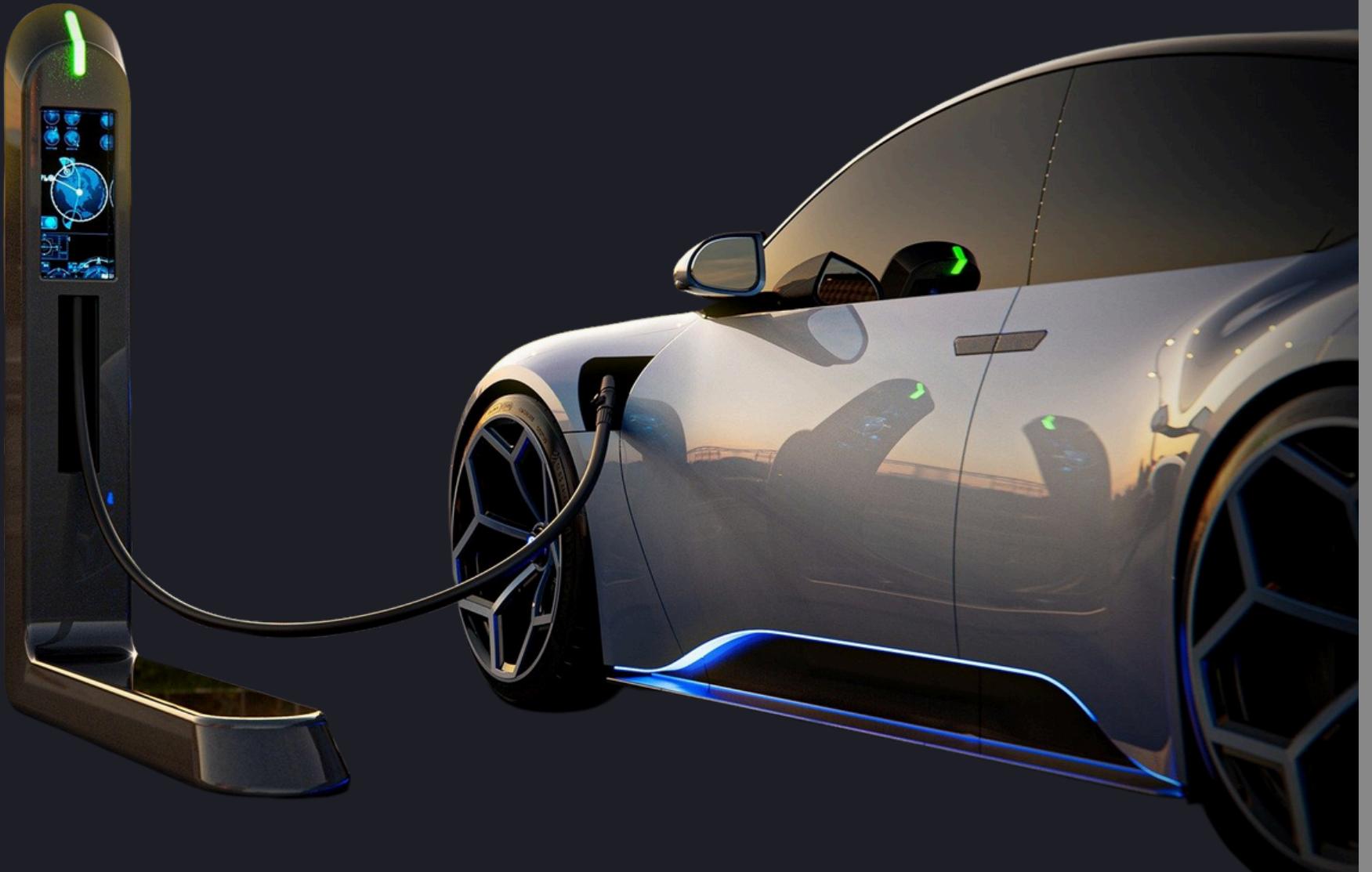




PLUG
IT

Version 2.0



Group 6: Ade Aiho, Heta Hartzell, Mika Laakkonen, Jonne Roponen

SPRINT 2

Table of Contents

- Backend 01
- Frontend 02
- Sprint Ceremony Insights 03
- Team Contributions 04
- Tasks Remaining 05

Backend

The backend uses Express.js for routing and middleware.

MongoDB is used for data storage, and connection is via Mongoose.

Environment variables are managed using dotenv.

Axios is included for making API calls.

Server Structure:

