Intel Unnati Industrial Training Program 2024

Team Alpha Coders

Problem Statement

GPS Toll Based System Simulation Using Python

Brief Solution

- Using Dijkstra’s Algorithm to calculate the distance between various cities
- Predefining the toll rates for a set of cities and distances between them
- Animating the Vehicular movement using Pygame library
- Using Tkinter and Networkx libraries for Visualising of graph

Features Offered

- Has a user interface for dynamic entry of Source and destination locations
- User can view the movement of vehicle and the path used through a animation created using pygame library
- One can view the toll value instantly as he views the animated path , alongside can view a graph matrix of all the various paths and the chosen path

Process flow

Initialization :

- Initializes Pygame and Tkinter components.
- Sets up global variables, constants, and data structures.

User Interaction :

- Users input start and end points via Tkinter GUI.
- Clicking "Calculate Toll" triggers computation of the shortest path and toll.
- Clicking "Show Graph" displays the graph with highlighted shortest path.

Graph and Animation :

- Uses NetworkX for graph operations (construction, shortest path).
- Utilizes Pygame for vehicle animation along the calculated shortest path.

Visualization :

- Matplotlib is used to visualize the graph within the Tkinter GUI

Technologies Used

Python Libraries:

- **NetworkX :** Used for graph creation and shortest path calculations (Dijkstra's algorithm).
- **Heapq :** Used for implementing the priority queue in Dijkstra's algorithm.
- **Pygame :** Used for animating the vehicle movement on the GUI.
- **Tkinter :** Used for creating the main GUI where users can input start and end points and view results.
- **Threading :** Used to run the animation in a separate thread to keep the GUI responsive.
- **Matplotlib :** Used to visualize the graph with the shortest path highlighted.

Global Variables and Initializations:

- **Pygame Initialization :** Setting up the Pygame window and defining colors.
- **Tkinter Initialization :** Setting up the Tkinter window and widgets.
- **Graph Positions :** Predefined positions for cities used in the Pygame animation.

Team Members and Contribution

Abhinav P V - Team lead, Contention creation and Report generation

Ashutosh Shukla - Writing Python code, Animation and Generation of Visuals

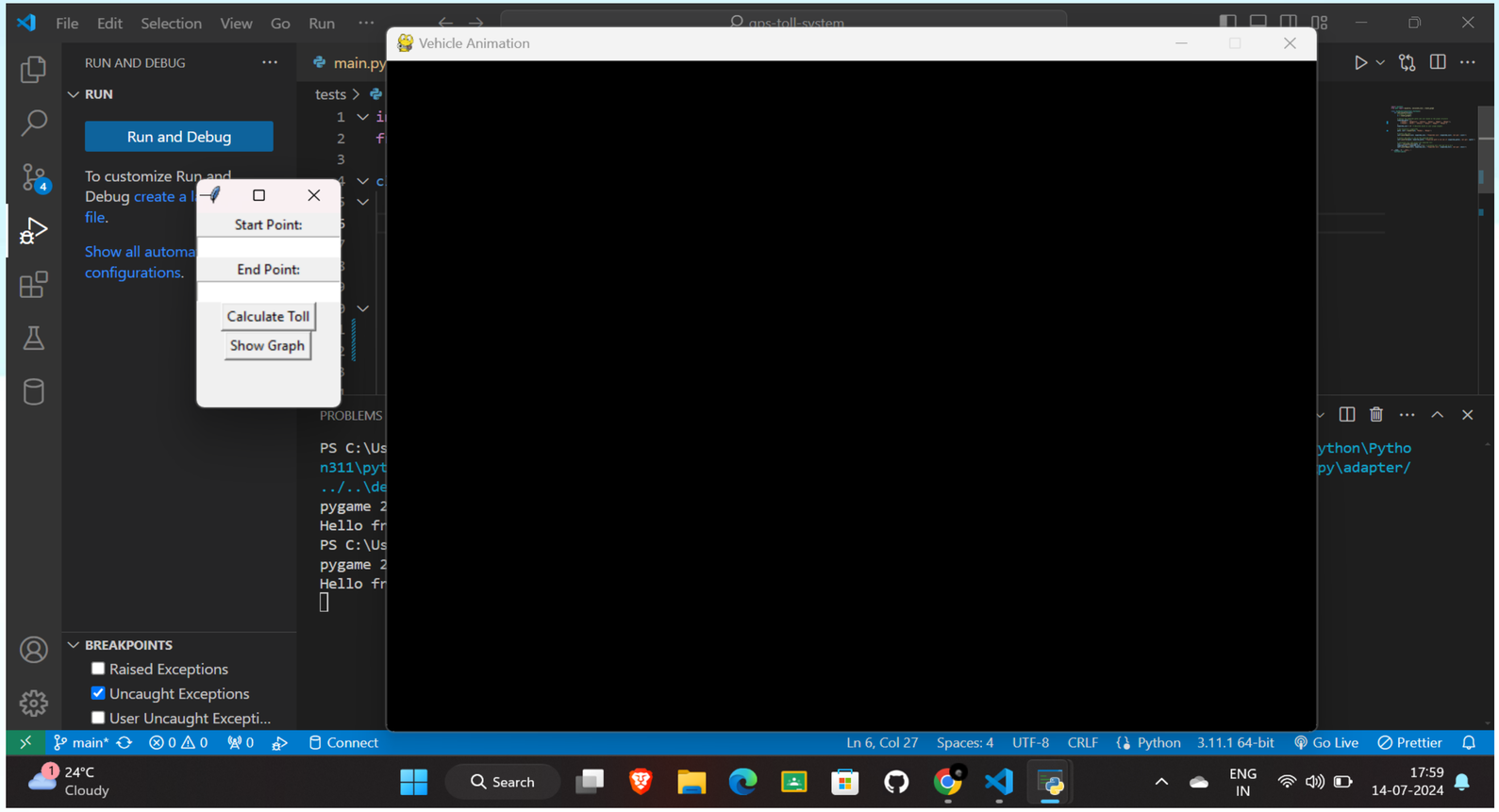
Ajitesh Kumara - Documentation, Sequence Diagrams and Flowcharts

