



SkillCourt Interactive Soccer Technology

Angelo Boggio, Cruz Villafranca, Marcus Munos, Andy Robles

Florida International University

Summer 2023 Capstone 2 Project

SkillCourt Interactive Soccer Technology

Students: Angelo Boggio

Mentor: Gummi Traustason

Instructor/Faculty: Dr. Masoud Sadjadi

SkillCourt
PLAY ON PLAYER

App Summary

SkillCourt is a mobile and web application where players kick soccer balls at a physical LED panel. Their accuracy is recorded by a phone application

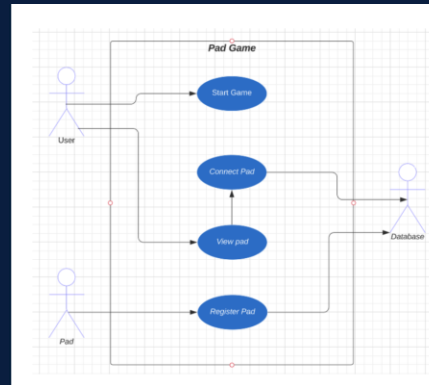
Problem

The biggest problem I faced was working on the firmware code for this system as it was having a lot of dependencies issues and hardware issues and then just moving the player statistics data around from one page of the application to the player statistics page

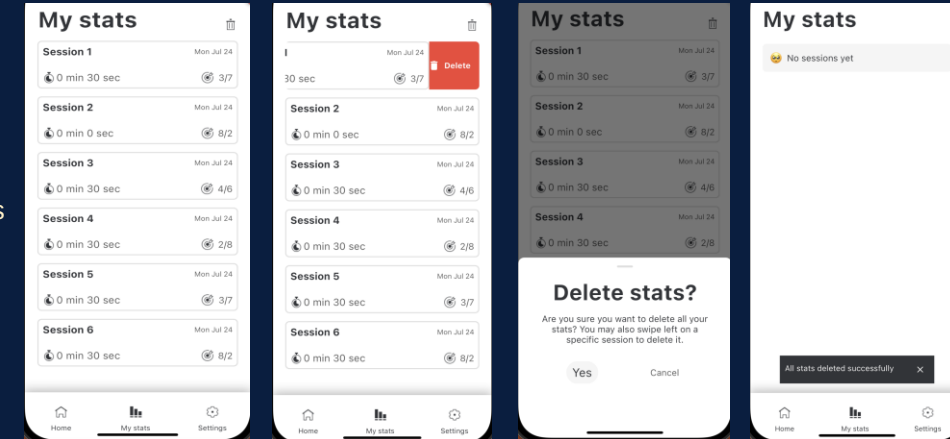
Verification

- 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

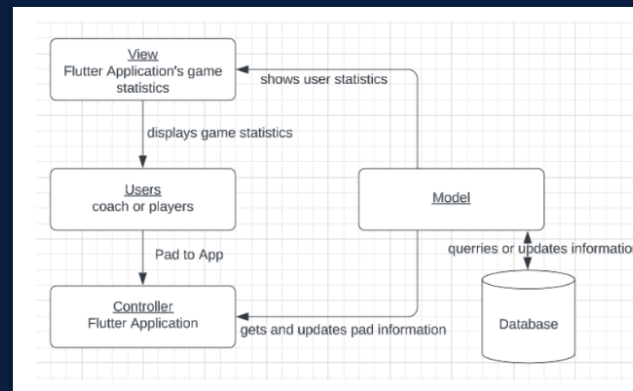


Screens



Screenshots show the features that were implemented into the player statistics page such as swiping to delete individual sessions, a scrollable session list, deleting all player sessions simultaneously, and having a full functional bottom navigation bar

System Design



ACKNOWLEDGEMENT

The material presented in the poster is based on the work supported by FLORIDA INTERNATIONAL UNIVERSITY Gummi Traustason. Thank you to my Capstone II members.

FIU

**Engineering
& Computing**



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The SkillCourt Interactive soccer Technology application and website were out of date and needed to be fully updated and upgraded.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.

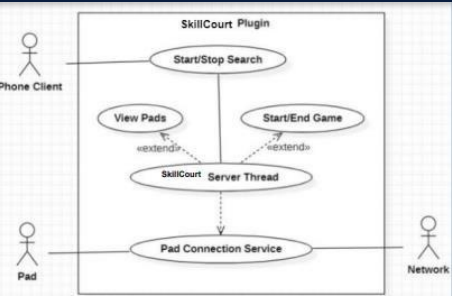
VERIFICATION

- 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.

