DOMAIN PROJECT-1

SYNOPSIS

on

Topic Name: Pathfinder AI

S.No.	Student's Name	Roll Number	SAP Id
01	Keertivallabh Sharma	R2142220539	500107149
02	Jyotiraditya Singh	R2142220530	500107065
03	Shrey Sharma	R2142220169	500101806
04	Rishav Singh	R2142220149	500105685

Under the guidance of

Dr. Panduranga Ravi Teja

Professor

School of Computer Science

UNIVERSITY OF PETROLEUM AND ENERGY STUDIES

Dehradun-248007

Aug-Nov, 2024

Index

Chapter 1: Abstract

The rapid growth of urban areas and the increasing need for efficient transportation solutions demand innovative approaches for route planning and construction when connecting places like big cities. The government cannot solely rely on the shortest route to connect two cities; they must also consider optimal pathways that integrate major landmarks, key locations, and important destinations. Our project aims to develop a prototype application that employs a combination of pathfinding algorithms to find the optimal path for constructing a road connecting cities. The proposed solution will aim at allowing users to interactively upload city-wide maps, select specific destinations, and mark key points such as hospitals, offices, government buildings, airports, and stations. The application will compute the most efficient route based on factors like the priority of important locations. Through this project, we aim to demonstrate the feasibility and practicality of algorithm-driven decision-making in navigation applications, providing a valuable contribution to enhancing user navigation experiences.

Chapter 2: Introduction

Efficient route planning is an important aspect of improving urban mobility and reducing travel time, especially in developing nations like India, where expanding infrastructure and urbanization pose notable challenges.

This project focuses on creating a prototype web application that applies basic pathfinding algorithms to address some of these limitations in a simplified manner. The application allows users to interact with city-wide maps, select multiple destinations, and prioritize key locations such as hospitals, government offices, and airports. Based on these inputs, the system calculates routes that deviate from straight-line paths to account for the prioritization of these locations.

While this solution is limited in scope, it aims to demonstrate how even basic algorithmic approaches can be tailored to address specific use cases like urban navigation and planning. The project involves building a simple user interface, implementing algorithms to handle user inputs and map data, and testing the application on sample datasets to evaluate its functionality.

Through this effort, the project aims to highlight the potential for developing accessible tools that consider user priorities and practical constraints, providing a small but meaningful step toward better route planning in urban settings.

Chapter 3: Problem Statement

The goal of the project is to develop an application that will visualize a path between two given locations based on prioritized locations and landmarks; as a means to aid the construction of highways.

Chapter 4: Literature

Efficient design and planning for navigation have long been studied in the context of route optimization [1]. Various algorithms and models, such as Dijkstra’s algorithm, A* search, and genetic algorithms, have been employed to enhance route efficiency, reduce travel time, and minimize costs [2]. Dijkstra's algorithm, introduced in 1959, is a widely used method for finding the shortest path between nodes in a graph and is commonly applied in navigation problems. However, its traditional form does not account for multiple dynamic factors, such as traffic density, route feasibility, or the priority of essential locations, which are crucial in real-world scenarios [3].

Research on route optimization in diverse contexts highlights the importance of considering multiple criteria beyond mere distance [4]. Studies suggest that incorporating factors such as population density, location accessibility, economic benefits, and environmental impact can lead to more effective and user-friendly navigation solutions. Multi-criteria decision-making (MCDM) models and Geographic Information Systems have increasingly been utilized to improve the accuracy and relevance of navigation planning in various geographic and socio-economic contexts.

In recent years, there has been growing interest in integrating GIS with optimization algorithms to enhance navigation applications. It provides a powerful platform for handling large spatial datasets and visualizing complex geographic information, while algorithms like Dijkstra’s can perform efficient computations on these datasets. Previous works have demonstrated that combining with optimization techniques allows for more interactive, data-driven navigation processes that consider multiple constraints and objectives [5].

This project builds on the existing literature by developing a prototype navigation application that integrates modified pathfinding algorithms with GIS-based tools. Unlike traditional approaches, our application allows users to interactively upload maps, select destinations, and mark critical locations, considering diverse criteria such as traffic density, route feasibility, and strategic importance. This approach aligns with the needs of diverse regions, where varied geographic, demographic, and economic conditions require adaptable and context-sensitive navigation tools [6]. By leveraging the strengths of both classic algorithms and modern GIS technology, this project aims to contribute to more efficient, user-friendly, and data-driven navigation solutions.

Chapter 5: Objectives

1. **Optimal Route Planning**

    - Develop a road/path constructing algorithm that can visualize a path between two locations and skew accordingly, based on important locations.

2. **Real-Time Data Integration**

    - Incorporate real-time geographic data and map APIs (e.g., Google Maps) to provide accurate and dynamic route planning.

3. **User-Centric Features**

    - Enable users to upload custom city-wide maps, mark critical locations (hospitals, offices, stations, etc.), and modify the straight path between the two locations, based on the critical locations.

4. **Algorithm Visualization**

    - Implement and visually represent popular pathfinding algorithms like Dijkstra, curve drawing algorithms etc.

5. **Decision-Making Support**

    - Provide actionable insights for real-world applications like highway or road construction planning by optimizing routes based on cost, distance, and priority factors.

