

SkillCourt Interactive Soccer Technology

Students: Andy Robles

Mentor: Gummi Traustason

Instructor/Faculty: Masoud Sadjadi, Florida International University

PROBLEM

The SkillCourt Interactive soccer Technology application and website were out of date and needed to be fully updated and upgraded. On the application side: all packages and SDKs that SkillCourt used (Dart, Android Tools, Gradle and flutter) needed to be upgraded to their newest versions. Class paths had to be rerouted, migrated, and updated as well. On the website side: the readme file for installing the S.I.C.T, database, database ids, Docker Compose files, NesUS dependencies, packages from Prisma and NodeJS, and database references needed to be updated

SYSTEM DESIGN

Flutter is being used to implement the existing system. Pages pertaining to interacting with the pads were made in the preceding semester in

Because Flutter does not enable native multi-threading, Swift and Kotlin were added as alternatives. We were able to continue utilizing Flutter for the whole application, however, thanks to the recent release of Isolates from Flutter.

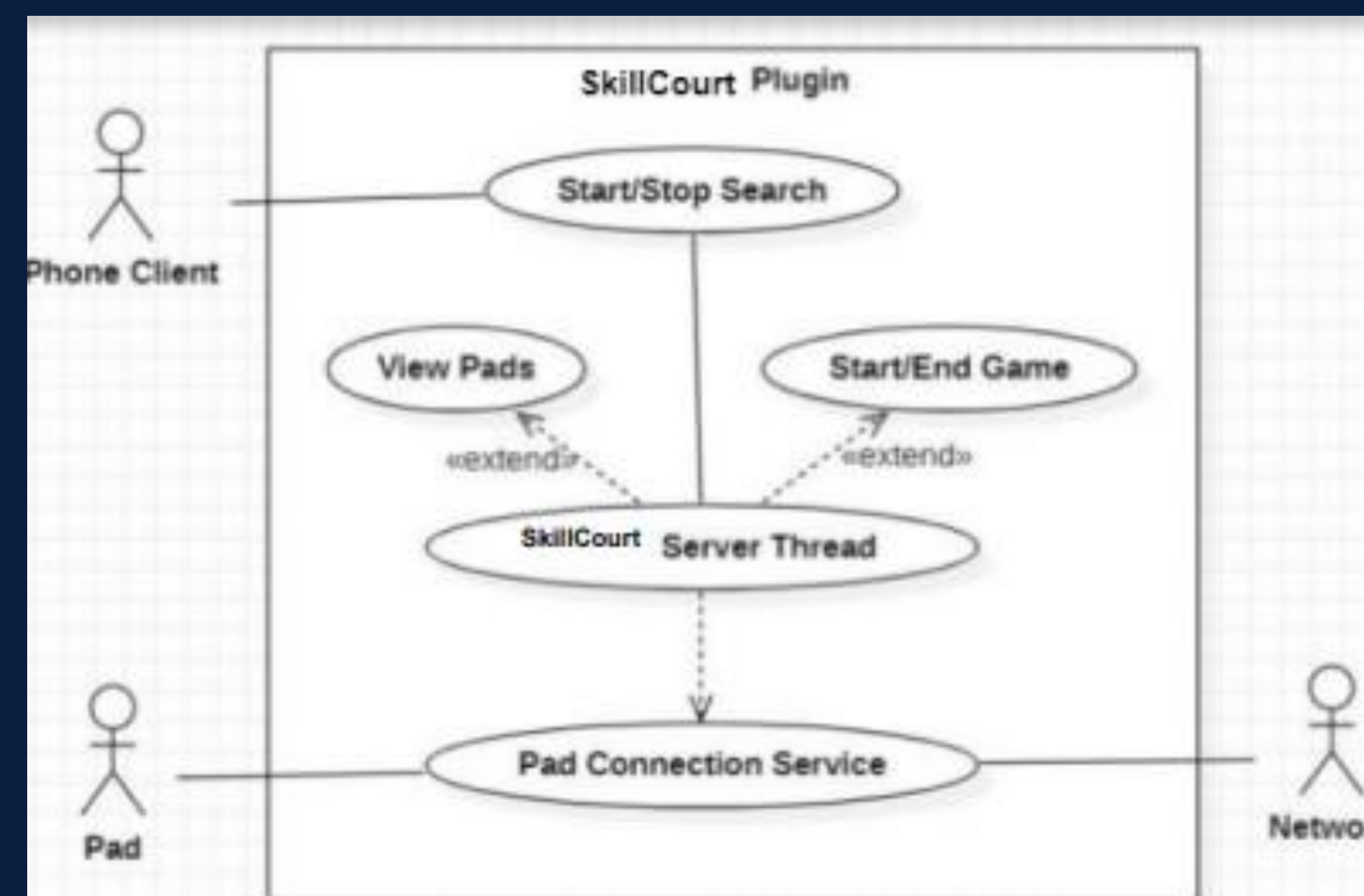
VERIFICATION

- Worked on the firmware for the pads
- Completed the player statistics page and implemented ways to delete player game data/have all previous sessions displayed for the player
- Setup the global navigation bar that's used throughout the entire application

CURRENT SYSTEM

The SkillCourt App and website seems to be fully functional on our local systems; although, we do not have the pad to properly test the application. The application is still in a barebones like state and needs a couple improvements in its design layout as well as a couple more features

