



DailyCommuter

Group #5

Conor Zhang, Nissim Ram, Emmanuel
Owusu-Ampaw, Kamil Paczkowski,
Gideok Min, Boruch Khazanovich

05.01.2025

Overview

Why would you need an app for your commute?

- Stay informed with real-time alerts that may impact your commute.
- Get push notifications when delays may affect your usual departure time.
- Discover alternate routes to help you save time during disruptive
- Know exactly when your next bus or train is arriving — no guess



Frontend Implementation

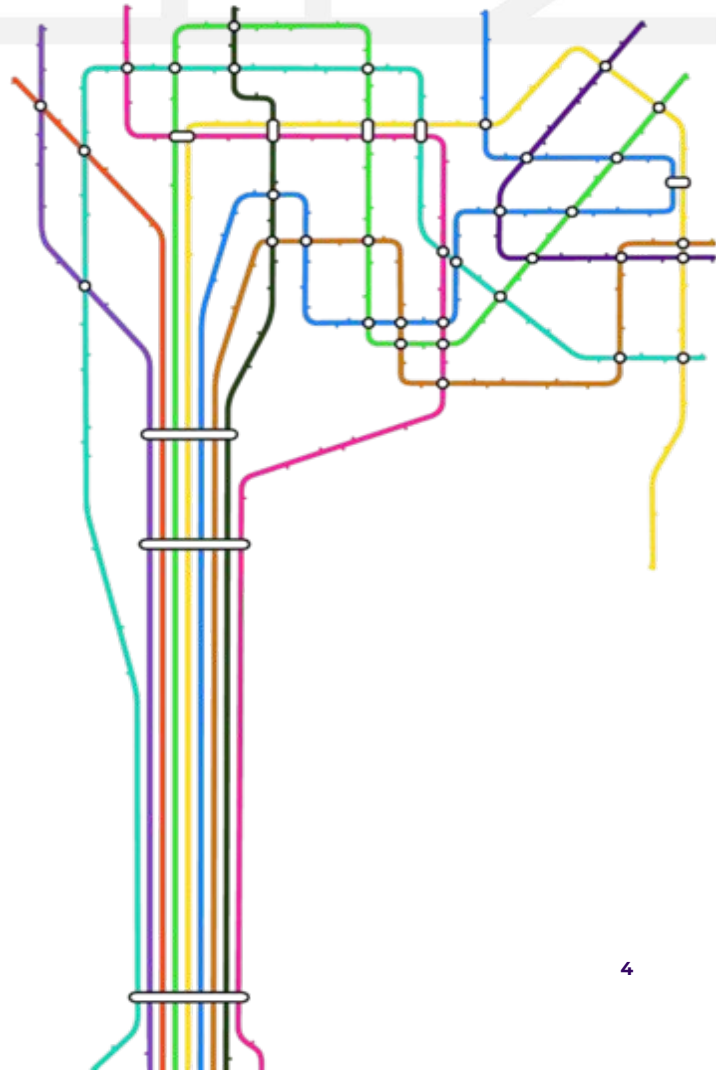
- Responsive design
 - for mobile and desktop
- Visual Inspiration
 - Design will draw from Transit app's clean visual style for intuitive user experience
- Core technologies
 - React
 - OpenStreetMap for mapping
 - Chakra UI component library for non - map components like Forms, Dropdowns, Buttons etc.



Features Implemented

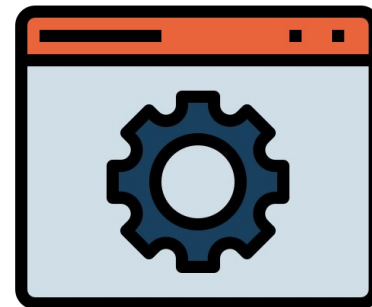
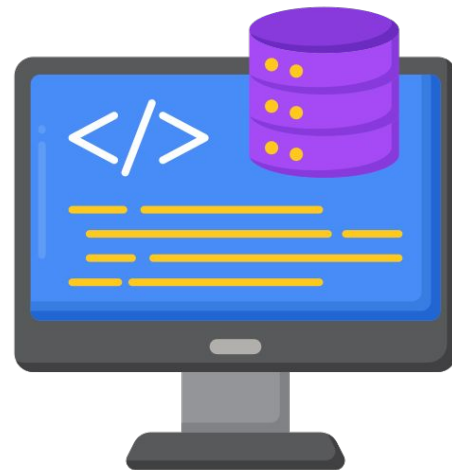
General Implementations

- User setup
- Ability to set and define user routes
 - Routing for users to their destination
 - Tracking of the time it takes each user to get to their destination (estimated)
- Interactive Map:
 - Displays all individual stops



Backend Implementation

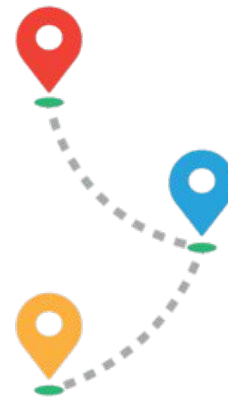
- Backend is implemented using Flask (Python) with Sqlite3.
- As of today, the database is retrieving updates for:
 - All general train updates
 - Service alerts for trains and busses
- The above data is updated on loading the homepage of the app



Features to be Implemented

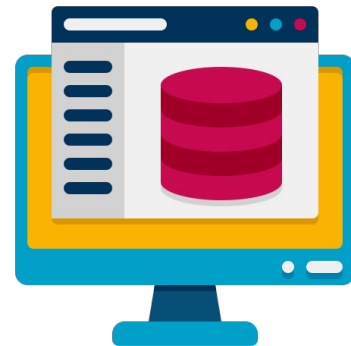
App Specific Implementations

- Route Tracking
 - Provide alerts for delays, service interruptions, and arrival times
- Smart Notifications:
 - App sending reminders when to leave to be on schedule.



Backend Implementation

- Still to be implemented:
 - Retrieve updates for busses.
 - Retrieve alerts for elevator and escalator outages
 - Create filter functions for all searching and filtering needed for the frontend
 - Store user's saved routes and stations
- While the login and authentication has already been implemented using the basic Flask functionality, we still need to set up the database to store the username and passwords



Live Demo

Future Improvements

Scaling & Performance Enhancements

- Database Upgrade
 - Switch to a more scalable database like PostgreSQL to handle concurrent user request efficiently.
- Caching
 - Using catching for frequently accessed transit to improve load times and less API calls
- Switch to a mobile application, as users are on the go and would love to look at this on their phone!



<https://api.mta.info/#/subwayRealTimeFeeds>

<https://github.com/DailyCommuter/DailyCommuter>