EleNA : Elevation Based Navigation

Indentation&Semicolon

- Sai Vineeth Kumar Dara
- Rohith Siddhartha Reddy Bheemreddy
- Tushita Singh
- Rachana Ponagandla

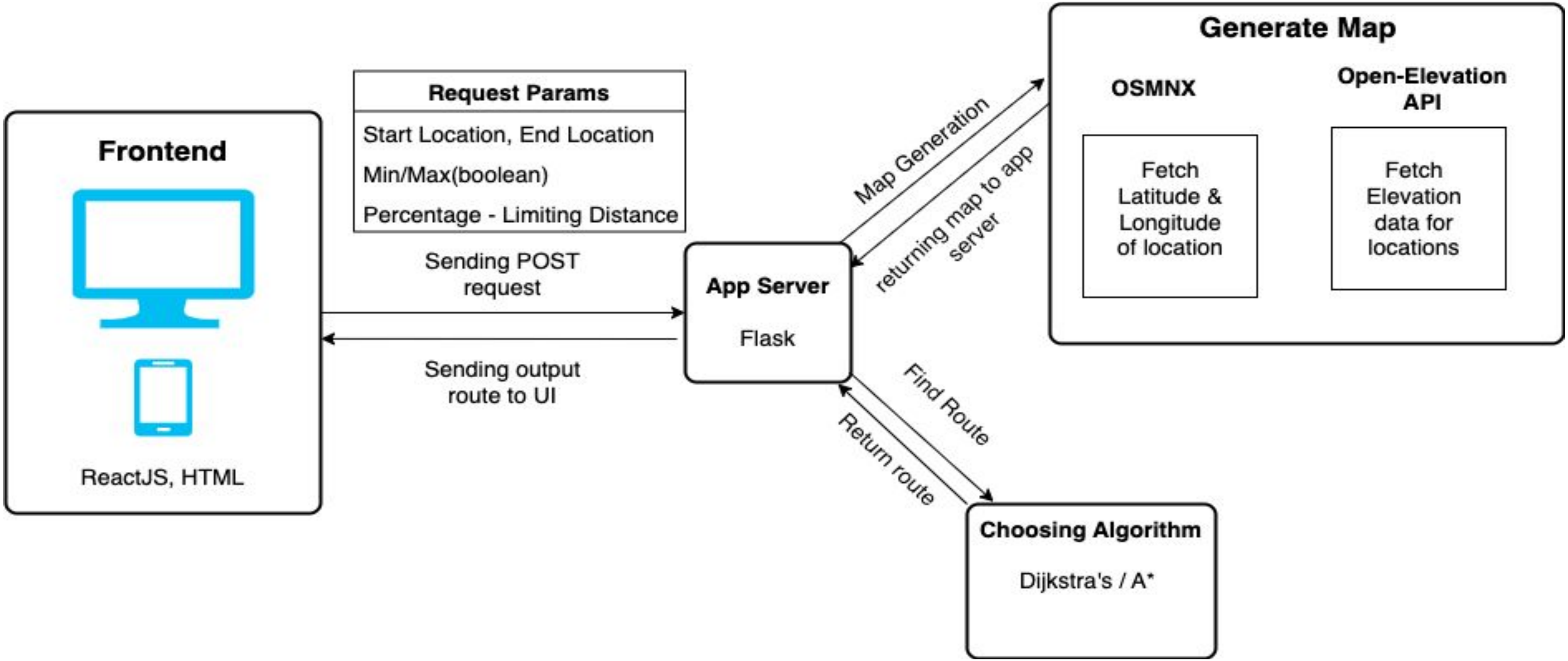
Github repo: https://github.com/rohithsiddhartha/Indentation-Semicolons

Problem Statement

Elevation-based Navigation (EleNa) is an application that, given a start and end location, computes a route that maximizes or minimizes the elevation gain and limits the total distance between the locations to x% of the shortest path.

Maximizing the elevation gain could be useful to joggers/bikers who may be looking for an intense workout. On the other hand, minimizing the elevation gain could be useful for those who don't prefer steep climb in the the route.

Architecture



Backend Algo

- App server receives the user’s POST request containing source, destination, min/max(boolean) and shortest path percentage.
- Generates the Map using OSMNX(finds latitude and Longitude of locations) and Open-Elevation API(provides elevation data of locations)
- App server receives the Map Information and passes it to a routing algorithm(Dijkstra’s or A*) to find the route
- App server receives the route and sends the response to UI

Tools:

- Flask API (Python)

Frontend GUI Components and Tools

UI contains following components

Input Components:

- Source and destination address
- The shortest path percentage
- The max or min elevation button selection

