Location Simulator

You are given two locations - A & B. You are provided with latitude & longitude of them (short form LatLngs). You need to calculate 'real' points on the road that connects A & B. You need to use data from **google directions API .**

Sample Input: LatLngs for point A and point B.
A is 12.93175, 77.62872 and B is 12.92662, 77.63696

Sample Output: LatLngs at a constant **distance interval** of **50 m** between A & B on the road.
12.93175, 77.62872
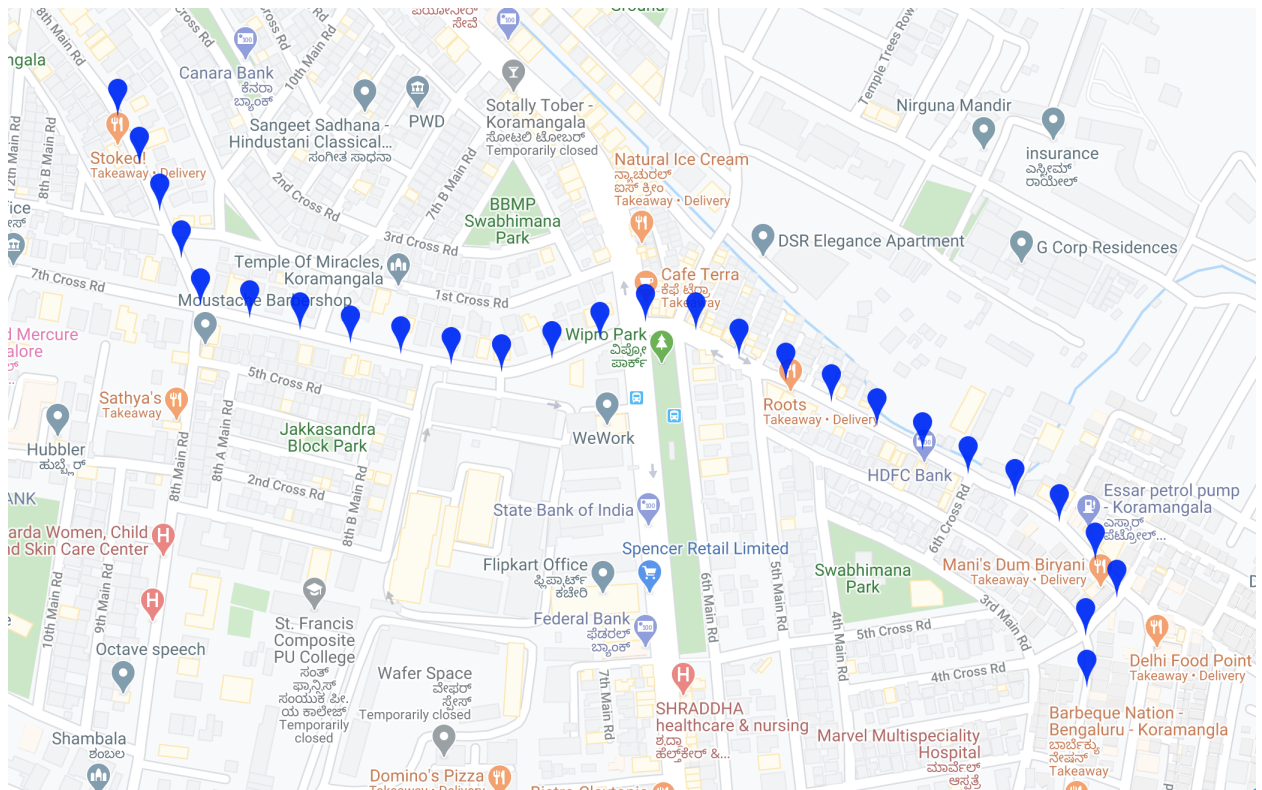12.93166, 77.62852
12.93125, 77.62870
.

.
12.92713, 77.63719
12.92668, 77.63717
12.92662, 77.63696

Sample plot:

Before you start to code

1. Research on Google Directions API - figure out the right response fields that can help you get the data. Figure out how you can use the data as the data might not be readily consumable.
    API Key: *AIzaSyAEQvKUVouPDENLkQlCF6AAap1Ze-6zMos*
2. Figure out how you are going to satisfy the criteria of points at fixed intervals. You need to write your own logic around this. Use hints to make sure that you are working in the right direction.

Now you can start to code

3. You are now ready to code. Fire up your IDE & start to write the code. We like to see clean/production code but we understand you are working on a new problem. If you want to get something up quick & dirty, make it working & then focus on the refactor, that is fine.
4. Test your output using one of the following tools and attach screenshots for the interviewers review.
    https://mobisoftinfotech.com/tools/plot-multiple-points-on-map/ (format: *19.070, 72.877,*)
    https://gpspointplotter.com/ (format: *43.697,-73.983*)

Evaluation Criteria:
This is the criteria for the evaluation of your solution in order of precedence:-
1. Accuracy of the solution.
2. Edges cases covered in the solution.
3. Code quality (should be production quality)

Once done, make sure that the code is runnable and email your zipped code along with screenshots and a READ ME to *careers@locus.freshteam.com*.

Hints:
1. Assume the distances handled by the location simulator are much smaller than the radius of the earth, hence coordinate geometry can be used for interpolation, as the area of interest can be considered flat.
2. What does a step represent in real life? Is the start and end of the step sufficient to plot an accurate path? How will it handle a curved road?
3. Feel free to use libraries available online and/or code available on stack overflow. In case of the latter, improve the code to follow best practices like you would do if you were copying the code to your own project.