



EGCI340: Software Design

Drop In Application

6580969	Veerapat Leepiboonsawat
6581147	Trinnaya Damrongpatharawat
6680552	Akanee Suthiudom
6681224	Nathan Tanaboriboon
6780093	Thanasanan Leardkasemruthai
6780704	Jakapat Huayrahan

Submitted to Asst. Prof. Dr. Lalita Narupiyakul
Mahidol University International College
Trimester 1/2025-2026

Drop in

1.1 Problem definition and objective

Problem definition

- Organizing a team for playing sports is often difficult if we have a friend who always wants to play sports with us, but cancels at the last minute, so we don't have enough teammates to play against other teams. In addition, for university students, there is no application to make a sports complex reservation, or check if university courts are free unless we walk to the location in person. We think that many people may face the same issue, so we want to develop an application called "Drop In" which means finding a buddy to play sports with.

Objectives

- Our main objective is to solve the shortage of players caused by teammates or friends canceling the plan. After a lot of observations on the students in the university, we found out that many of them have a friend who always cancels on plans. From these problems, we want to build an application which allows users to form a team, find a friend, or join the team for the sport they want to play. Moreover, as a supporting feature, we want the app to allow students to reserve specific sports courts at the university directly through the app.

1.2 Feasibility study document and Project Plan

For this project, we started from the annoying problems of failing appointments with our friends.

It wasted so much of our valuable time and, most importantly, it ruined our vibe for the day.

Therefore, we knew we had to come up with a remedy to this rather niche but important everyday problem. We ended up designing a whole new system.

Starting from the beginning, we found that we all have a common interest in sports, but we often faced two distinct hurdles: sometimes we had a team but no court, some other times we had a court but no team. We came to an agreement that teaming up for sports within the university is quite difficult. So our model towards the problem is described as “The easier it is to book, the higher the chance of people not showing up.”

We initialized the project scope to be within the university itself. This is because the first priority was the safety of the users. The app is meant for **students only**, utilizing the existing Mahidol student ID card credentials paired with the campus database for verification. Allowing strangers to use the app would impose a safety risk, so strict exclusivity is required.

For our functional goals, we designed the application to serve two main groups: those who want to find people to play with, and those who just want to reserve a room.

- **Create Game Mode:** Users can host a match by selecting the sports type, date, time, location, and max number of people. We added a layer of control here: “Public” allows any student to join instantly, while “Private” allows strangers to see the game but requires them to send a request to the host first.
- **Reservation Mode:** Since MU lacks a dedicated app for sport complex reservations, our app serves this purpose. A user can simply book a slot without creating a team.
- **Join Mode & Timetable:** Users can search for games by host name or location. Once joined, the “Timetable” feature allows them to track incoming matches so they can plan their schedule.

The Attendance & Penalty System To solve the problem of users flaking after reserving, we implemented a unique feature inspired by the **Anywheel QR system**. This is a critical part of our Phase 1 development. To prove attendance, a user must scan a QR code at the facility to "join," and scan again to "leave." If a user enters a match but fails to scan, the system flags them as missing. Users have 3 chances to misbehave; if they fail to show up or scan three times, penalties are applied. This ensures that the community remains reliable and efficient.

Milestones and Development. For our milestones, we are aiming to follow the incremental model.

Increment 1: Operational Foundation

The primary deliverable is the Sport Complex Reservation System, which allows users to view availability and book facilities. Simultaneously, the Game/Match Creation and Join System is implemented, enabling users to host lobbies or join existing games. To close the loop between the app and the physical location, a Scan System (QR-based) is developed to validate bookings and grant entry at the venue. This lets the app at least function.