6. **Interactive and Intuitive Interface**

    - Deliver a user-friendly interface that simplifies interaction with the system, making it easier to upload maps, visualize routes, and analyze the results.

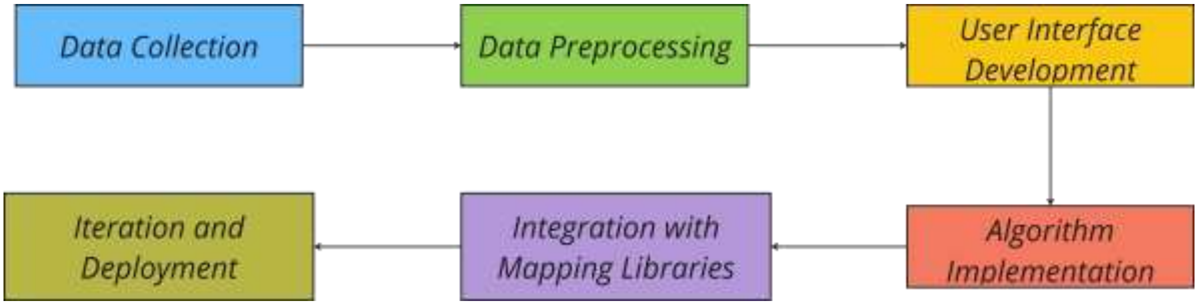
7. **Cost and Impact Estimation**

    Developing a report generation system that:

    - Estimates construction costs based on road length.
    - Evaluates environmental impacts like forest cover cleared.
    - Calculates estimated construction time and employment opportunities.

Chapter 6: Methodology

The project starts with *Data Collection*, where city-wide maps are obtained from various sources like Open Street Maps (OSM), Google Maps Platform, ArcGIS etc. The next phase is *Data Pre Processing*, in this phase the obtained maps are cleaned and converted into a compatible format for the application. Next, comes *UI development* focusing on creating an interactive user interface for convenience and easy navigation. Following this, the *Algorithm implementation* phase involves applying modified pathfinding algorithms like Dijkstra's, A*, etc., to determine optimal pathways for road construction. The application is then enhanced through integration to visualize the maps and roads effectively. Finally, the project undergoes iteration and deployment, where we will test it and iterate over times, to refine it to satisfaction.



Data Structures Used:

- **Priority Queue (Min-Heap)**
  Used In: Dijkstra's Algorithm
  Purpose: Efficiently retrieves the next node with the smallest tentative distance during pathfinding.
  Significance: Optimizes the performance of the shortest path computation, reducing time complexity.

- **Graph Representation (Adjacency List/Matrix)**
  Used In: Dijkstra's Algorithm, De Casteljau Algorithm
  Purpose: Represents the network of nodes and edges, including weights or control points for curves.
  Significance: Provides a structured way to traverse and manipulate the network for route planning and visualization.

- **Dynamic Arrays (for Control Points)**
  Used In: De Casteljau Algorithm, Bézier Curves
  Purpose: Stores and processes control points to compute smooth curves.
  Significance: Enables iterative calculations and recursive subdivisions required for creating Bézier curves.

- **Stack/Recursion Stack**
  Used In: De Casteljau Algorithm
  Purpose: Handles recursive subdivision of control points to generate Bézier curves.
  Significance: Simplifies the computation of smooth curves by leveraging recursive principles.

- **2D Coordinate Arrays**
  Used In: Bézier Curves
  Purpose: Stores x and y coordinates for nodes and control points.
  Significance: Facilitates accurate plotting and visualization of curves and paths.

SWOT Analysis:

- **Strengths:**
  Pathfinder AI leverages advanced algorithms like Dijkstra's and Bézier curves, offering robust computational capabilities. Its user-friendly design and scalability make it suitable for practical applications in navigation, urban planning, and logistics. Additionally, its customizability enhances user engagement and adaptability.

- **Weaknesses:**
  The project faces challenges with computational efficiency for large datasets and has a learning curve for less tech-savvy users. Its reliance on accurate map data and the time-intensive nature of development pose potential limitations.

- **Opportunities:**
  There is significant potential for integration with AI, IoT, and real-time data analytics, opening doors to predictive modeling and enhanced functionality. Collaboration with GIS providers and expansion into new markets like logistics and autonomous systems could further solidify its impact.

- **Threats:**
  Pathfinder AI competes with established navigation solutions like Google Maps. Dependency on third-party map data and evolving technological landscapes could challenge sustainability. Additionally, legal and privacy concerns may arise when handling user-provided map data.

