Final Project

EleNa: Elevation-based navigation

Sakshi Bhalerao

Snehal Thakur

Vidhi Mody

Vrushti Mody

Problem Statement

Elevation-based navigation system (EleNA) Navigation systems optimize for the shortest or fastest route. However, they do not consider elevation gain. Let's say you are hiking or biking from one location to another. You may want to literally go the extra mile if that saves you a couple thousand feet in elevation gain. Likewise, you may want to maximize elevation gain if you are looking for an intense yet time-constrained workout. The high-level goal of this project is to develop a software system that determines, given a start and an end location, a route that maximizes or minimizes elevation gain, while limiting the total distance between the two locations to x% of the shortest path.

1. Introduction

1. Purpose

   The objective of our project is to develop a system that takes into account the elevation gain while moving from one location to another. It takes the following input:

   1. Start location (Dynamic Dropdown)
   2. End location (Dynamic Dropdown)
   3. Maximizes or minimizes elevation gain (Radio Button)
   4. % increase from shortest path (Circular Spinner)

   Based on the input, it outputs the route from source to destination along with elevation gain.

2. Scope

   In traditional map routing algorithms we get the shortest/ fastest route from source to destination. But they do not take into account the elevation gain. Let's say you are hiking or biking from one location to another. You may want to literally go the extra mile if that saves you a couple thousand feet in elevation gain. Likewise, you may want to maximize elevation gain if you are looking for an intense yet time-constrained workout. We have designed our system keeping this use case in mind.

2. Overall Description

1. Project Functions

   The user can do the following:

   1) Input the source
   2) Input the destination
   3) Input the elevation type (min/max)
   4) Percentage increase from shortest distance
   5) Search for the route from source to destination

   The source and destination is checked to ensure a valid address has been entered before the user can search for the route. Based on the input provide we output the optimal route from source to destination using **A*and Dijkstra's** algorithm.

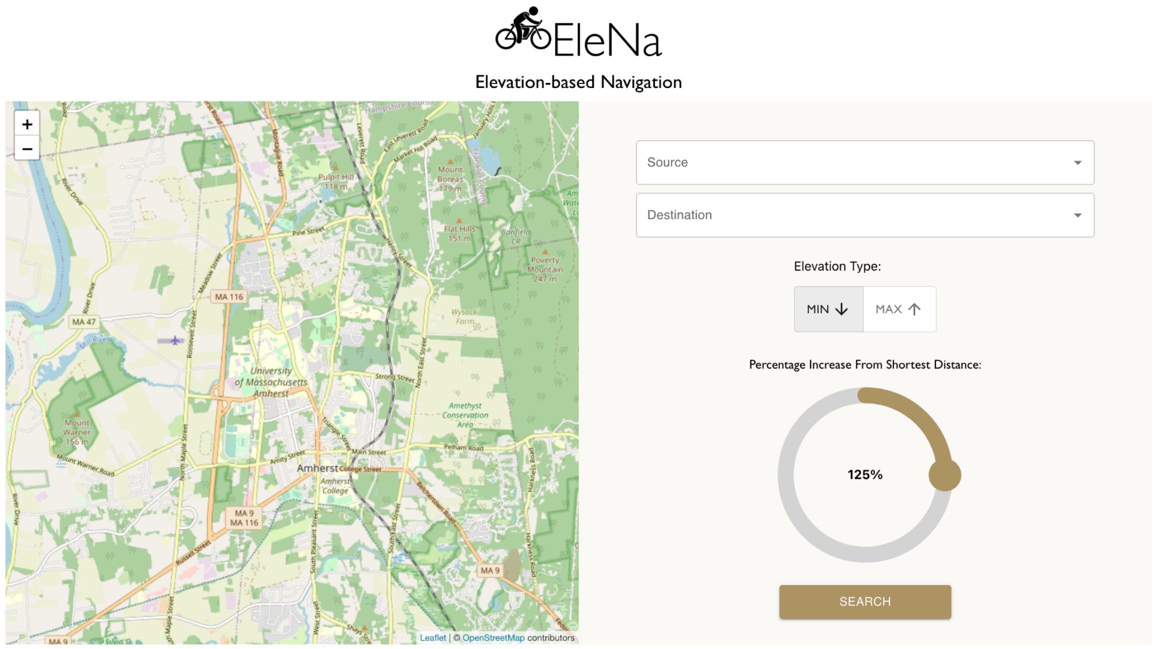
3. Interface Requirement

1. User Interface
   a. Frontend
      i. React.js
      ii. HTML
      iii. CSS
      iv. JavaScript
   b. Backend
      i. Flask server
      ii. Python