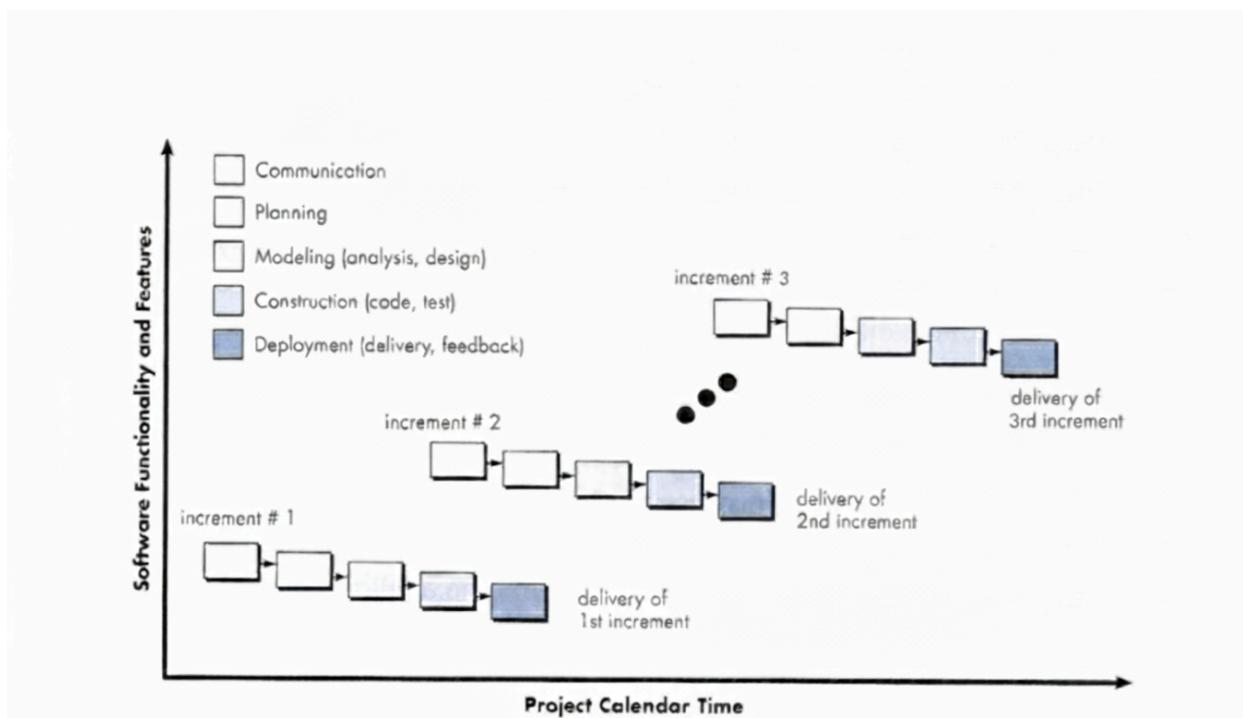
Increment 2: Discovery and Management

The second phase focuses on enhancing user experience and facilitating easier interaction with the platform. It introduces a Player Rating System, which is designed to be optional to reduce user pressure, while still allowing for skill or sportsmanship feedback. The system is also upgraded with a dashboard for History and Incoming Match Viewing, helping users track their schedule. Furthermore, the Search System is significantly improved during this phase to include Location Priority, ensuring that users are presented with matches and facilities closest to them

first. Not only that, we have implemented a search function. Instead of endless scrolling through undesired matches, they can now search up for a specific match from the game name.

Increment 3: Analytics and Visualization

The final phase aims to increase user retention and polish the visual interface. A Stats System is introduced to track and display user performance data, adding a layer of gamification to the experience. Additionally, Advanced Graphic Functions are integrated into the location system; this upgrades standard list views into more interactive visual elements, as rich map interfaces with detailed facility locations information, providing a premium look and feel to the application.



1.3 Software requirements specification (SRS) document

User Requirements

Using Application Interface

1. Log In/ Sign Up

The user must be validated as MU students to have an account on the application.

Users should be able to sign up given they have not had an account, and log in when they already have one.

2. Personal Statistics Management

The user should be able to look at his/her statistics of playing sports on the page, including time played for each sport shown in a bar graph, and there will be top 3 of sports the user had spent time playing the most, as well as top 3 facilities where he/she had been to most frequently. This section will be private to the user only. The profile must also display the user's reliability score, which is dynamically calculated based on their attendance history and the current state of their 3 penalty points/chances. This section can be viewed by other users.