Devender Singh - Inputting ideas and Enhancement of Code

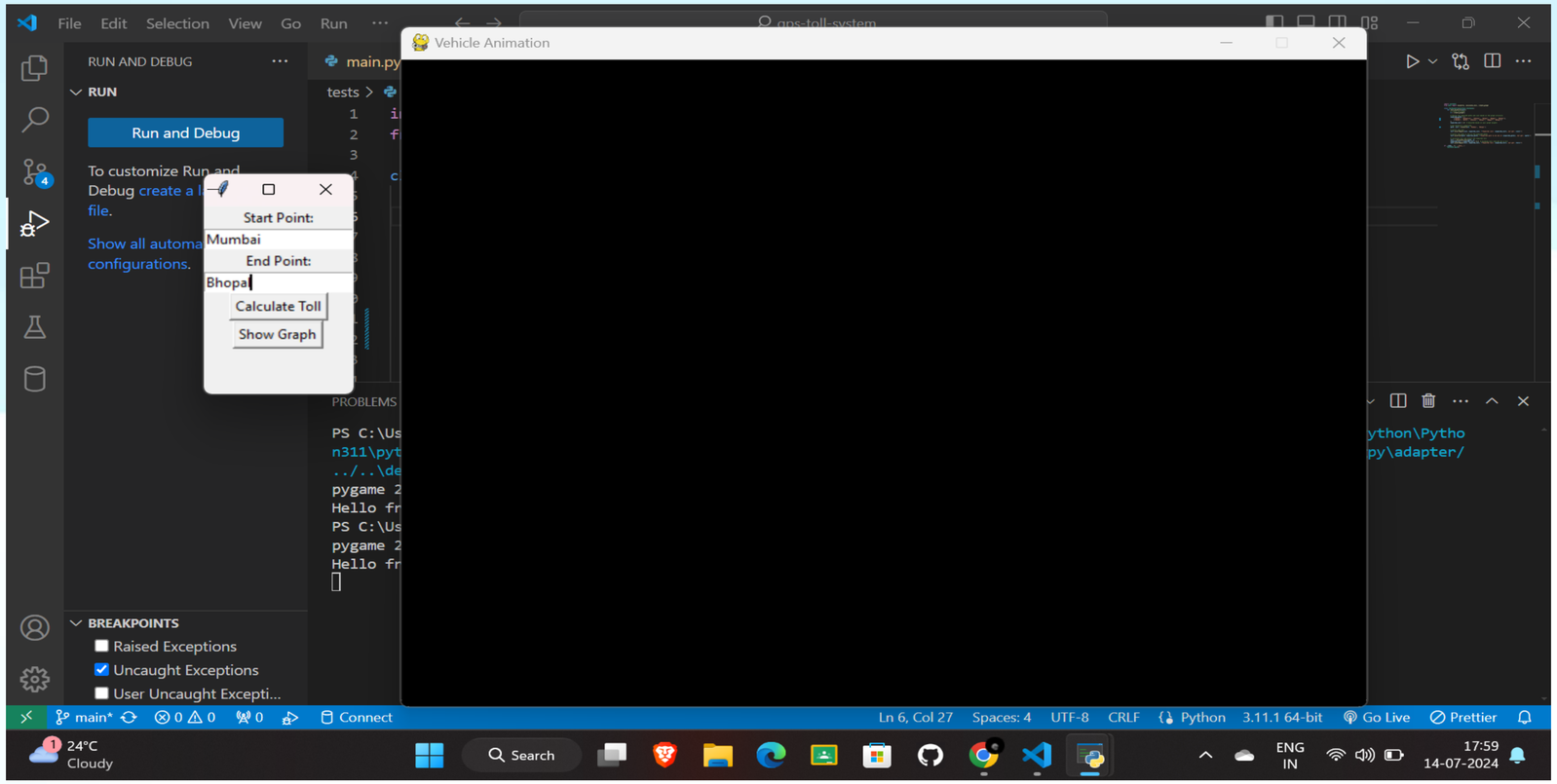
Conclusion

Our GPS-based toll system aims to revolutionize toll collection by ensuring transparency, economic efficiency, and user satisfaction. By accurately calculating tolls based on the distance traveled and providing clear visualizations of the shortest paths, we promote a fairer tolling process. This system not only