2. Hardware Interface
   a. VSCode
   b. 2.0 GHz Processor

4. Functional Requirements

1. Data model that represents the geodata.



EleNa

Elevation-based Navigation

Source

Destination

Elevation Type:

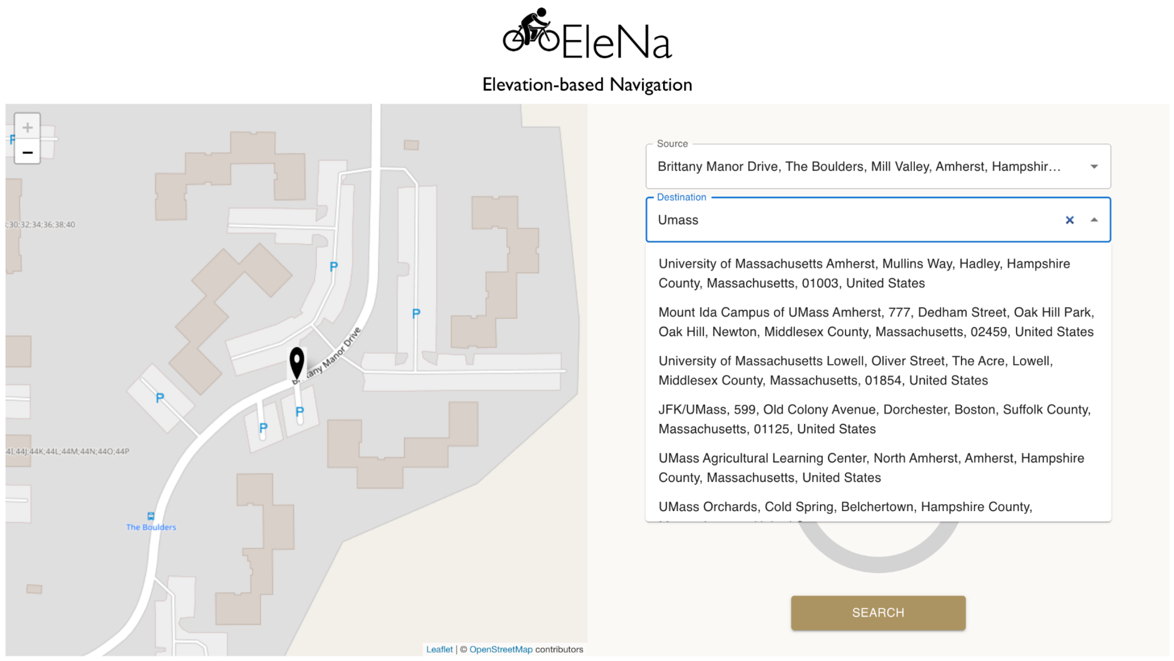
MIN ↓ MAX ↑

Percentage Increase From Shortest Distance:

125%

SEARCH

2. A component that populates the data model, querying.



EleNa

Elevation-based Navigation

Source: Brittany Manor Drive, The Boulders, Mill Valley, Amherst, Hampshir…

Destination: Umass

University of Massachusetts Amherst, Mullins Way, Hadley, Hampshire County, Massachusetts, 01003, United States

Mount Ida Campus of UMass Amherst, 777, Dedham Street, Oak Hill Park, Oak Hill, Newton, Middlesex County, Massachusetts, 02459, United States

University of Massachusetts Lowell, Oliver Street, The Acre, Lowell, Middlesex County, Massachusetts, 01854, United States

JFK/UMass, 599, Old Colony Avenue, Dorchester, Boston, Suffolk County, Massachusetts, 01125, United States