Strengths: Advanced algorithms Visual appeal Practical use cases Scalability Customizability	**Weaknesses:** High computational demand Learning curve Limited scope without maps Time constraints
Opportunities: Integration with emerging technologies Expansion into new markets Collaborations User-centric features	**Threats:** Competition Data dependency Technological changes Legal and privacy concerns

Areas of Application

The Pathfinder AI project, leveraging pathfinding algorithms like Dijkstra's, A*, and Breadth-First Search (BFS), offers significant potential in various fields. By integrating mapping technologies like OpenStreetMap and implementing algorithm visualizations, this project can serve multiple industries. Below are key areas where Pathfinder AI can be applied:

1. Urban Planning and Infrastructure Development

Pathfinder AI can be highly valuable in urban planning, particularly in road construction and infrastructure development. The project's ability to calculate the most efficient routes and optimal paths can assist urban planners in designing transportation systems, highways, and urban layouts. By analyzing traffic flow, urban density, and geographical constraints, the system can suggest better routes for public transportation, reduce congestion, and improve city layouts. The inclusion of control points allows for more flexibility in adapting designs to fit specific city needs.

2. Logistics and Supply Chain Management

In logistics, efficient route planning is essential to reduce transportation costs and delivery times. Pathfinder AI can be used by logistics companies to optimize routes for trucks, delivery vehicles, and supply chains. By implementing algorithms like Dijkstra's and A*, the system can determine the shortest and most cost-effective paths for delivery, taking into account road conditions, traffic, and time constraints. This optimization leads to significant cost savings and improved operational efficiency in industries such as e-commerce, food delivery, and freight management.

3. Emergency Response and Disaster Management

Pathfinder AI can also be applied in emergency response and disaster management, where quick decision-making is critical. For first responders like ambulances, fire trucks, and rescue teams, optimizing routes in real time can save lives. It can analyze current road conditions, traffic data, and disaster zones to propose the fastest route to a specific location. With the ability to integrate with live data feeds, the system can offer dynamic route adjustments as new obstacles or changes in the situation arise, ensuring that emergency services reach their destinations as quickly as possible.

4. Geographic Information Systems (GIS) and Mapping

As an advanced tool for geographic data analysis, Pathfinder AI can be integrated into Geographic Information Systems (GIS) for mapping, spatial analysis, and environmental monitoring. GIS professionals can use the pathfinding algorithms to study and visualize geographic data, calculate distances, and determine optimal paths across terrains, such as for pipelines, cables, or conservation efforts. The system can also be used in flood modeling, land-use planning, and environmental conservation to help create more sustainable and effective strategies for land management.

