1. Description

A company wants to reduce the costs of transports and overall carbon footprint of their working force by matchmaking the coworkers on a number of factors such as routes to their destination, schedule, working hours and contacts by ensuring they have similar arrival and departure times. All information will be derived from Microsoft Teams features such as personal calendar, contacts and notification system.

2. Main client

The application's primary client is a firm whose employees commute daily and seeks to reduce both transportation costs and carbon emissions using route matchmaking based on factors such as the employees' schedules and routes,

3. Requirements

Client: The secondary client, i.e the users will download the application, which will require to be already logged in with Microsoft Teams and sync their credentials in order to be able to connect with our carpooling app. Also, not only when creating the account, but everytime when logging in on the app, the user will have to choose between driver or passenger. When setting up their account and selecting the "driver" choice, the app will require to upload a picture of their valid drivers license.

Business: has a database with all the employees' Microsoft Teams email

Once the company completes registration, authorized users (HR or managers) gain access to a dashboard featuring statistics in a user friendly interface, such as CO2 savings estimate, total carpools completed and employee participation rate.

Employee registration: employees sign up using their company email and complete a personal profile with their: company e-mail, password, full name, home address, and driver status indicating whether they would like to drive or not. If said employee does decide to be a driver, they will then provide additional details: car make and model, available seats in their vehicle, license plate number, fuel type and other custom preferences they choose to add

Once an employee completes registration, they gain access to a personalized dashboard with key features to enhance their carpooling experience. They can add multiple pickup or drop-off locations, a real time map displaying their default route and additional stops that updates dynamically, a "carpool buddy list" that includes the names and profiles of people they will share a ride with, and driver details in case they're a passenger. You can contact people using each of these profiles, and the list updates automatically.

5. Technical requirements

For backend storage AWS is used for storing workers' details. The Azure authentication system is employed for workers' connection. Obviously, Microsoft Teams' API is integrated in the functionalities of the app. Each worker needs a Microsoft Teams subscription such that their data can be analysed by the automatic algorithms of route matching. For route optimization and live traffic tracking Google Maps API.

6. Using Scenarios

A new company employs a lot of people, and obviously the cost of the traveling to the workplace has increased, so it needs our carpooling application solution:

Firstly, the manager will create a company account and will be asked to provide the database of Microsoft Teams emails.

The manager could follow which of the employees already made an account.

Every morning and evening, the application runs the automatic algorithm based on each user's data. The schedule is imported from Microsoft Teams and preferences from our application. At the end, each user will be notified which is the route assigned to him. He then can reject the automatic offer and select any other drive to travel with. If no input is provided the systems assumes he s traveling according to the route proposal. At the end of the route statistics are computed, but not before receiving feedback from the drivers.

We used the ChatGPT generation tool to enhance and reformat the written paragraphs, not to come up with the ideas and overall structure of the application. It was not used for the UML use case diagram.