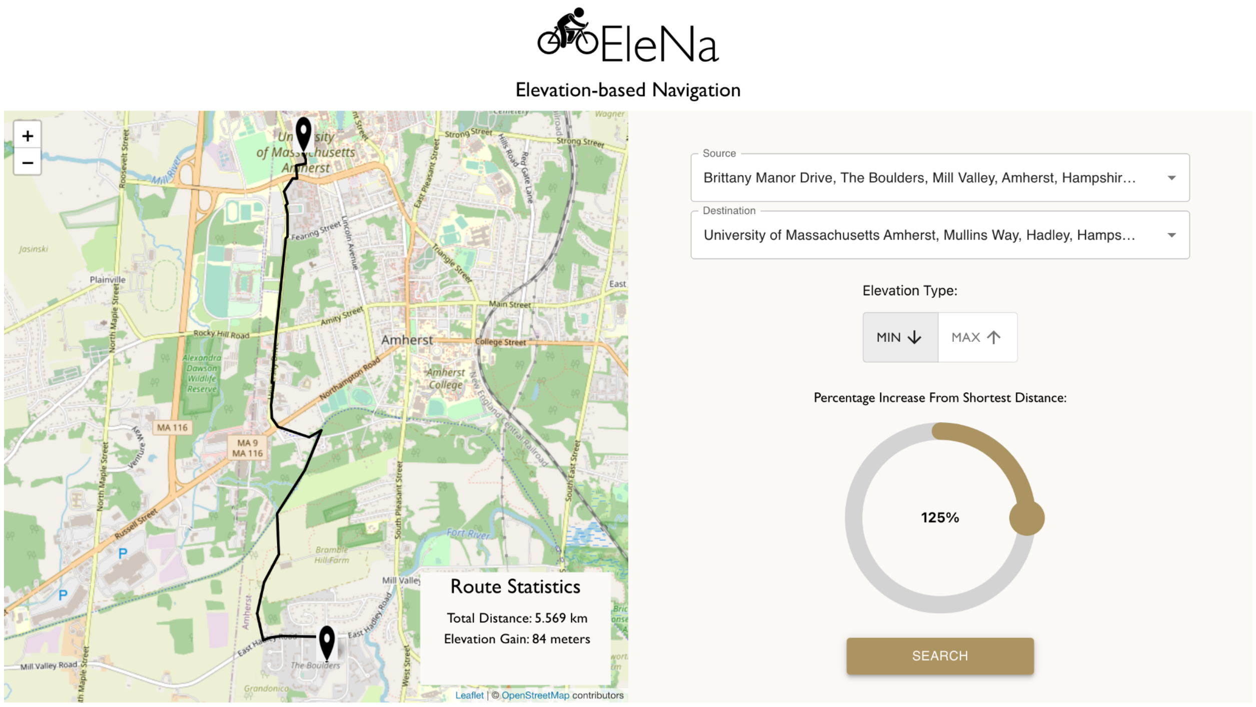
UMass Agricultural Learning Center, North Amherst, Amherst, Hampshire County, Massachusetts, United States

UMass Orchards, Cold Spring, Belchertown, Hampshire County,

SEARCH

3. The actual routing algorithm that performs the multi-objective optimization.

4. A component that outputs or renders the computed route.


EleNa
Elevation-based Navigation
Source
Brittany Manor Drive, The Boulders, Mill Valley, Amherst, Hampshir…
Destination
University of Massachusetts Amherst, Mullins Way, Hadley, Hamps…
Elevation Type:
MIN
MAX
Percentage Increase From Shortest Distance:
125%
SEARCH
Route Statistics
Total Distance: 5.569 km
Elevation Gain: 84 meters
Amherst
Leaflet | © OpenStreetMap contributors

5. Non - Functional Requirements

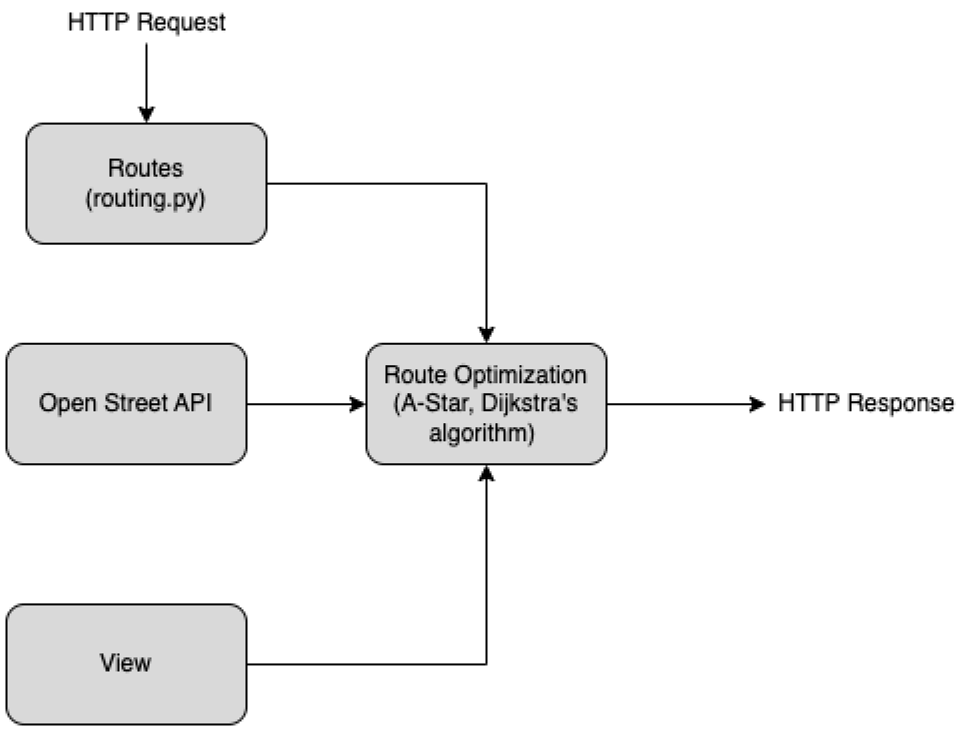
1. Performance Required
    a. Intuituive User Interface
    b. Low query time (used caching for faster results)
2. Security Required
    a. The behavior of the software must be correct and predictable.
    b. The application must not accept invalid data.