3. Creating Game and Reserving Facility

The user must be able to create a new game by providing all essential details. This requires mandatory fields for a game title (e.g., "Let's play together"), the specific sport category (e.g., Basketball), and a maximum player number. He/she must define the game's schedule using date and start/end time selectors. The user must have selected the sport category before an attempt to set the location as they will be directed to the map page with markers only on chosen-sport-category facilities. There will be available facilities and rooms for a user to choose from. In addition, The host (user who created the game) must be able to set the game to private, requiring joining users to send a request for approval, and to public, allowing any student to join instantly until the maximum number of players is reached. Finally, when the user confirms the booking, the game will be posted, the court will be reserved, and the user will receive a successful creation notification.

4. Joining Game

The user will be able to view the list of nearby both going-on, and upcoming games that they can choose from. He/she must be able to search games by game title, sport category, location, or host name, then the list of filtered games will be displayed with key details (date, time, place, and spots remaining). User can then select the game and click to join the match, which, upon successful joining the details of the game will then be automatically added to the user's schedule.

5. QR Scan

Users should be able to scan a facility's dedicated QR code for both arrival (check-in) and departure (check-out) to confirm their attendance for the game appointment. Not scanning when entering and/or leaving the court results in a penalty point/chance deduction.

6. Feedback System

After joining the game, there will be a pop-up page for users to give out feedback to the host or any other player, which is optional. Users can leave comments about them by choosing whether to be anonymous or not.

7. Timetable

The user can see the incoming games (by default), when selecting the specific game, the details of the game with location will be popped up. The map will be displayed with a marker to show the location of the sports complex if the user clicks to see the location on the previous page. On the same detailed-game page, user is able to cancel the game. There is an option to view history games user had joined in the past with the same details, the location map, and cancellation function as the incoming game part.

System Requirements

Functional Requirements

1. Authentication and Profile Management

Actors Involved: Users, MU Database

FR-1.1 The system must validate the user's MU student ID upon sign-up and login, restricting access only to authenticated students.

FR-1.2 Users must be able to view their personal statistics of most preferred sport type, and most frequently visit sport facilities.

FR-1.3 The user's reliability score must be viewable by other logged-in students.

2. Game Creation and Facility Reservation

Actors Involved: Users, Sport Facility Database, Sport Facility Administrators

FR-2.1 The user (host) must be able to input the game's details: title, sport, location, date/time, max players, and create the game. The system shall post the game details on the application page for other users to join.

FR-2.2 The system must display real-time map markers on available facilities according to the host's game creation information (i.e., date and time, and sport category).

FR-2.3 The system shall reserve a place on the sport court database in accordance with the host's game creation details.

FR-2.4 The host must be able to set the game to private, requiring joining users to send a request for approval.

FR 2.5 The host must be able to set the game to public, allowing any student to join instantly until the maximum limit of player is reached.

3. Game Discovery and Joining

Actors Involved: Users

FR-3.1 The system shall retrieve the user's real-time location, and display games in the nearby user's location.

FR-3.2 The system shall find and display a list of games according to user search filters by game title, sport type, location, and host name.

FR-3.3 The system must send a join request to the host for private games, and only register the player upon host acceptance.

FR-3.4 The system must allow instant joining of public games with immediate confirmation and real-time update of remaining slots.

4. QR System

Actors Involved: Users

FR-4.1 The system shall recognise the sport facility's dedicated QR that user scan upon their arrival and leaving, and take record of their status.

FR-4.2 The system must maintain a user's 3 penalty points/chances. Failure to check-in or check-out for a joined game results in a point deduction as the user being flagged as "missing", or "overstaying".

5. Post-Game and Rating

FR-5.1 The system must let all participants of the game rate or comment on the other players 15-30 minutes after the game's scheduled end time.

FR-5.2 The system must update the user's personal timetable with upcoming events and their statistics section with completed game history.

Non-Functional Requirements

1. Performance Requirements

NFR-1.1 Maximum time for the game List or personal statistics page (standard text/image pages) to load fully: 5 seconds.

