

# OPTIMAL ROUTE PLANNER

B i n a r y X O R

## TEAM

Paridhi Bhagwat  
Vaishnavi Mane  
Avani Chawardol  
Siddhi Dubewar

# PROBLEM STATEMENT

Design an efficient route planner that helps users find the shortest path between two locations in a city



# ANALYSIS

## DIJKSTRA'S ALGORITHM

Guarantees optimality and simplicity, but may be inefficient for large graphs and cannot handle negative edge weights.

## A SEARCH\*

Reduces search space by exploring from both start and goal nodes simultaneously, but implementation complexity is higher.

## BIDIRECTIONAL SEARCH

Efficient and optimal, but heavily reliant on heuristic quality and can be complex to implement due to heuristic function requirements



# OUR SOLUTION

Our solution approach involves two key components:

1. Clustering destinations
2. Traveling Salesman Problem (TSP) for each cluster

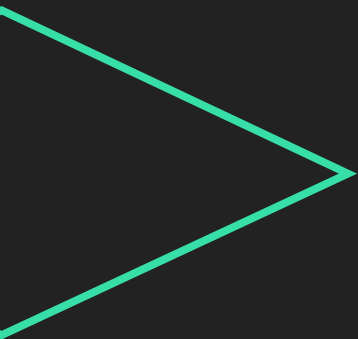
To achieve the clustering, we used the Gaussian Mixture Model (GMM), a probabilistic model that identifies clusters in our dataset.

Once destinations are clustered, we then apply the Traveling Salesman Problem (TSP) solver to find the optimal route for each cluster.

