What went well

- Database integration with Firestore had some initial road bumps but eventually ended up working very well and provided a lot of the needed functionality.
- The cloud functions especially helped us since they enabled the creation of push notifications to notify users when posts were created by people that they follow.
- The UI came together well and created a visually appealing and functional app.
- Again, we met all of our goals this Sprint. We actually finished more than we thought we would. We spent a total of 75+ hours working this sprint (as evidenced by our commits), and we have created a pretty polished product.
- Adding the manager view wrapped up a lot of the functionality we needed to allow the app to be fully functional.

What didn't go well

- Although we do have a view in place for the rec center to create an account and manage the UA part of the app, we were never able to fully integrate with the rec center as planned due to the COVID-19 outbreak.
- A lot of time was spent writing new methods instead of refactoring old code; this meant bugs in old code would be hard to fix due to the new methods' dependence on them.
- We had some trouble unit testing due to the application's dependence on Firestore and future methods.

What could be improved

- Better user preferences load from a file local to the device or other. This could help provide a better user experience for both developers and users.
- Loading from a cache could also help in reducing the amount of loading from the back end that is being done. This would involve storing a smaller set of the data base in the local file system either following people or the posts that will happen in the next N days.
- Refactor the code to get a maintainable code structure and clear places where functions should end up. This could also help to decouple the logic of the program from the form of the app.
- There are some unused functions and dependencies that could be cleaned up.

Challenges

- Some of our original widgets were stateless; updating them to stateful was very complex and was like taking 4 steps back for every step forward we took.
- Some of the libraries we were using failed to work with some of the other libraries we were using. They would produce undefined behavior, this caused some challenges in troubleshooting the code and figuring out what was causing the issues we were facing.
- Different code styles make understanding what the other group members are trying to do confusing. Some of the code is written with an OO paradigm and

some of it is written with an imperative paradigm. This made coupling the code harder due to the different paradigms that were used.