Chapter 7: System Requirements

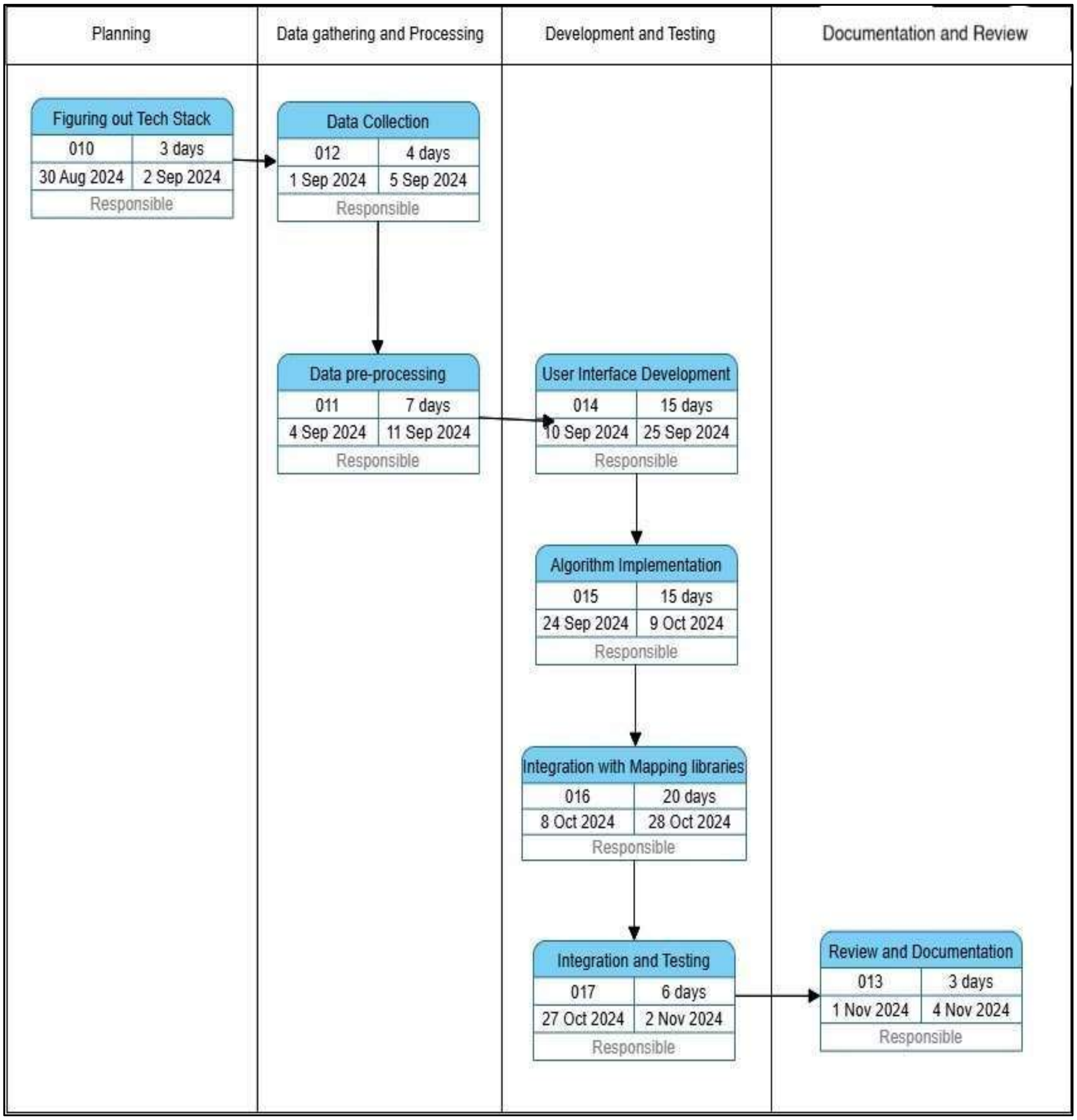
1. **Software Requirements**

Operating System	:	Windows 10/8/7 (32-bit or
64-bit)/ Linux Software	:	Text Editor, Browser
Compiler	:	GCC, Python

2. **Hardware Requirements**

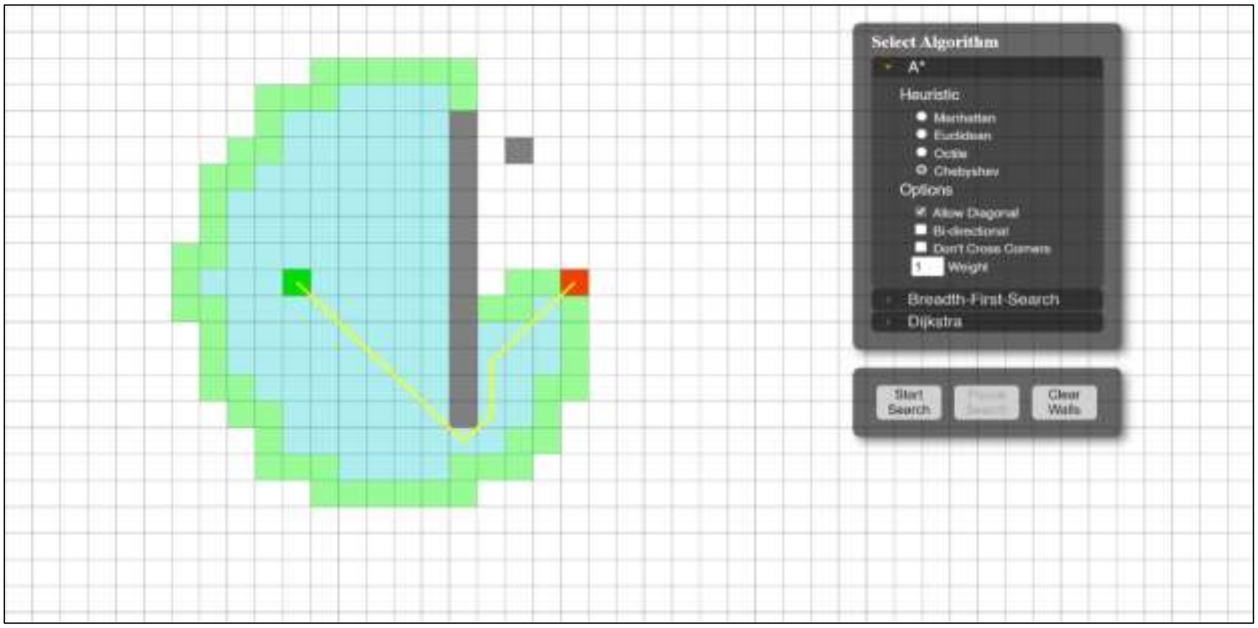
Processor	:	Dual Core 2.7 GHz or better
RAM	:	512 MB or higher
Disk Space	:	512 MB

Chapter 8: PERT Chart



Chapter 9: Visual Documentation

Developed algorithm visualization as an aid for implementing the algorithms onto map datapoints.



