend part of the app won't be evaluated, so that you don't have to spend any extra time on it. However, backend implementation along with configuration of AWS services will be closely inspected. Especially the AWS side of things will be the point of interest for the reviewer.

Bonus points will be awarded for additional thinking and good reasoning on designing and implementing the backend architecture and how backend will communicate with frontend (for example - validation of the data of incoming request, etc).

Super extra bonus points will be awarded for integrating AWS Cognito service (at least to "sign in" with user credentials), and adding Authorizer (example) to API Gateway for "/getCurrentWeather?city={city}" endpoint