NFR-1.2 Average time for joining game and QR interactions to process: 1 second.

NFR-1.3 Real-time court availability and player count updates must occur within 100 milliseconds of a transaction (booking, joining, cancelling).

2. Safety Requirements

NFR-2.1 The system must cooperate with facility administrators to prevent new bookings and notify users instantly if the court goes “unsafe” or “closed”.

3. Security Requirements

NFR-3.1 All personal user data and authentication credentials must be protected.

NFR-3.2 The system must strictly follow authorization checks (e.g., only the host can accept join requests for a private game).

NFR-3.3 The user’s current location must not be shared or accessible by the other users, and not be used for other purposes.

4. Software Qualities

NFR-4.1 The database architecture must be capable of supporting up to 500 concurrent active users and handle peak load without degradation in performance.

NFR-4.2 The system must be functional, well-calculated on real-time player counts, court availability, adding users to the game, and update them correctly across all users' application interfaces.

System Constraints

1. Hardware Requirements

CON-1.1 Scanning QR code requires the device's camera.

CON-1.2 The application can be run on smartphones with IOS version 15 or above, and version 12 or above for Android operating system.

2. Software Requirements

CON-2.1 The application must utilize a reliable, open-source library for fast and accurate QR code decoding across different lighting conditions and camera qualities.

Domain Requirements

1. The application must integrate with the MU Sports Complex's master schedule data for initial court availability and facility names, which will be the authoritative source for location data.
2. The system must strictly follow the university's booking policies regarding maximum reservation hours, advance booking limits, players limit, or any other regulations.
3. All activities of users done on application must comply with MU code of conduct, with a process for reporting and investigating inappropriate user behavior.

For 1.3 section, please refer to the use case diagram down below for further clarification of certain parts.