6. Deliverables

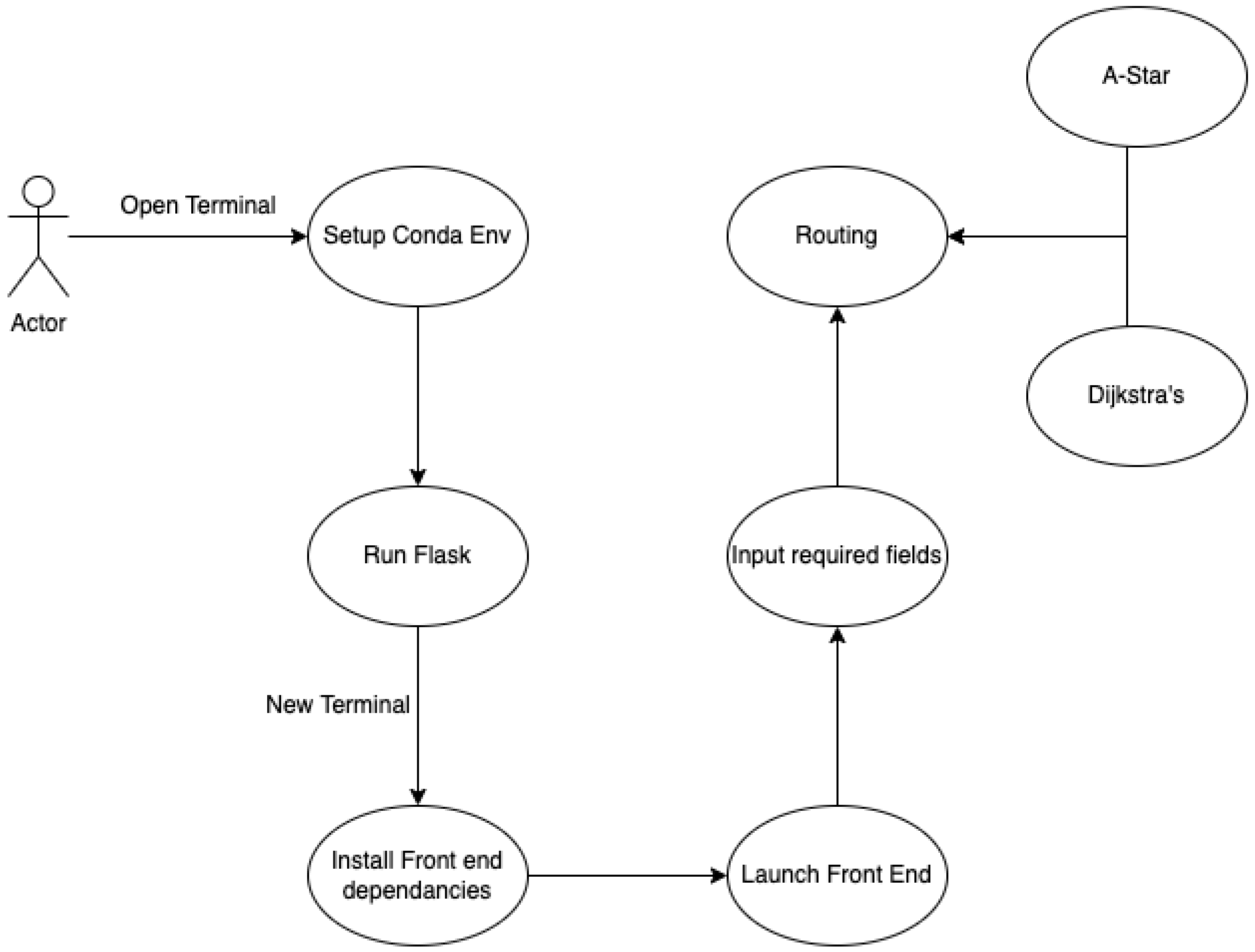
1. GitHub Repository link: https://github.com/vidhi-mody/CS520-EleNa
2. Steps to run the project:
   https://github.com/vidhi-mody/CS520-EleNa/blob/main/README.md