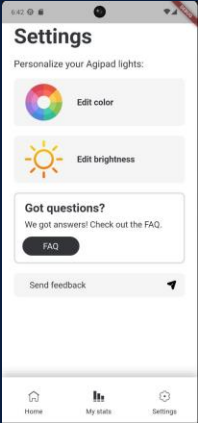
REFERENCES



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



ATTENTION: Subheadings shown above is a recommended structure for a research/scientific poster. Main elements of a poster are Introduction, Research and Conclusion. Researchers can customize the section headers based on their projects' needs. REMOVE THIS TEXT ONCE YOU START WORKING ON YOUR POSTER

ACKNOWLEDGEMENT

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

Florida International University

Summer 2023 Capstone 2 Project

Skillcourt

Students: Cruz Villafranca

Mentor: Gummi Traustason

Instructor/Faculty: Dr. Masoud Sadjadi, Florida International University

PROBLEM

Currently the application is still in an early state, the hardware does not properly connect to the software, this is an issue we tried to fix, but without access to the actual hardware it was not possible.

SYSTEM DESIGN

Four our current system, the app works by communicating with the database to get the games the user has played, the app will then let you connect the pad hardware and play a new game.

VERIFICATION

The way we are verifying this project is by determining that the project's UI looks good, which we will demonstrate in the summary

CURRENT SYSTEM



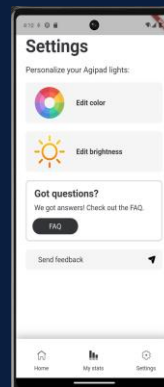
The old system design contained a lot of uncomfortable buttons and hard to read text, along with a tacky color palette.

OBJECT DESIGN

The user will have multiple different games assigned to them, each game will contain other properties such as game type, the points and the user accuracy metrics.

SUMMARY

This is a quick preview of one of the pages that I worked on in the app, as you can see the UI now looks much cleaner than the previous system. We have enlarged buttons for easier user interaction and a much cleaner color palette.



REQUIREMENTS

The goal of skill court is to provide soccer players and enthusiasts with the ability to train easily anywhere. So, our requirements are to implement hardware and fix the UI for the current application.

IMPLEMENTATION

It took some time to get this application running, we spent a very long time fixing package issues and UI bugs we found. However, this allowed us to get a very good understanding of how the application worked and how its data flowed.

REFERENCES

Code documentation we used:

<https://docs.flutter.dev/get-started/install>

ACKNOWLEDGEMENT

The material presented in this poster is based upon the work supported by Gummi Traustason. We thank Gummi Traustason for his/her/their assistance and mentorship that I/we received throughout the senior design project.

SkillCourt

Students: Marcus Munoz

Product Owner: Gummi Traustason

Mentor: Joseph Quinn

Instructor/Faculty: Dr. Masoud Sadjadi, Florida International University

PROBLEM

SkillCourt is currently being developed to address the strict and challenging training regiment that soccer players endure. SkillCourt allows players to train anywhere by themselves without any assistance.

However, we have some current problems:

- There are problems with creating a raspberry pi emulator to simulate the connection for the pads.
- There is a library required either needs to be fully researched or created to connect pads to flutter application

SYSTEM DESIGN

The System is designed to use a raspberry pi to connect the LED pads to the application. Using a custom-built firmware application to connect the inputs to the mobile application. The mobile application is being developed in Flutter. This will then be connected to a database to store user profiles and user statistics.

CURRENT SYSTEM

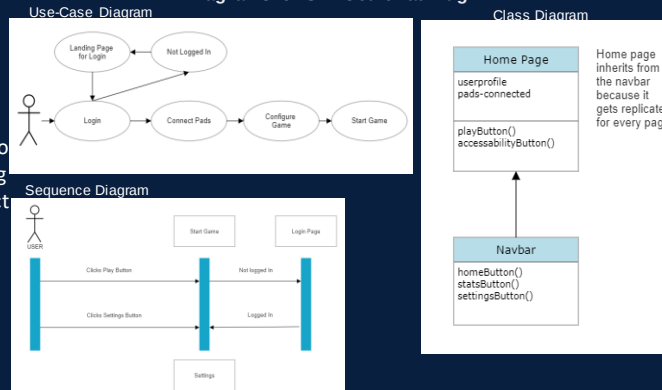
The current iteration of the project is being implemented with the use of Flutter, which in turn uses Dart as it's programming language. Flutter allows the developer to write in Dart and then it is transpiled into both Swift and Kotlin depending on the device you are targeting.

