Milestone3

1. we use Google Map API as a new service.
2. some links of tutorial/video
   i) https://www.youtube.com/watch?v=OLaQ_WdNkvQ&feature=youtu.be
   ii) https://github.com/googlemaps/google-maps-services-python
   iii) https://developers.google.com/maps/documentation/distance-matrix/start
3. some sample code: （example request）(example response)

   i) Distance Matrix

   The following example requests the distance matrix data between Washington, DC and New York City, NY, in JSON format:

```
https://maps.googleapis.com/maps/api/distancematrix/json?units=imperial&origins=Washington,DC&destinations=New+York+City,NY&key=YOUR_API_KEY
```

   Below is a sample response, in JSON:

```
{
   "destination_addresses" : [ "New York, NY, USA" ],
   "origin_addresses" : [ "Washington, DC, USA" ],
   "rows" : [
      {
         "elements" : [
            {
               "distance" : {
                  "text" : "225 mi",
                  "value" : 361715
               },
               "duration" : {
                  "text" : "3 hours 49 mins",
                  "value" : 13725
               },
               "status" : "OK"
            }
         ]
      }
   ],
   "status" : "OK"
}
```

   ii) Geocoding

   Geocoding is the process of converting addresses (like a street address) into geographic coordinates (like latitude and longitude), which you can use to place markers on a map, or position the map.

   The following example requests the latitude and longitude of "1600 Amphitheatre Parkway, Mountain View, CA", and specifies that the output must be in JSON format.

```
https://maps.googleapis.com/maps/api/geocode/json?address=1600+Amphitheatre+Parkway,+Mountain+View,+CA&key=YOUR_API_KEY
```

   Below is a sample Geocoding response, in JSON:

```
{
   "results" : [
      {
         "address_components" : [
            {
               "long_name" : "1600",
               "short_name" : "1600",
               "types" : [ "street_number" ]
            },
            {
               "long_name" : "Amphitheatre Parkway",
               "short_name" : "Amphitheatre Pkwy",
               "types" : [ "route" ]
            },
            {
               "long_name" : "Mountain View",
               "short_name" : "Mountain View",
               "types" : [ "locality", "political" ]
            },
            {
               "long_name" : "Santa Clara County",
               "short_name" : "Santa Clara County",
               "types" : [ "administrative_area_level_2", "political" ]
            },
            {
               "long_name" : "California",
               "short_name" : "CA",
               "types" : [ "administrative_area_level_1", "political" ]
            },
            {
               "long_name" : "United States",
               "short_name" : "US",
               "types" : [ "country", "political" ]
            },
            {
               "long_name" : "94043",
               "short_name" : "94043",
               "types" : [ "postal_code" ]
            }
         ],
         "formatted_address" : "1600 Amphitheatre Pkwy, Mountain View, CA 94043, US
         "geometry" : {
            "location" : {
               "lat" : 37.4267861,
               "lng" : -122.0806032
            },
            "location_type" : "ROOFTOP",
            "viewport" : {
               "northeast" : {
                  "lat" : 37.4281350802915,
                  "lng" : -122.0792542197085
               },
               "southwest" : {
                  "lat" : 37.4254371197085,
                  "lng" : -122.0819521802915
               }
            }
         },
         "place_id" : "ChIJtYuu0V25j4ARwu5e4wwRYgE",
         "plus_code" : {
            "compound_code" : "CWC8+R3 Mountain View, California, United States",
            "global_code" : "849VCWC8+R3"
         },
         "types" : [ "street_address" ]
      }
   ],
   "status" : "OK"
}
```

iii) Reverse Geocoding

Reverse Geocoding is the process of converting geographic coordinates into a human-readable address.

The following is a example requests. It specifies that the output must be in JSON format.

```
https://maps.googleapis.com/maps/api/geocode/json?latlng=40.714224,-73.961452&key=YOUR_API_KE
Y
```

Below is a sample Reverse Geocoding response, in JSON:


```
{
   "plus_code" : {
      "compound_code" : "P27Q+MC New York, NY, USA",
      "global_code" : "87G8P27Q+MC"
   },
   "results" : [
      {
         "address_components" : [
            {
               "long_name" : "279",
               "short_name" : "279",
               "types" : [ "street_number" ]
            },
            {
               "long_name" : "Bedford Avenue",
               "short_name" : "Bedford Ave",
               "types" : [ "route" ]
            },
            {
               "long_name" : "Williamsburg",
               "short_name" : "Williamsburg",
               "types" : [ "neighborhood", "political" ]
            },
            {
               "long_name" : "Brooklyn",
               "short_name" : "Brooklyn",
               "types" : [ "political", "sublocality", "sublocality_level_1" ]
            },
            {
               "long_name" : "Kings County",
               "short_name" : "Kings County",
               "types" : [ "administrative_area_level_2", "political" ]
            },
            {
               "long_name" : "New York",
               "short_name" : "NY",
               "types" : [ "administrative_area_level_1", "political" ]
            },
            {
               "long_name" : "United States",
               "short_name" : "US",
               "types" : [ "country", "political" ]
            },
            {
               "long_name" : "11211",
               "short_name" : "11211",
               "types" : [ "postal_code" ]
            }
         ],
         "formatted_address" : "279 Bedford Ave, Brooklyn, NY 11211, USA",
         "geometry" : {
            "location" : {
               "lat" : 40.7142484,
               "lng" : -73.9614103
            },
            "location_type" : "ROOFTOP",
            "viewport" : {
               "northeast" : {
                  "lat" : 40.71559738029149,
                  "lng" : -73.9600613197085
               },
               "southwest" : {
                  "lat" : 40.71289941970849,
                  "lng" : -73.96275928029151
               }
            }
         },
         "place_id" : "ChIJT2x8Q2BZwokRpBu2jUzX3dE",
         "plus_code" : {
            "compound_code" : "P27Q+MC Brooklyn, New York, United States",
            "global_code" : "87G8P27Q+MC"
         },
         "types" : [
            "bakery",
            "cafe",
            "establishment",
            "food",
            "point_of_interest",
            "store"
         ]
      },

   ... Additional results truncated in this example[] ...

   ],
   "status" : "OK"
}
```

4. In order to allow users to find the nearest hospital in the shortest time, we would like to achieve if users send their location, then this chatbot can get the distance to the nearest hospital and the shortest time to get there. First, chotbot get the user's location and find the nearest hospital, then get its location. Second, we will use Geocoding to convert addresses into geographic coordinates, then get the shortest distance between two location. Finally, the chatbot will return the result of the shortest distance and the shotest time. For further development, we hope we can achieve the chatbot return different results based on the vehicle used by results include the number of traffic lights on the route.