7. Diagrams

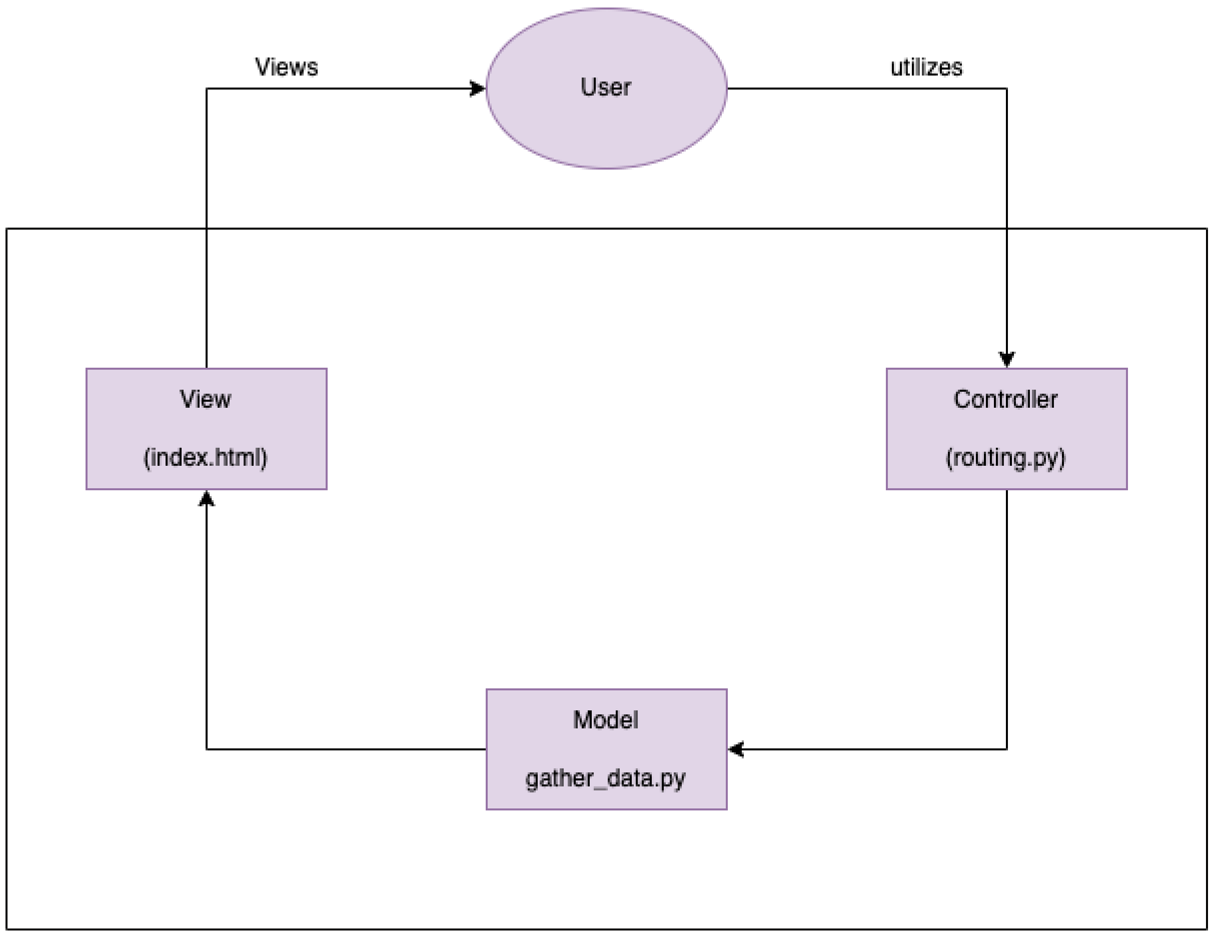
1. Application Design



2. Flow Diagram



3. Model View Controller Architecture


Views
User
utilizes
View
(index.html)
Controller
(routing.py)
Model
gather_data.py