Select Algorithm

A*

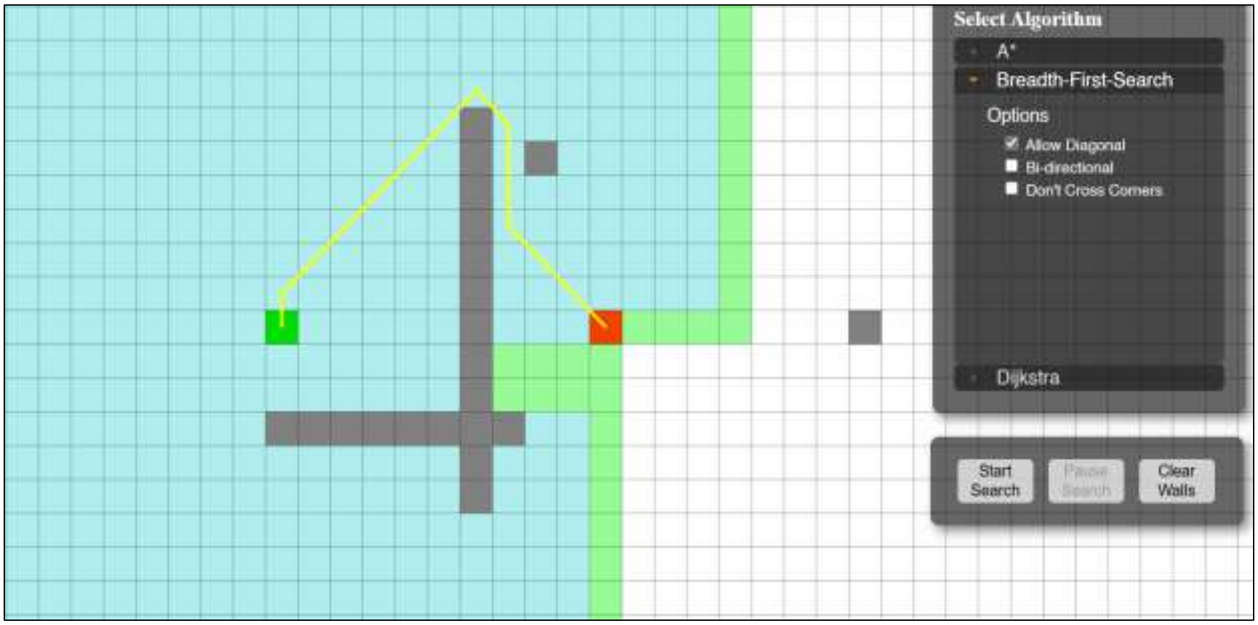
Heuristic

● Manhattan
● Euclidean
● Octile
○ Chebyshev

Options

☑ Allow Diagonal
□ Bi-directional
□ Don't Cross Corners
1 Weight

Applied algorithms like Djikstra, A*, Breadth-First Search for visualization.



