CAB432 Assignment 2 Proposal

**Application Description**

The application used in this assignment will be based on the Restaurant Finder which is created for assignment 1. However, a few tweaks will be made to increase the load on request and processing such as increasing in range, display more results, etc.

**APIs Selections**

·         Zomato API

o   Get nearby restaurant information, pictures and locations with the input search keys

·         OpenWeather API

o   Using location data from Zomato, search for the weather in the area.

·         Google Map API

o   Retrieve the current location

o   Displays search results on the map

**Use Cases**

**Use Case 1: User want to search for a specific restaurant by name**

When search button is clicked, the typed searched key is used to hit Zomato API. The results will be displayed on Google Map as markers. When user selected a marker, another request will be sent to OpenWeather API using its location coordinates from Zomato API’s results. All finalized filtered results are shown in InfoWindow from Google Map API library. In each InfoWindow for each restaurant, there will be tags of the restaurants’ cuisines which users can click to search for the cuisines that they interested.

**Use Case 2: User want to look for nearby restaurant using their current location**

Upon launching the web application, the user is prompted to allow/decline their exact location to be exposed. If the user decline this but wish to use this feature later, they can use this feature by clicking a “using my current location” button in the search container. When user allow or choose to use their location to find nearby restaurants, their coordinates are then used to request Zomato API. Again, results are shown on the map as markers and; upon selecting marker, its location will be used to make a request to OpenWeather API for its current weather. All filtered results will be shown in the InfoWindow. Similarly, there will be cuisines tag which users can utilize if they suddenly want to search for restaurants of which cuisines, they are interested in.

**Technical Description**

**Flow chart**

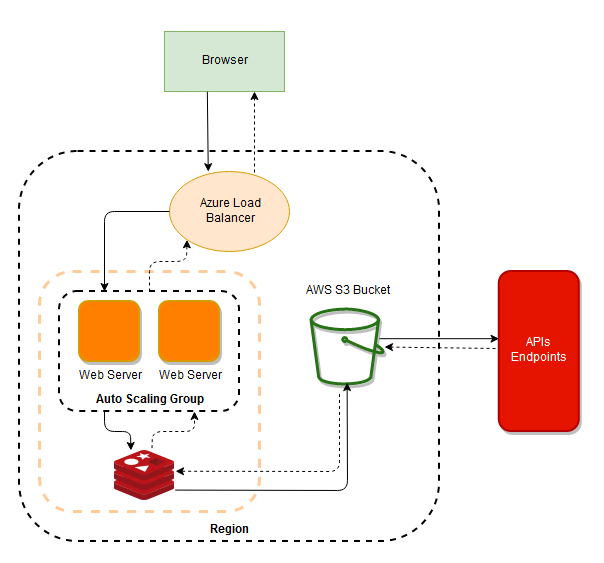


Figure 1 Flow chart of the application process

**Persistence**

*Redis Cache*

* To be used a local storage on VM instances to store the data of most recent search terms. If the current search key has previously search for, the data will be drawn from cache.

*AWS S3 Storage*

* Will be used as the second layer of data storage when the same search keys are entered multiple times. If it does not exist in the S3 storage, server will get the data from the APIs.

**Scaling**

*Azure Load Balancer*

* Aims to scale the instances by requests instead of scaling under heavy loads from the application process. Initially, values and settings will be similar to the assessable practical. However, this still subject to change if necessary.