

GPS Trails Summary

By: Alekh Meka, Ayaan Siddiqui, Karan Vishwakarma, Maseeh Khan

GPS Trails is a client-server based web application that is designed to meet the needs of nature enthusiasts searching for parks, national reserves, and hiking trails. GPS Trails allows a user to search trails nearby their location that they can explore, or by using the explore page which uses an interactive map to find places to explore. This application also provides the feature for users to write reviews on trails that they have visited providing information to users that want to potentially explore, such that anyone can read the reviews and find relevant information regarding the specific park or trail. Additionally, the web app also allows users to add new trails that they have discovered or have hiked on previously, so we can grow the content on the web app and stay up-to-date with any new parks/trails.

GPS Trails uses three individual components that work together to create the GPS Trails web app, which consist of a client facing Angular front-end, with a NodeJS(NestJS) backend REST API and a MySQL database for persistent storage of user details, account information, trail/park information and reviews. In addition, GPS Trails also makes sure that all user information is secured by ensuring no passwords are ever saved in plain text, by using BCrypt to hash all passwords that are saved within the database, so in a case where the database is ever breached hackers are unable to steal passwords due to the nature of one-way hashing algorithms. We have also implemented JWT (JSON Web Tokens) to store encrypted user identity on the client side which allows us to check for user authentication with the API without any page reloads and update content appropriately using global state management with NgRx.

Based on the priorities of our consumers, we have developed unit tests for each component in the UI to provide a frustration-free experience for the user. Before each release, we performed regression testing, where we ran all unit tests in our system at the time. Integration tests have also been developed to confirm that our API responds to our front end and our MySQL database populates appropriately with the user's content. Our integration tests were improved throughout the semester as we tailored them to the developmental needs of information storage. We were able to add tests for our API server to ensure that passwords were being hashed properly and that no two passwords are ever the same, this ensures that in an event where our database is breached, hackers will be unable to find out passwords based on most used passwords.

The inspection of the components was performed in a systematic procedure driven process for analyzing the different aspects. The code structure, readability of the documentation, inter-component communication were scrutinized through kick off meetings and various inspection methods which enhanced the usability of the platform.

GPS Trails hopes to foster a new community of individuals that are interested in hiking trails and helping new hikers to start hiking and exploring using information provided by expert/advanced hikers on specific parks and trails. Through this idealized goal, we have prioritized the need for user security and put verification methods in place prior to content being added to our application. For example, each trail that is added by a user through our Add-A-Trail component gets checked by the database administrator for validity and then added to the trails table in the MySQL database. This allows the admins of the site to control incorrect information, or information that was added intentionally but with accidental mistakes.