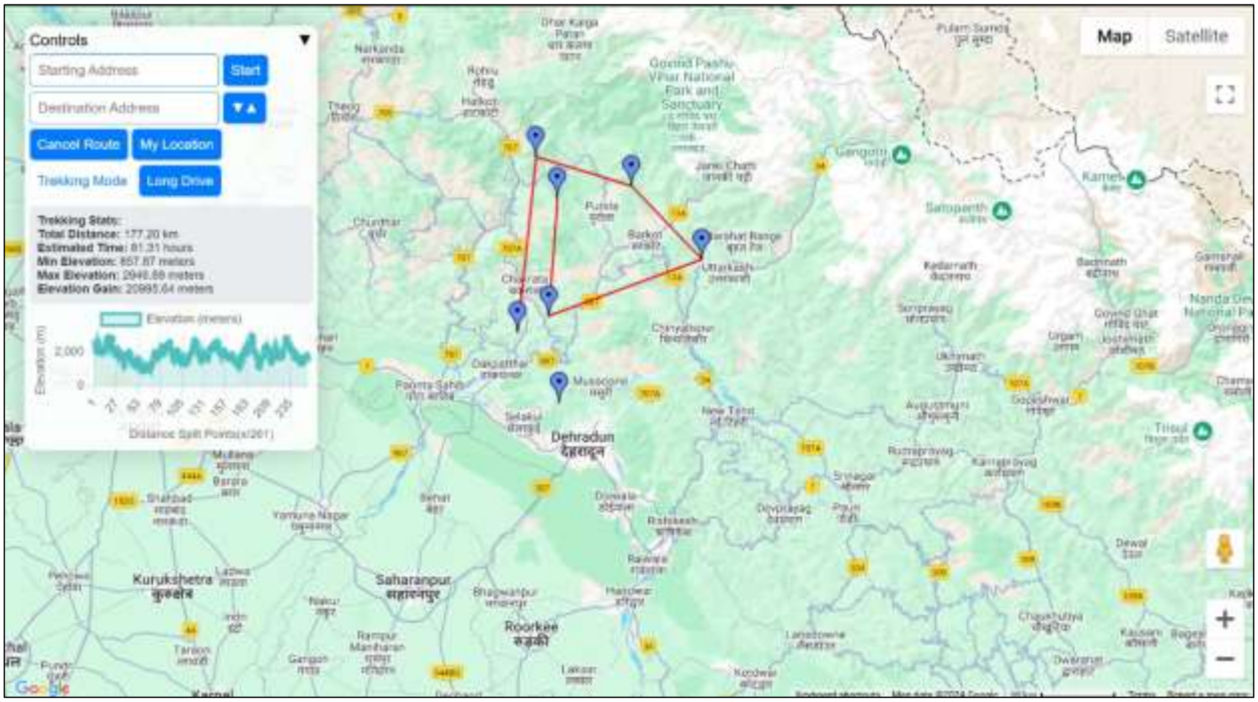
Select Algorithm

Breadth-First-Search

Options

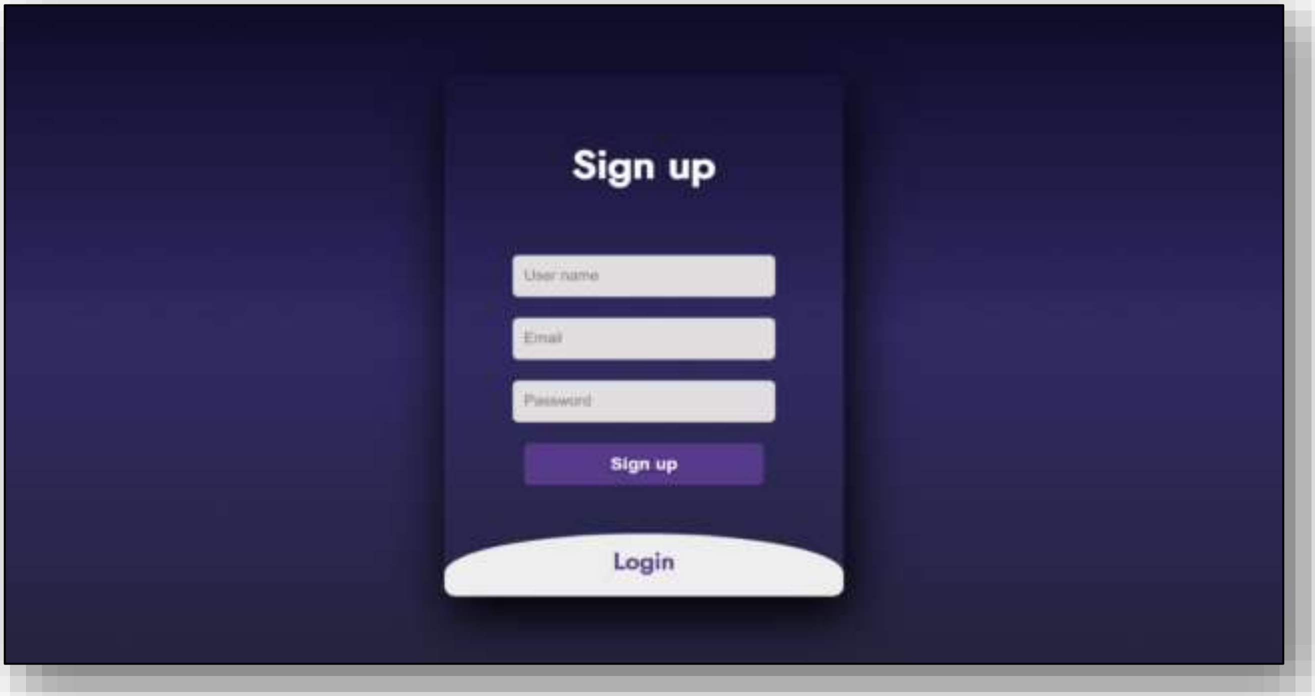
☑ Allow Diagonal
□ Bi-directional
□ Don't Cross Corners

Integrated Open Street Maps into the codebase and implemented features such as straight distance mapping.

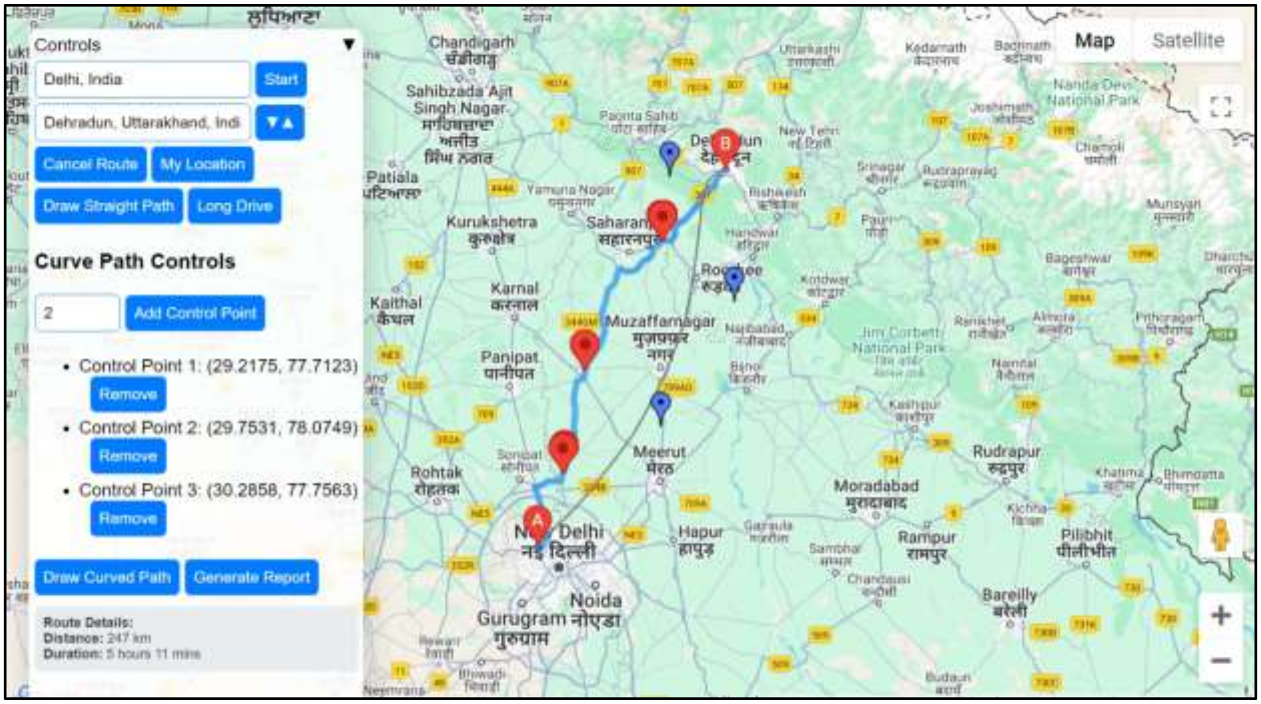


Trekking Stats:
Total Distance: 177.20 km
Estimated Time: 81.31 hours
Min Elevation: 857.87 meters
Max Elevation: 2946.89 meters
Elevation Gain: 20995.64 meters

