


PROJECT AND TEAM INFORMATION

Project Title

Smart Travel Planner using Dijkstra Algorithm for Shortest Path and Cost Estimation.

Student / Team Information

<p><i>Team Name:</i></p>	<p><i>Super Kings</i></p>
<p>Team member 1 (Team Lead) <i>Name-Kartik Khatri</i> <i>Student id - 2520230252</i> <i>Email id-Kartikkhatri376@gmail.com</i></p>	
<p>Team member 2 <i>Name -Jatin Kadiyan</i> <i>Student id -2520230400</i> <i>Email id- Kadiyanjatin28@gmail.com</i></p>	
<p>Team member 3 <i>Name-Kaustubh Jiwal</i> <i>Student id- 2520230053</i> <i>Email id- jiwalkostubh@gmail.com</i></p>	

PROPOSAL DESCRIPTION

Motivation

- > Many people face difficulty in finding the shortest and cheapest route between two places.
- > Manual route selection may take more time and may not give optimal results.
- > There is a need of a smart system that can calculate shortest path automatically.
- > This project uses Dijkstra Algorithm to find shortest distance between source and destination.
- > It shows how Graph data structure can solve real-world travel problems efficiently.

State of the Art / Current solution

- > Today, applications like Google Maps provide route and distance information.
- > These systems use complex algorithms and real-time traffic data.
- > However, they do not clearly explain how shortest path algorithms work.
- > Our project focuses on simple implementation of Dijkstra Algorithm for learning purpose.
- > It provides shortest path and approximate travel cost in an easy way.

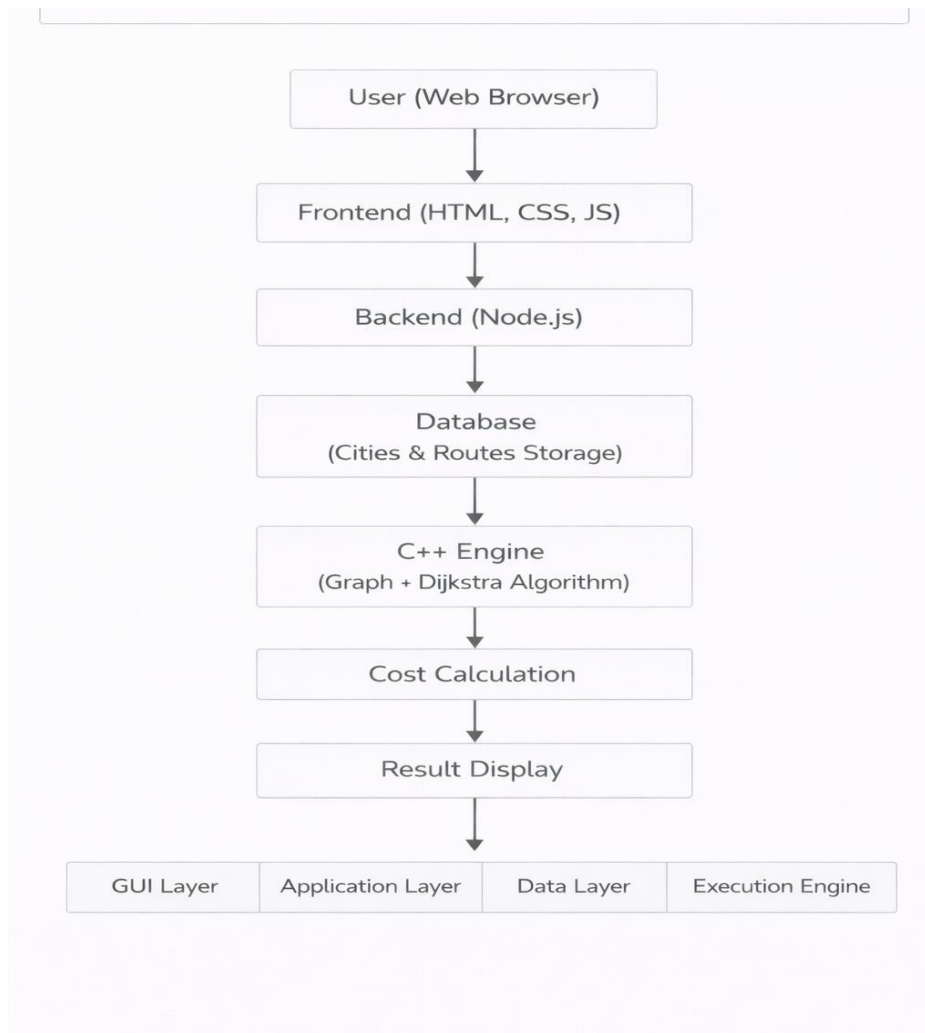
Project Goals and Milestones

- > To design and develop a Smart Travel Planner system using Graph data structure.
- > To implement Dijkstra Algorithm for calculating the shortest path between source and destination.
- > To calculate approximate travel cost based on total distance.
- > To represent cities as nodes and routes as edges using adjacency matrix or list.
- > To create a simple menu-driven interface for user input and output display.
- > To integrate shortest path module with cost calculation module.
- > To test the system with different inputs to ensure accuracy and efficiency.
- > To debug errors and improve system performance.
- > To prepare final documentation and complete project submission.

Project Approach

- > The system will be developed in a modular way.
- > The frontend will be created using HTML, CSS, and JavaScript for user interaction.
- > Node.js will be used as the backend to handle user requests.
- > A database will be used to store cities, routes, and distance data.
- > The backend will fetch required data from the database.
- > Graph will represent cities as nodes and roads as edges.
- > Dijkstra Algorithm (implemented in C++) will compute the shortest path.
- > Cost will be calculated using $\text{total distance} \times \text{rate per km}$.
- > The final result (shortest path and total cost) will be displayed to the user.
- > Development and testing will be done using an IDE.

System Architecture (High Level Diagram)



Project Outcome / Deliverables

- > Fully working Travel Planner system.
- > Implementation of Dijkstra Algorithm.
- > Shortest path and cost calculation feature.
- > Clear understanding of Graph and Algorithm usage.
- > Final source code and documentation.

Assumptions

- > Number of cities is fixed.
- > Distances between cities are predefined.
- > No real-time traffic consideration.
- > Cost is approximate calculation only.

References

GeeksforGeeks – Dijkstra Algorithm
Data Structures Notes
Official C/C++ Documentation