REQUIREMENTS

In order to work on this project, the proper environment is needed. The following is necessary:

- Git
- Android emulator
- FlutterSDK along with Dart
- Python 3 (for the pad emulator)

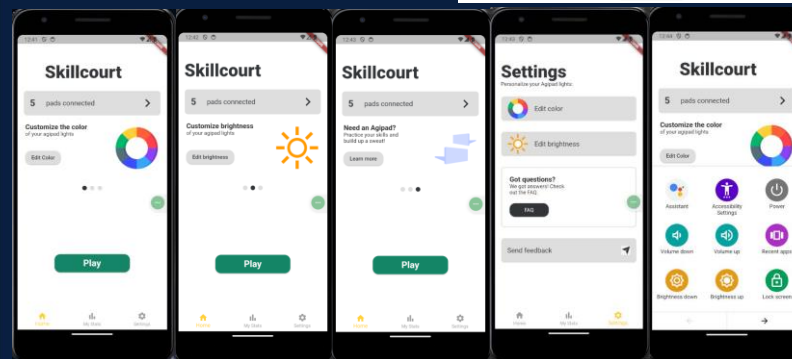
OBJECT DESIGN Diagrams for SkillCourt Pad Pugin



IMPLEMENTATION



SCREENSHOTS



SUMMARY

- Created new Home Page
- Created new Page for Navbar
- Created Navigation to other Pages
- Added Accessibility options

FUTURE

Finish other pages:

- Saves
- Play configuration
- Game Creation
- Play timer screen

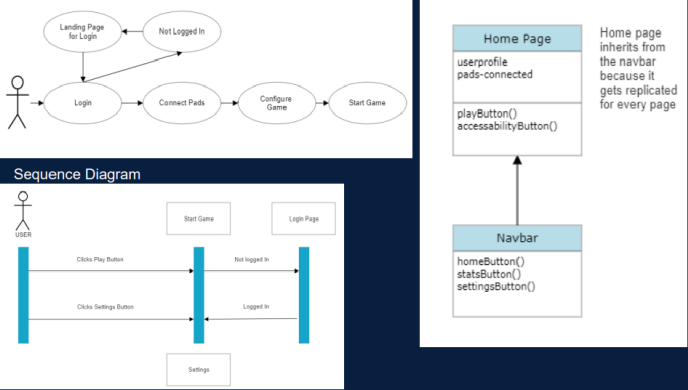
ACKNOWLEDGEMENT

The material presented in this poster is based upon the work supported by Gummi Traustason. Thanks to all my Capstone 2 members and Franz who provided the re-design for the application

PROBLEM

- ✦ SkillCourt is an app that was created to document the extensive and strict training that soccer players go through. This app allows soccer players to go through their
- seek the assistance of anyone other than themselves
- ✦ Outdated designs and limited accessibility
- ✦ emulator to stimulate connection between the pads
- ✦ to create and connect pads to flutter the application

OBJECT DESIGN

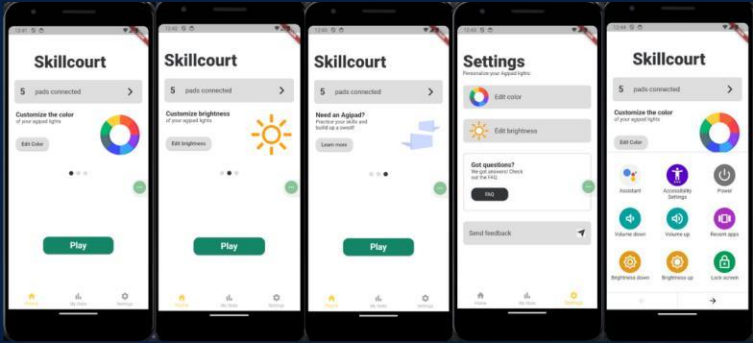


ACKNOWLEDGEMENT

The material presented in this poster is based upon the work supported by Gummi Traustasun. We'll thank Franz for his/her/their assistance and mentorship that I/we received throughout the senior design project.

CURRENT SYSTEM

- ✦ The current project is implemented it is programming language. Flutter allows the developer to write code in Dart and then it is translated into Swift and Kotlin



Implementation



SUMMARY

- ✦ Accomplished
 - ✦ Navigation bar creation
 - ✦ Options for accessibility
- ✦ Future Projects
 - ✦ Customize the pad creating a game
 - ✦ Lessening whitespace on home pages

SYSTEM DESIGN

- ✦ The System is designed to use a raspberry pi to connect the LED pads to the application is used to connect the inputs to the mobile application. The mobile app was developed in Flutter which

profiles and user statistics.

REQUIREMENTS

- ✦ Git
- ✦ Flutter SDK along with Dart
- ✦ Java JRE version 1.8 or greater the pad emulator)