Created a sign-up page for the application, allowing new users to register and create an account for access to the system.



Implemented Bezier curves using De Casteljau's algorithm to enhance straight paths by introducing smoother, more flexible modifications for improved trajectory adjustments.



Controls

Delhi, India

Dehradun, Uttarakhand, Indi

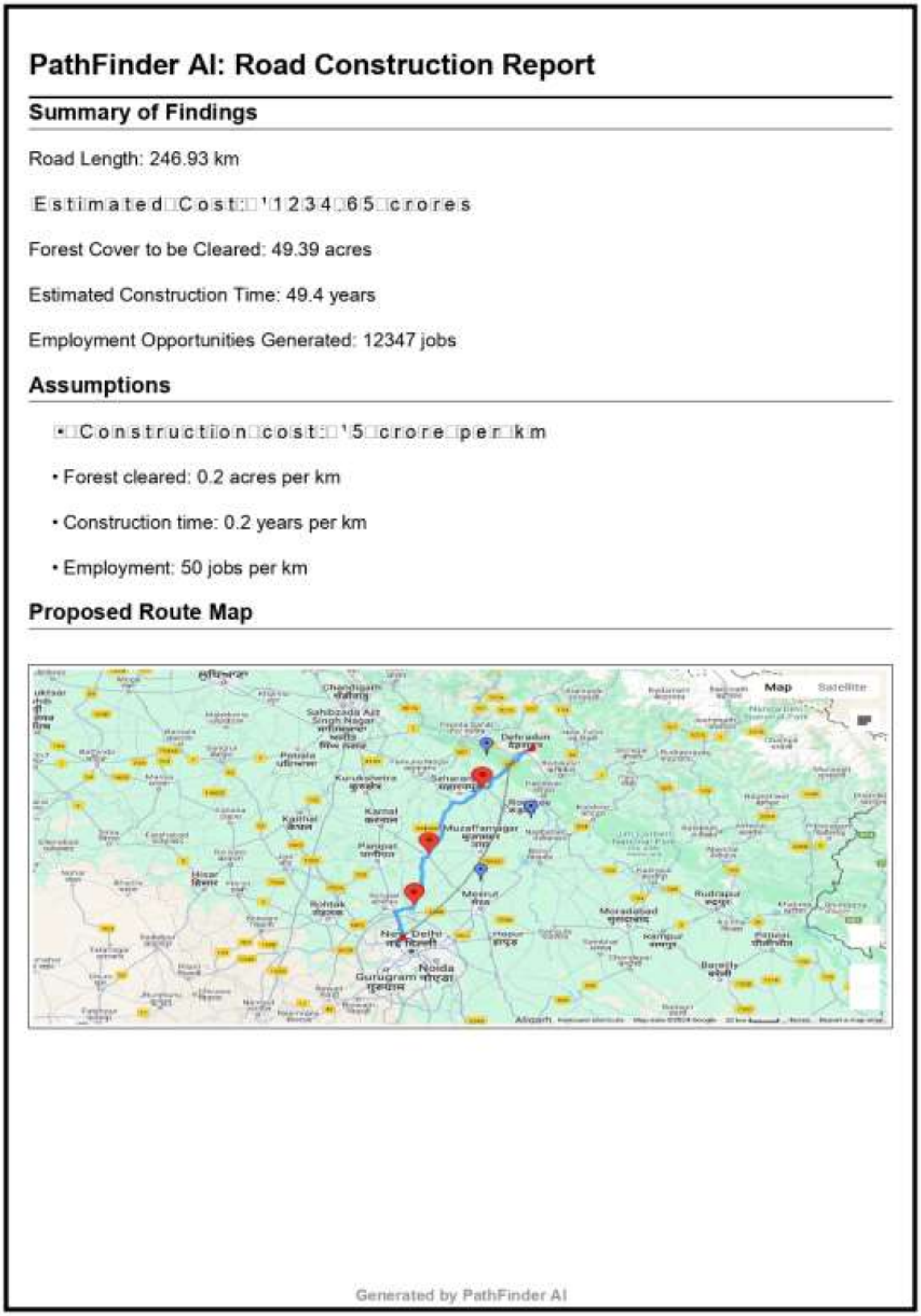
Curve Path Controls

2

- Control Point 1: (29.2175, 77.7123)
- Control Point 2: (29.7531, 78.0749)
- Control Point 3: (30.2858, 77.7563)

Route Details:
Distance: 247 km
Duration: 5 hours 11 mins

Developed a report generation functionality to create detailed, automated reports summarizing route planning, algorithm usage, and performance insights for easy analysis and sharing.