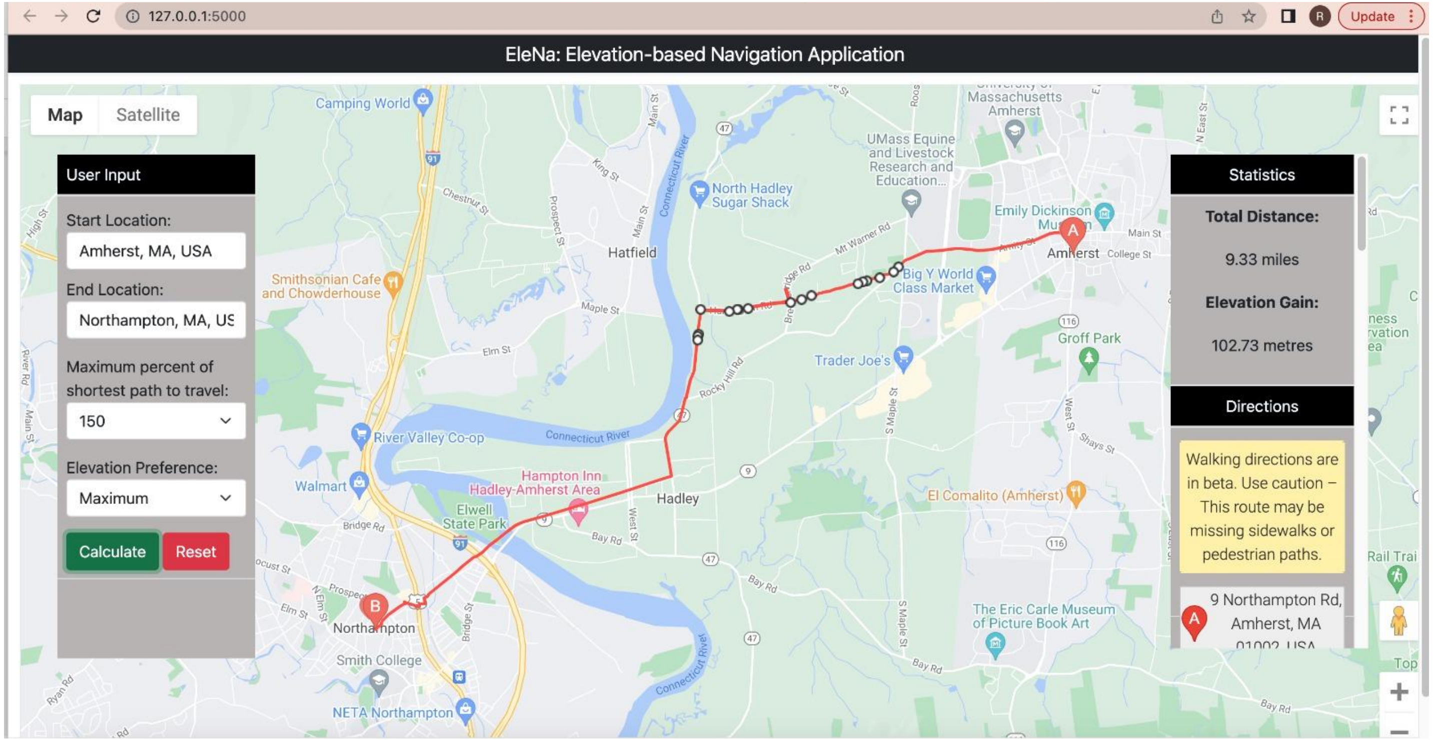
Output Components:

- The Map Containing Output Route
- Total distance
- Elevation gain

Tools:

- ReactJS, HTML, CSS, Bootstrap

UI Mock Design



127.0.0.1:5000

EleNa: Elevation-based Navigation Application

User Input

Start Location: Amherst, MA, USA

End Location: Northampton, MA, US

Maximum percent of shortest path to travel: 150

Elevation Preference: Maximum

Calculate Reset

Statistics

Total Distance:

9.33 miles

Elevation Gain:

102.73 metres

Directions

Walking directions are in beta. Use caution – This route may be missing sidewalks or pedestrian paths.

9 Northampton Rd, Amherst, MA 01002, USA

Backend Components and Tools

Backend contains following components

- Map Generation Component
    - Fetch Coordinates
    - Fetch Elevation Data
- Route Finding Component
    - Dijkstra’s Algorithm

Tools:

- Flask, OSMNX, Open Elevation API

Evaluation/ Testing

These are the following things we considered while manual testing and automated testing

- Location Validation
- Path Elevation output
- Path Length output
- Path Validation check
- Location Coordinates
- Dijkstra Algorithm

We have written unit test cases to check the correctness of the class methods. Multiple testing scenarios are shown in the UI testing report file shared.

Best software practices used

These are some of the best software practices we used in our project:

1. Implemented MVC architecture.
2. Avoided magic strings by declaring constant strings.
3. Understandability: Documentation - we added comments to classes and methods to make the code more understandable. Also we added ReadMe and requirements file for easy code setup.
4. Testability: Implemented unit test cases to test the tool. Also we conducted thorough manual testing.
5. Appropriate naming conventions for better understanding of the code.
6. While pushing the changes we also did manual review of the changes by confirming the correctness of changes from team mates.
7. Conducted weekly meetings to keep everyone in the team updated regarding the project status.

Why choose our Elena system

Here are few interesting things we implemented thats sets us apart from other Elena systems:

1. Instead of just showing the path, we are showing directions from the source to destination too.
2. Our tool works for every location in USA and not just limited to a specific area (Eg.Amherst)
3. Cache implementation - to make the tool more user friendly.
4. Our project supports MVC architecture.
5. System is tested using unit test cases and thorough manual testing.

Thank you

Github repo: https://github.com/rohithsiddhartha/Indentation-Semicolons