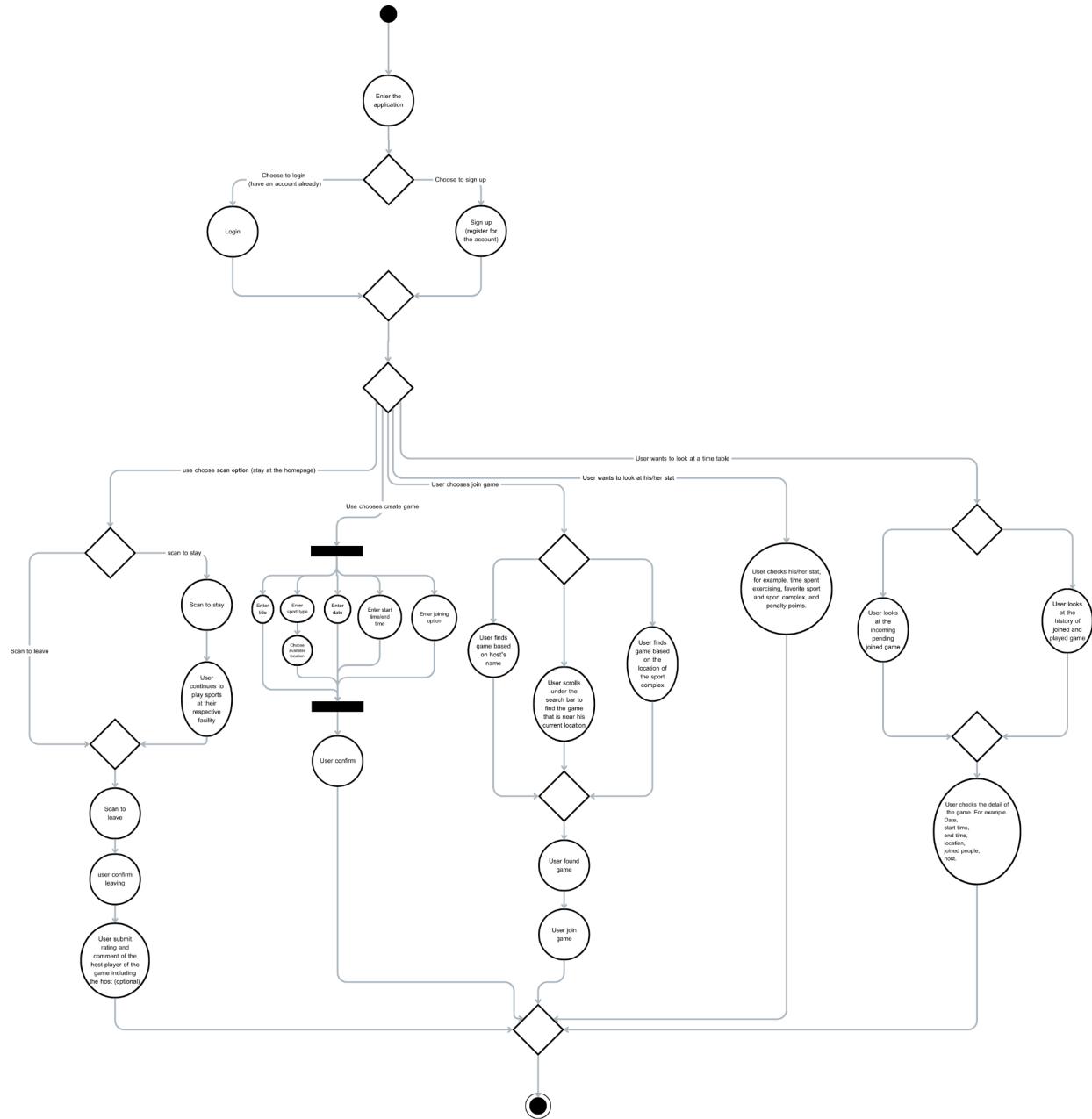
1.4 Design specification document

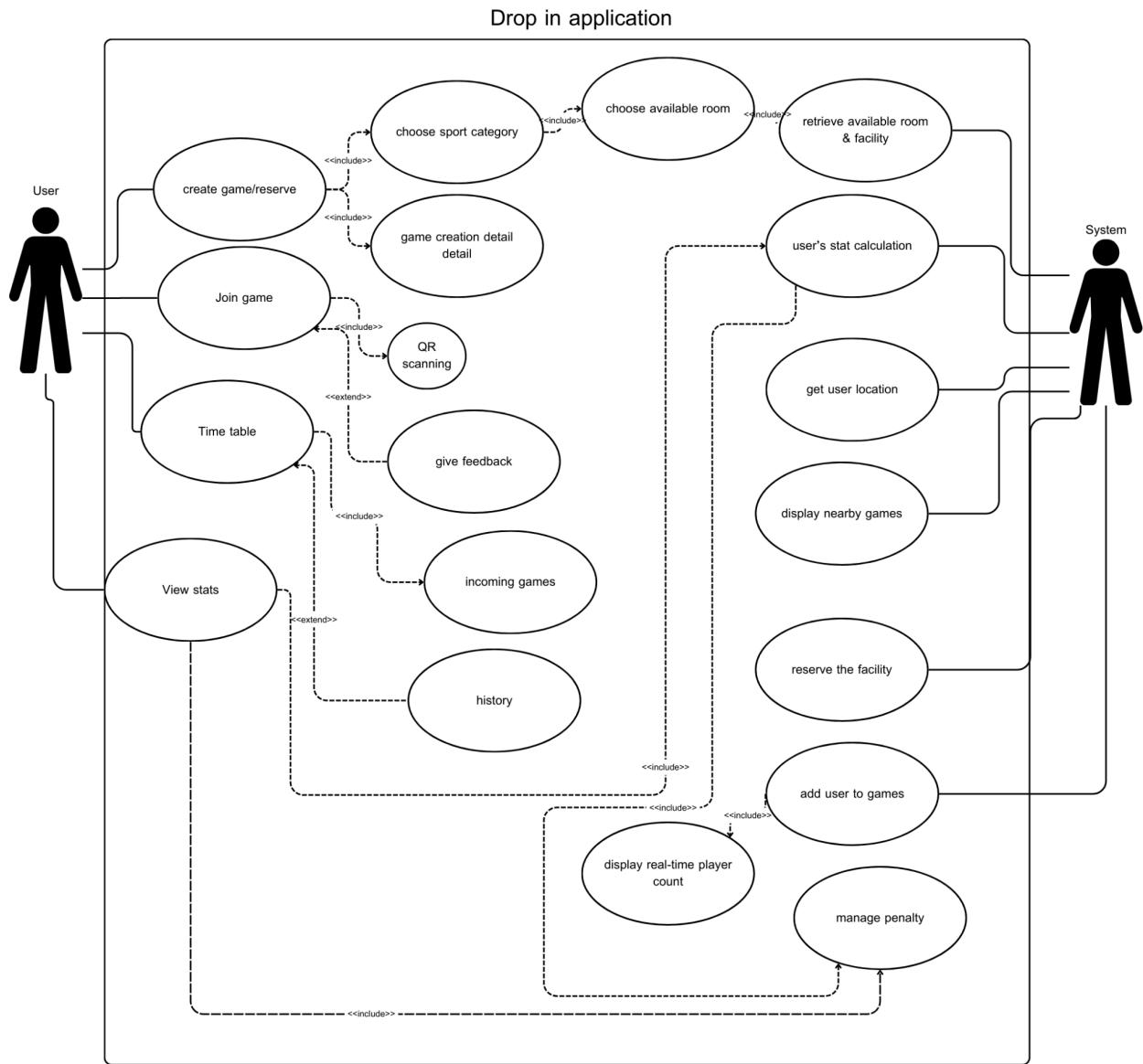
(a) Software Process Model

Incremental Model: Given that there is no rush for the application development (as the app is a standalone software with no external system dependencies), the project deadline is flexible and

dynamic. This makes models like RAD and Agile unnecessary. The project is based on existing technology that will be further developed and modified to meet the defined problem. Therefore, the evolutionary process model isn't suitable for our application development. Additionally, the concept of the project is clear, straightforward, and simple, allowing for further features to be added later to enhance the app's functionality. This aligns with the core principles of the incremental software process model, so we have decided to adopt it as the guideline for our development.

Activity diagram :





Use case diagram :

1.5 Technical Documentation

(a) Programming Languages

Dart: Dart is a programming language popular for mobile app development. It is strongly typed (like Java and C++) and Object-Oriented, making it easier for developers with experience in C++ and Java to learn. Dart is compatible with cross-platform mobile projects, supporting both iOS

and Android, which has become a standard in the industry. It also offers libraries and frameworks for both UI/UX and logic design, making it more efficient than traditional web development, which often requires multiple languages like HTML, CSS, and JavaScript. Dart provides everything in one language.

