



Spicy Avocado

Team Reflection

In retrospect, there are many things we would do differently, many things we would improve, and many things that went well with our project.

With regards to things that went well, we did well in our development of our app and we made significant progress in each iteration (PoC, MVP, and final product). We were able to accomplish goals that we set in our project planning document and were able to incorporate some new features that came up as a result of feedback from our PoC and MVP stages. Additionally, we were able to successfully integrate Google Maps into our app, which was one of our biggest challenges since it involves an extensive use of the Google Maps API. We first integrated Google Maps into our app in our first iteration of the app (during the PoC) and continuously made improvements to the app over the course of its development, which ultimately produced good results with regards to custom markers, overlay buttons, and a loading animation.

Moving on to aspects of the project that could have been improved, we think that testing could have been a larger component of our process. When we developed new features such as the Account concept and multi-stop trips, we tested them very quickly, and just made sure they worked consistently before moving on to tackle the next goal. However, this sometimes caused us to miss bigger, more careful bugs, only for these bugs to show up later on when we were working on a different feature. This would cause us to be forced to go back and forth between different parts of the app, which made the process slower and less efficient. We could have definitely improved on this by having a designated section of our development process for thorough testing of the features implemented recently. This way, we could have caught these hidden bugs earlier on and fixed them in time before moving on.

In the future, we could optimize for performance by caching trips so as to reduce the time spent making Google API calls for similar trips. We could also improve the security of our app by hashing passwords instead of having them in plain text and storing API keys in a safe config file instead of storing them as a variable in the backend code. In addition to that, we could hide the password with asterisks as the user types it in the frontend. For

persistence, we could use a database to store information about users and trips instead of storing them in memory.

Overall, we believe development for the project went well and we really enjoyed the experience of working on a real-world type project that potentially be used outside of the class in the future. Thank you for giving us the opportunity to work together on this final project for 6.170!