config/ connects to app.js for database setup

controllers/ connect to routes/

routes/ connects to app.js for defining API endpoints

models/ connect to controllers/ and MongoDB

middleware/ connects to app.js

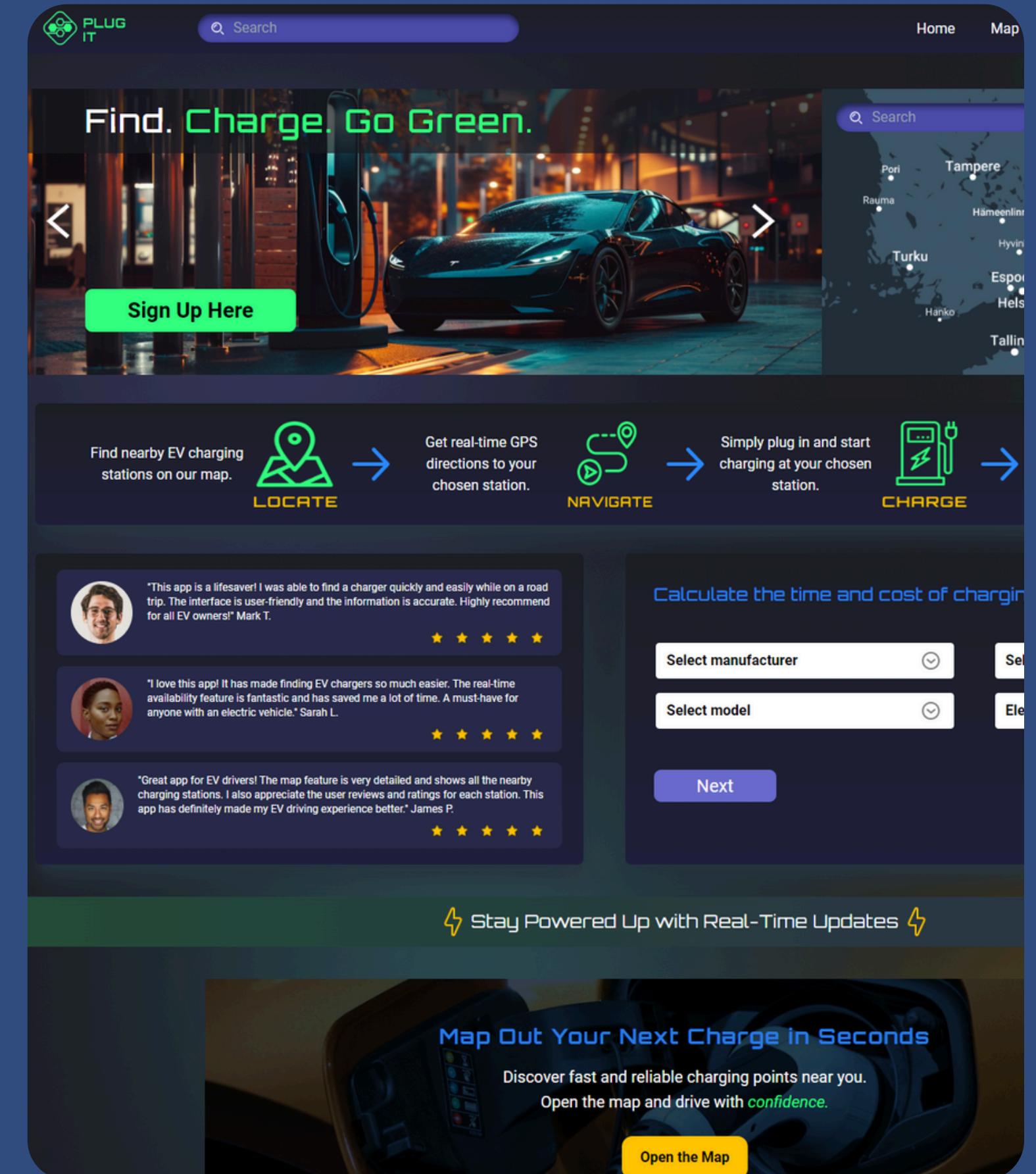
Frontend

Current Features

The client-side is a react application. It uses react-router-dom for navigating between pages, Tailwind CSS for styling, has responsive design and a map for showcasing charging stations. It also has other useful features, such as google login.

Client Structure

- React application
- Components folder with subfolders for each major component
- TailwindCSS and Vite config files
- React-router-dom for navigating between pages
- Layout and Outlet used for displaying components
- Map made with maputnik and stored in MapTiler Cloud



Tasks Remaining

- Connect the front-end React app to the back-end API
- User authentication BE role-based access (admin/user) integration, FE JWT
- Select an EV database to provide necessary data for the time and cost calculator feature
- Implement features for MyPage for more personalized experience
- Refine and create more intuitive user experience
- Develop an information page and ensure all footer links correctly navigate to their respective articles
- Testing BE and FE enhancements



Sprint Ceremonies

Daily Scrums

- Were held daily during work days, bar a few exceptions.
- Improvement in consistency and productivity from last sprint.
- Group found daily scrums valuable and essential in sharing progress and allocating new tasks.
- Trello was used more actively to track progress and allocate tasks.

Sprint Review

What was accomplished:

- Semi-functional, responsive, and styled client
- Groundwork for login and registration
- Routing between application components
- All necessary controllers, routers, models, and middleware for testing with mock-data.
- Data persistence with database incorporation
- Functional map with mock-data

Problems encountered:

- Map proved extremely challenging to implement
- Styling certain components was difficult (unexpected behaviors)
- Refactoring main before merging caused issues

Plan going forward:

- Prepare for sprint 3, further develop skills, finish developing some missing features. The group is well set to finish the project on time and has positive outlooks to implement all wanted functionalities.

Sprint Retrospective

Loved

Group dynamic, daily scrums, learning new skills

Loathed

CSS struggles, map difficulty, git merging

Longed for

Presentation requirement details, better tutorials and resources for map

Learned

React, TailwindCSS, MongoDB, Routing, Middleware, Express.js, Google-Login, MapLibre, among others...



Team Contributions

Ade Aiho

- Writing controllers and routes for the backend
- Testing with Postman
- Studying
- Map component page functionality and styling

Heta Hartzell, PO

- Writing models for the backend
- Generating mock data with LLM
- Testing with Postman
- Integrating database, formatting backend code
- API research and contacting providers
- Login and register components
- Presentation slides

Jonne Roponen, SM

- Multiple responsive react components and their functions
- Applying final styling for cohesive look and adherence to prototype.
- Creating pages and links between them
- Scrum master duties

Mika Laakkonen

- Many responsive react components and their styling
- App routing and navigation
- Overall client structuring
- Google login and user profile data persistence
- Presentation slides



Thank you!

