**Travel Together – Final report - Amit Edward & Gal Ben-Hagay**

* **What made you choose the topic for our project? How did you imagine it? Did it come out the way you imagined it?**

We like to travel, and we want to create an application that connects people.

We think that an application like the one we create can't be found anywhere.

We imagine the application that connects people all over Israel and makes a competition between them.

Our project come out like we imagined him but if we would have liked to add a few more upgrades if we had more time.

* **What made you choose the technology (language/server/database, etc.) that you used? Were there any alternatives you considered?**

We chose React and TypeScript on the client side because of its simplicity and speed of development.

Node.js and Express on the server side work in the same language for the entire application.

For the database, we used MongoDB because its document structure is suitable for route and POI information, and it also has basic support for geospatial queries.

For maps, we used Leaflet because it is free and well-suited for displaying travel routes.

* **How do you divide your work?**

**Gal did:** (Login, register, change password, reset password by mail trap, tracks page, admin panel) front and back, sidebar, map, review page, go to my location, build the tracks schema and first user schema modify the add location on map added modes, option to navigate with Waze to the locations.

**Amit did:**

• added personal details to the register

• ⁠design (partially) the web application

• ⁠design the sidebar

• ⁠wrote the project documentation

• ⁠added a back button to all the pages

• ⁠added the option to show the user’s current location at any time

• ⁠added the option for the user to manually set their location

• ⁠Built the user system with all the personal details - first name, last name, E-mail, phone, birthday, nickname and password

• ⁠Developed the chat option

• ⁠Enable online chat functionality

• ⁠Implemented chat to display both the user’s name and their friend’s name when commenting

* **How did you design it? How did you plan the implementation? What did you do to test it?**

We designed the system as a client–server architecture: client side in React, server side in Express/Node, and database in MongoDB. We designed clear models (users, routes, messages), and pre-configured the API (CRUD for routes, permissions, authentication). We implemented in stages – first registration/login, then route and map management, and later chat and admin management.

* **What did you learn from doing the project? If you started again, would there be anything you would do differently?**

We gained hands-on experience building a complete end-to-end application – integrating frontend and backend, working with maps and geospatial tools, authenticating users, uploading files, and real-time with Socket.io. We also learned how to divide tasks, design an API, and test a system systematically.

If we were to start over, we would try to plan it more clearly and in advance, we would prepare a document that accurately characterizes and describes the milestones and developments we made along the way, we would test it with real external users, and maybe we would think about how the app would behave during load times.

* **Did you use ChatGPT to help with the implementation (or anything else)? If so, what did you use it for, and how useful was it?**

Yes, we used ChatGPT as part of the development. We used it mainly to solve specific problems (e.g. code bug fixes, library usage examples) or to merge the codes we wrote together.

It was very useful for saving time and learning quickly, but we always made sure to check, adapt and change what it suggested to make sure the code fit our needs.

* **What would you do if you have more time? Are there extra features you would like to add, or incomplete features, or anything you would like to redesign?**

If we had more time, we would have added advanced features such as route ratings, comments, and sharing between users, and advanced search with filtering by difficulty/distance levels.

In addition, we would have improved the interface design and user experience, added a mobile app, or provided offline support.