

Elevation-Based Navigation App (EleNA)

Requirements Specification

1. Introduction

The Elevation-Based Navigation App, or EleNA, is an altitude-aware program which can take elevation gain or loss into account when planning routes between two points. Unlike traditional mapping applications, EleNA allows users to specify a threshold distance that will search for the maximal or minimal elevation gain or loss.

2. System requirements and functional requirements

Product Perspective

EleNA will be a mapping web application that will have a focus on desired elevation gain or loss when considering a path between two points. It will have a Google Maps-style interface, with a list of preset modes to choose from (minimum elevation gain, maximum elevation gain, shortest direct path), and a threshold slider that will specify how willing you are to endure additional distance to meet your selected mode. When all the fields are specified and the user clicks on the “Route” button, the app will use a pathfinding algorithm to find an optimal route according to the constraints specified by the user.

Operating Environment

The operating environment will be a containerized webapp, deployable to any operating system. The webapp itself will be accessible via the internet, from any browser. The webapp will have a React frontend, and a FastAPI backend, with a MongoDB database.

Design Constraints

EleNA will be an online-only webapp. It will require a valid map API key, as the mapping API will be crucial to its operation.

3. External interface requirements

- Our application needs to have some sort of a map API from which we could derive the required dataset like nodes, elevation, and the likes from one point to another. Proposed map APIs could be OpenStreetMap, google maps API etc.
- We shall communicate with the different components using the HTTPS protocol.

4. Non-functional requirements

Usability

We are going to use a simple or familiar interface so that the user can automatically adapt to the UI. We want to have a minimalistic UI without much clutter. We plan to have a distance metric, threshold, route from-to, and use at most two UI to ask for input and output.

Testability

Mapping is a critical application that can have serious consequences if the route provided is incorrect. Therefore, we must ensure that the routing algorithm and interface is thoroughly testable.

Scalability

For scalability, EleNA will require a valid login, in order to avoid hitting the rate limit on our mapping API. It will also store a per-user navigation history so users can re-use a precomputed route from earlier, saving them time and computation time on the server.

Security

EleNA will use OAuth 2.0 for secure authentication with encryption, with encrypted session tokens to ensure that the user session is valid. For webapp security and safety, all user inputs will be sanitized to avoid SQL injection attacks or invalid input types.

Privacy

We plan to protect the privacy of our users and their search history by encrypting their login password using OAuth 2.0 encryption. The only identifying user data that will be stored is email, and that is for login purposes only. User data will not be sold or shared in any way to external services, and the mapping API will not know which specific users requested what routes, only that the webapp itself requested it.