

# MyLocator

## Background Reading

### **Google Maps API-**

We can integrate google maps in our android application. For integration, we require an API key from google. To get an API key, we need to register from a google account. Maps is a free to use google service but it has a limited number of requests under free tier. If we exceed the limit, we need to pay of extra requests. Another way is to register for a premium plan membership.

Although it is not a compulsion to have a google API but it does gives an added advantage if we use an API key such as we can use place ID instead of an address or latitude, longitude pair. An address always has risks, as the limit of characters is 8192 characters. Also, if we send an address as location, then it will be geocoded into latitudes and longitudes, which can be different from the ones a user could have send.

### **Google Maps Direction API-**

We also need a direction API along with a map API, so that we can get directions from our starting point to ending point, along with the possible routes. We can use a PolyLine to show the routes on google maps. To send an HttpRequest to the google server, we will use a link(which has a format of

<https://maps.googleapis.com/maps/api/directions/outputFormat?parameters>) with some of the required and some of the optional parameters. Required parameters include output parameters, which will be json in our case, origin, destination and key. The optional parameters that we require is travel mode(driving, walking, bicycling, transit), transit mode(bus, subway, train, tram, rail) and transit routing preference(less walking). Parameters and their values are separated by “=” between them and different pairs of values can be separated by pipe character( | ). To pass place id in origin, we need to pass “place\_id:ID” as place ID where “ID” is the ID of place. It is important not to have any blank spaces in the link and the length of the link must be less than 8192 characters.

### **Usage limit of free tier account**

- 2,500 free directions requests per day, calculated as the sum of client-side and server-side queries.

- Up to 23 waypoints allowed in each request, whether client-side or server-side queries.
- 50 requests per second, calculated as the sum of client-side and server-side queries.

Limit extension charges - \$0.50 USD / 1000 additional requests, up to 100,000 daily.

## **Google Street View**

Destination of Google Maps ends at the end of the road. After that, the users need to look for the location themselves, which can also be a bit difficult. For this problem, we can include a google street view, which will show the view of the location on which the user will reach. We can make it zoomable and movable. It will help our users for a better searching of locations. As per integration purpose, we can include a button which opens a street view either as a fragment or in a split screen, but fragment is preferable.

## **Amazon Web Services:**

Amazon web services is a software as a service(saas) based web application which lets you create your own databases and host it on the web on their servers. It gives a free cloud storage for a year followed by a paid subscription offer. Amazon Web Services offers many facilities. In that, we can use the EC2(Electronic computation version 2) configuration. For testing purposes, we can put the application on free micro tier, which can allow free connection and limited micro tier database. We can use parse server from Bitnami to use the database and to store data online for visited places. In a later model of our application, we can use schedules so that our application automatically detects the locations for everyday's travels and makes location schedules before head. We can store that data online and offline also.

## **Parse Server:**

It is an open source server based application. We will build database using parse server. We will integrate it in our EC2 cloud service from amazon web services. It consists of classes and instances of that classes, which can be added or removed for our functionalities.

**References:**

- <https://developers.google.com/maps/documentation/directions/intro>
- <https://developers.google.com/maps/documentation/javascript/directions>
- <https://developers.google.com/maps/documentation/directions/usage-limits>
- <https://developers.google.com/maps/documentation/javascript/examples/places-autocomplete-directions>
- <https://developers.google.com/places/place-id>
- <https://aws.amazon.com/documentation/ec2/>
- <https://docs.parseplatform.org>