enhances the travel experience for users but also provides a practical demonstration of graph theory applications, making it a valuable tool for education and analysis. Through this project, we strive to bring about a more transparent, efficient, and user-friendly toll collection system.

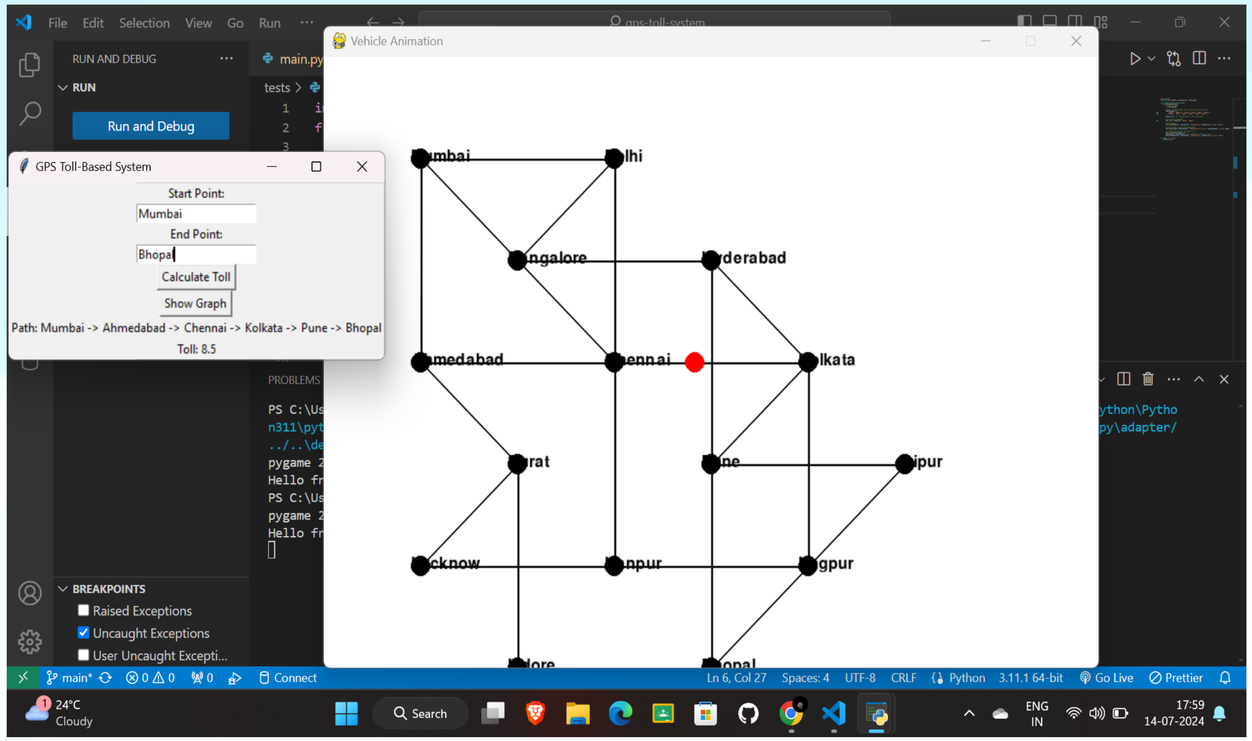
Start Point Selection



End Point Selection



Calculated Toll and Path



Graph Visualization