

## **Difficulties and Ways we Addressed Them**

1. We didn't know how to use React Native and related software. This made it really hard for us to implement features into our app.
  - a. To solve this, we followed tutorials, asked each other for help, and referred to online sources.
2. We didn't have enough time in section and lecture to do everything we needed for the week
  - a. Thus, we used Slack to discuss online or organized meetings outside of class time to meet up.
3. In the beginning, scrum meetings were often not focused, with members far away and not being able to hear what others said.
  - a. We confirmed that this was happening and agreed to always come together in a "blob" formation whenever we held a scrum. This way, everyone could be heard and addressed.
4. We often found a lot of merging conflicts when multiple branches were trying to merge into the testing branch
  - a. We decided that before we merge our branch, we would work with other branches that use similar code and merge them first before sending it to master
  - b. In addition, we made it clear that everyone should pull their starter code from the testing branch.
5. The testing software did not stop running even after the tests finished
  - a. Lucas put in a function that automatically stops the software after a certain amount of time passes. Although this "fails" the check, we can still see the individual tests and see if they passed or not.
6. We once accidentally pushed an individual branch into master, which caused a lot of problems for the app and repository.
  - a. We took away the privilege to push into master and only gave it to Lucas. Now, anything we had to push was pushed to a testing branch instead so it can be checked before being sent to master.
7. Pull Requests were not being pushed very fast (we relied on Lucas to review and push since he was the only one with the privilege)
  - a. We agreed that we should assign another person as a reviewer, and that person should quickly review the PR. This way, Lucas only has to approve it without taking more time to review it
8. Lab instructions were often unclear and it took some time to understand what was expected from us

- a. Before lab, Cameron would create a brief summary of what to do for the lab and post on Slack. In addition, Cameron would message the TA's if he had any questions about the lab instructions
- 9. When we first implemented map markers from Firebase, we found that many of the bathrooms shared identical coordinates (they were above or below one another). Thus, only one bathroom marker appeared in their position and it didn't show the ones underneath it.
  - a. To solve this, we changed the format of the map. Any location that has more than one bathroom will be displayed as one marker, but the information in that marker will display all the bathrooms.
  - b. In addition, we added a new color code of purple to designate locations where there are both male and female bathrooms.
- 10. When we were trying to synchronise the Settings Tab with the Map Tab, we were having a really hard time finding a way to store the data and retrieve it in the app.
  - a. At first, we thought about using Firebase as a way to store and get data. However, this would imply that every single user would need an individual Firebase entry, which wasn't feasible
  - b. In the end, we resolved this issue by using Async, a feature in React Native that could store and retrieve data in the app itself. By using Async, we managed to synchronize the Settings Tab with the Map Tab and turn in our code in time.