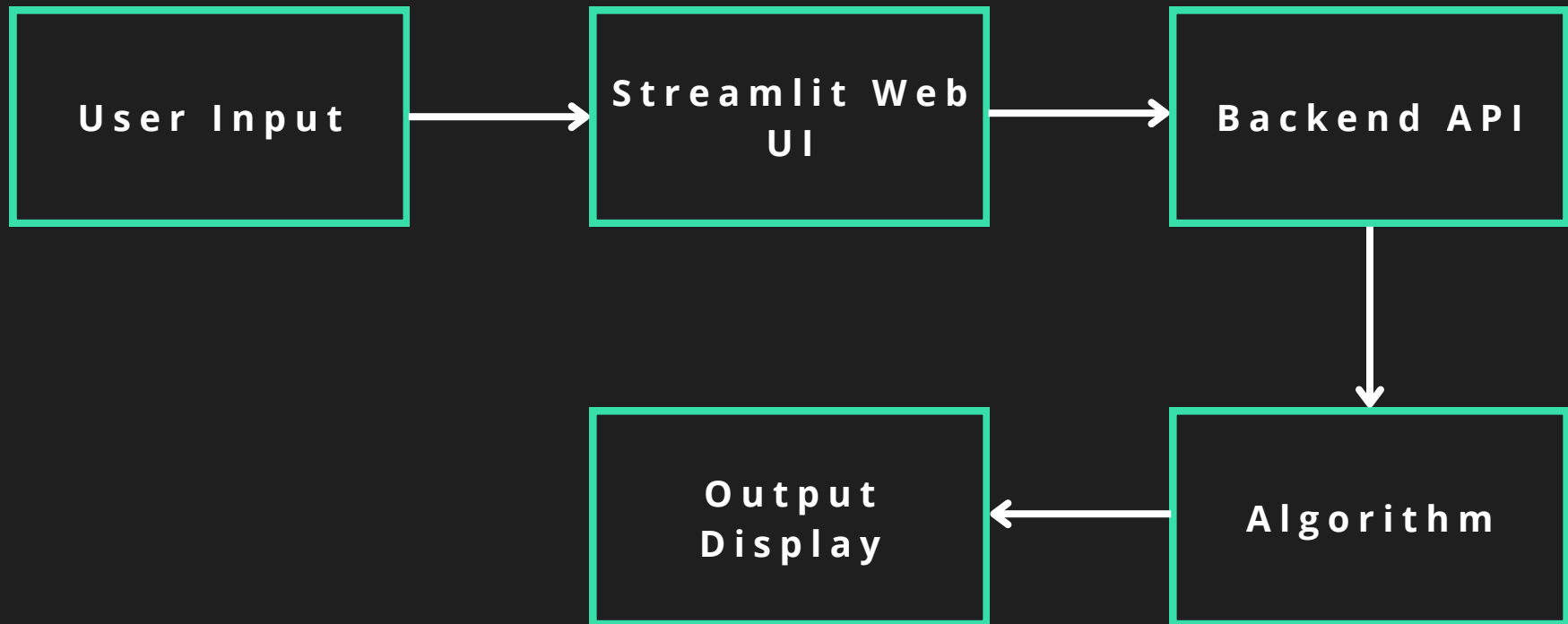


# TECHNOLOGIES USED



- **STREAMLIT:** Used for building the user interface and interaction
  - **FOLIUM:** Used for visualizing maps and markers
  - **OR-TOOLS:** Used for solving the Traveling Salesman Problem.
  - **GEOPY:** Used for geocoding destinations and obtaining coordinates.
- 

# ARCHITECTURE



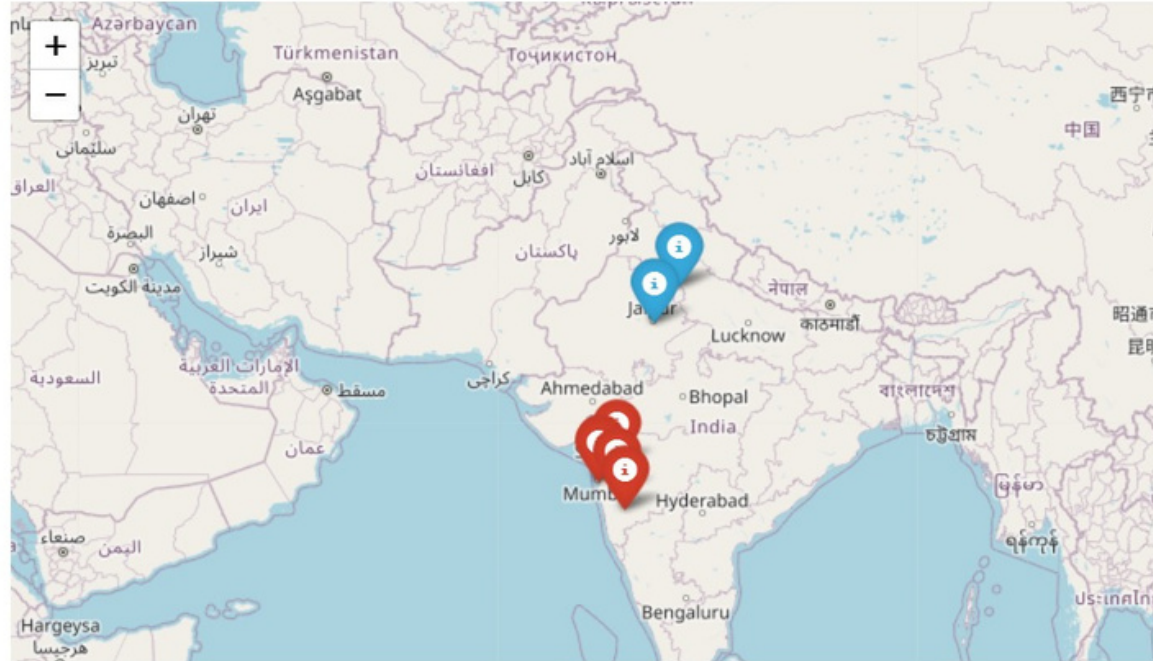
# DEMONSTRATION

## Travel Route Optimizer

Enter destinations (comma-separated):

Mumbai, thane, Pune, Delhi, Satara, Jaipur, Nashik

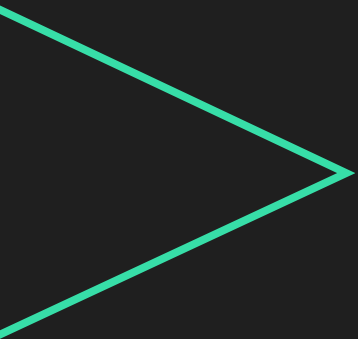
Number of Clusters:



# CONCLUSION



In summary, we've explored an optimized solution for planning travel routes, leveraging clustering and TSP solving techniques. By clustering destinations based on geographical proximity and applying TSP algorithms within each cluster, we've demonstrated an effective approach for optimizing travel routes.







**THANK YOU**