OBJECT DESIGN



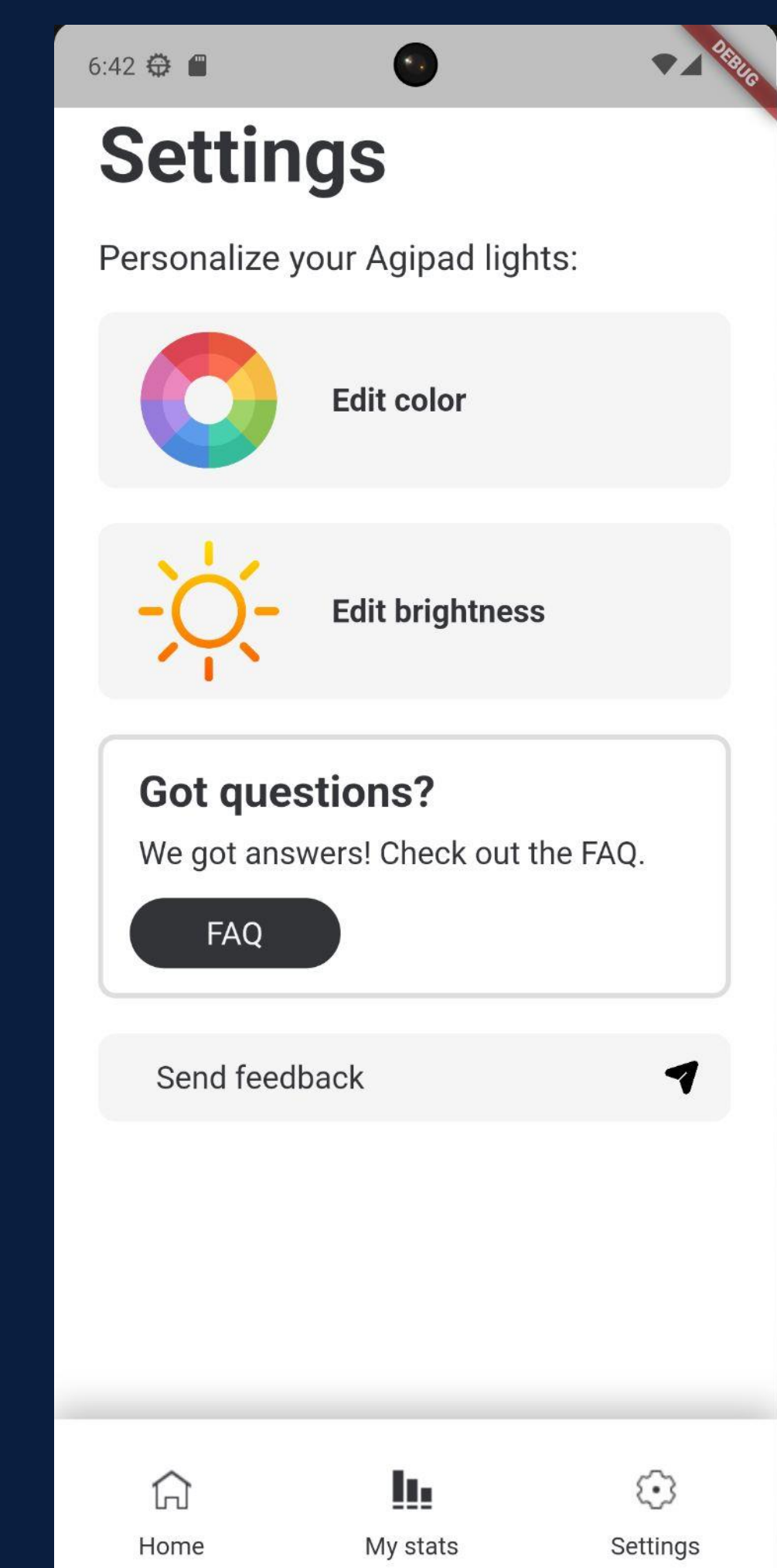
SUMMARY

I implemented the UI design for the settings page. I also initially created a test program that used a Python script to establish connections with simulated pads, maintain those connections until the simulated pads are shut down or unplugged, and regularly loop over fresh messages sent from the simulated pads to the Dart program. Once this was done, we added the new Isolate "threading" technology to the "agineek-app" program. The key objective was for the pads to establish and maintain a connection with the program. The pads were ultimately able to connect without disconnecting after numerous conversations. I eventually added error messages and tips to the already-existing forms. The application's error messages for inaccurate data and correction tips may need some improvement. Due to time constraints, I was unable to do more research. I was also going to make sure that some of the other registration forms were filled out.

REQUIREMENTS

Application	Website
Java	Github
Github	Node
Git	Npm
Git-flow-avh	PNPM
Flutter SDK	Docker Desktop
A phone emulator	Vim
Android tool chain	VSCode
PythonJ	
VSCode	

IMPLEMENTATION



REFERENCES



ACKNOWLEDGEMENT

The material presented in this poster is based upon the work supported by XXXXX (name of the project sponsor: federal, state or local government, or corporate partners, where applicable). We/I thank XXXX for his/her/their assistance and mentorship that I/we received throughout the senior design project.