Project-hand Over

GREEN TRAVEL TRACKER

Group O

Group members: Alen John Varghese

Pooja Prakash

Shafana Nizam

Task:

- Check all requirements for ambiguity, incompleteness, and imprecision.
- For each issue, document how it was resolved in the handover meeting with the previous group. Note that this should be done for most or all of the requirements that have the above issues (ambiguity, incompleteness, and imprecision).
- In case, an issue does not exist, e.g., no imprecision, clearly explain that this is the case.

Solution:

- Requirements
    - Carbon Footprint Calculator: The project aims to develop a carbon footprint calculator. Users input their transportation mode and distance travelled. The system calculates carbon emissions based on UN-approved methods, with input validation ensuring accuracy and promoting informed sustainable choices.
    - Eco-friendly Route Planner: The app facilitates eco-conscious travel by recommending sustainable routes via public transport, car, or biking. Users input starting point and destination, and the app outputs optimal routes, prioritizing emission reduction and resource conservation. Validation mechanisms ensure suggested routes align with user preferences and feasibility.
    - Sustainability Tips: The system provides travel and packing tips, enhancing user experience. Tips cover eco-friendly practices, weather updates, and sustainable travel habits, ensuring users make informed and environmentally conscious choices throughout their journey.

During the handover session, the group B outlined their project requirements for the "Green Travel Tracker." Originally, the requirement was to integrate a carbon footprint calculator, which they specified should be implemented solely within the console interface. The calculator needs an option where users could input their transportation mode and distance to accurately calculate carbon emissions. The transportation modes suggested by Group B encompassed walking, public transport, car, train, and bike. However, walking was excluded due to its

impracticality for distances beyond 3km, acknowledging the potential inconvenience for users traveling longer distances. We recommended the use of sample data, considering the challenge of obtaining real-time public transport data, which is not readily available. They agreed with our suggestion.

In our discussion about the travel route planner, they addressed the need for a system that suggests the most sustainable travel routes, considering options such as public transportation, car, or biking. They requested a system that determines the best mode of transport among these options, while also prioritizing optimal travel routes to minimize emissions and resource consumption. Lastly, they proposed providing users with tips which will enhance user's journey.

Algorithm for `Carbon Emission Calculator`:

1. Start:

   - Print a welcome message.

2. Input Distance:

   - Prompt the user to enter the distance travelled (in kilometres).

   - Validate the input to ensure it's a positive number.

3. Input Mode of Transport:

   - Present the user with options for modes of transport (Car, Bike, Train, Bus).

   - Prompt the user to choose a mode of transport.

   - Validate the input to ensure it's within the range of available options.

4. Calculate Emission:

   - Based on the distance and the chosen mode of transport, calculate the carbon emission using predefined emission factors.

   - Display the calculated carbon emission for the journey.

5. End.

Algorithm for `Eco-friendly Route Planner`:

1. Start:

   - Prompt the user to enter the starting location and destination.

2. Read CSV File:

- Attempt to read the CSV file containing transport route information.

- If the file doesn't exist, print a message and exit.

3. Search for Route:

- Iterate through each line in the CSV file.

- Extract route information such as starting location, destination, and availability of transport modes.

- If a route matching the specified starting location and destination is found, proceed to the next step.

4. Calculate Carbon Emission:

- For the selected route, calculate the carbon emissions for each segment (from starting location to intermediate stop and from intermediate stop to destination) based on available modes of transport and their associated emission factors.

- Determine the least carbon emission mode for each segment.

5. Display Route Information:

- Print the intermediate stop, available modes of transport with carbon emissions for each segment, and the least carbon emission mode for each segment.

- Print the overall least carbon emitted mode of transport for the entire route.

- Print the total carbon emission for the journey.

6. End

These algorithms outline the sequence of steps performed by each class to accomplish their respective tasks.

Future Extension:

- Creation of User-Interface.
- Integration of optimal path by considering total distance.