PathFinder AI: Road Construction Report

Summary of Findings

Road Length: 246.93 km

Estimated Cost: '1234.65 crores

Forest Cover to be Cleared: 49.39 acres

Estimated Construction Time: 49.4 years

Employment Opportunities Generated: 12347 jobs

Assumptions

- Construction cost: '5 crore per km
- Forest cleared: 0.2 acres per km
- Construction time: 0.2 years per km
- Employment: 50 jobs per km

Proposed Route Map

Generated by PathFinder AI

Achieved Deliverables:

1. **Integration of OpenStreetMap**

   Successfully integrated OpenStreetMap data into the codebase for accurate mapping and route planning.

2. **Straight Distance Mapping Feature**

   Implemented the ability to calculate and display straight-line distance between selected points on the map.

3. **Algorithm Implementation**

   Applied and integrated the following algorithms:

   - Dijkstra’s Algorithm
   - A Algorithm*
   - Breadth-First Search (BFS)

   for efficient pathfinding and route optimization.

4. **Algorithm Visualization**

   Developed visualizations for the implemented algorithms, aiding in better understanding and analysis of algorithmic processes on map datapoints.

5. **Map DataPoint Integration**

   Implemented functionality to apply pathfinding algorithms to specific map datapoints, allowing users to visualize optimal routes on a map.

6. **User Interaction Features**

   Developed user interface features to allow users to interact with the map, select points, and visualize paths computed by the algorithms.

7. **Efficient Visualization for Algorithm Comparison**

   Created an effective platform to compare the performance and results of different pathfinding algorithms through visual representation on the map.

8. **Route Visualization**

   Successfully implemented route plotting on the map using Google Maps API.

   Integrated functionality for displaying straight paths and curved paths (Bézier curves) between selected points.

9. **Pathfinding with Dijkstra's Algorithm**

   Implemented Dijkstra's algorithm for calculating the shortest path between two points.

   Enabled dynamic graph generation from user-added control points and markers.

   Displayed the shortest path on the map as a highlighted polyline.

10. **User Interaction Features**

    Enabled users to add and remove control points dynamically on the map.

    Provided interactive options to reverse the route or adjust markers for flexible route planning.

11. **Cost and Impact Estimation**

    Developed a report generation system that:

Estimates construction costs based on road length.

Evaluates environmental impacts like forest cover cleared.

Calculates estimated construction time and employment opportunities.

12. **Map Integration**

    Integrated Google Maps API with support for:

    Real-time geocoding of start and end points.

    Autocomplete for user inputs (e.g., start and end locations).

    Calculation of distances and directions between points.

13. **Elevation Data Integration**

    Incorporated elevation data for trekking routes using Google Maps Elevation API.

    Displayed elevation statistics, such as total gain and estimated trekking time, for user selected paths.

14. **Customizable Trekking and Long Drive Modes**

    Implemented a trekking mode with support for adding multiple trekking markers and calculating cumulative distances.

    Developed a long drive mode for generating scenic routes and evaluating round-trip paths.

15. **Map Interaction Enhancements**

    Allowed users to interact with the map for:

    Adding custom control points by clicking.

    Viewing landmarks and nearby places dynamically based on the chosen path.

16. **Report Generation**

    Automated generation of a detailed PDF report with:

    Route statistics, cost, and time estimations.

    Impact analysis and recommendations for sustainable road construction.

    Screenshot of the proposed route map.

Future Refinements:

- Improvising the algorithm associated with deviation for the roadway.

- Adding features like heatmaps, 3d integration etc.

- Options to upload custom city-wide maps, to be able to work with various formats like .tiff, .bmp, .shp, geojson and other raster and vector formats.

- Report generation mentioning details like cities selected, selected route, cost estimation, impact etc.

- Adding educational features, that gude the user through the interface of application.

- Integrate Elevation and other factorsi like community dynamics, Historical Data Analysis, job opportunities etc. for a more comprehensive view of the application.

Chapter 10: References

Github Repository:
https://github.com/Slyphx/Minor--1

- https://docs.mapbox.com/help/glossary/mapbox-js/
- Google Maps Platform. (n.d.). Google Maps API Documentation.
- Project Management Institute. (2017). A Guide to the Project Management Body of Knowledge (PMBOK Guide) (6th ed.).
- https://www.youtube.com/watch?v=J7eyI1ClDOA
- https://www.iirs.gov.in/iirs/sites/default/files/pdf/2023/Overview_of_Remote_Sensing_and_GIS_Applications_E-Book.pdf
- https://www.sciencedirect.com/journal/isprs-journal-of-photogrammetry-and-remote-sensing
- [1] https://www.researchgate.net/publication/route_optimization_design
- [2] https://en.wikipedia.org/wiki/Dijkstra%27s_algorithm
- [3] https://www.journalofnavigation/algorithm_limitations
- [4] https://www.sciencedirect.com/multi_criteria_navigation
- [5] https://www.researchgate.net/gis_optimization_techniques
- [6] https://www.mdpi.com/data_driven_navigation_solutions