SQL (Standard Query Language): Our application uses MySQL for the database, and SQL is the language used to manage it. SQL is used to add, delete, and manage database schema and raw data in the database.

(b) Tools and Environments

Application Framework:

Flutter: Flutter is a cross-platform UI toolkit that uses Dart. It acts like a game engine (similar to Unity) for apps. Instead of relying on standard Android XML buttons, Flutter uses its own rendering engine to draw everything on the screen. This allows us to write code once and deploy it to both Android and iOS without needing to learn Swift (for iOS) or Kotlin (for Android) separately.

Database:

MySQL Database (self-hosted within the university): To store large amounts of user data, including personal information and sports game data, we use MySQL. It is an open-source database widely used by small teams and large corporations due to its reliability.

Development Environment (IDE):

Visual Studio Code (VS Code): A popular and easy-to-use code editor that most developers are familiar with.

Android Studio (SDK Tools): Used mainly for its Android Emulator and build tools (Gradle).

This lets us simulate an Android device on a computer to test features like the "Location" map and "QR Camera" permissions during development.

Version Control:

Git & GitHub: Used for coding sharing and code version management. It acts like a shared google drive with fork and merge functions to help with code version control. Also widely used around the world in tech sectors.

1.6 User Interface design

User journey

Case: User creating session

Journey Phases	Creating session	Config session detail	Book session	People join	Meet up and check in at destination	Check-out	Feedback
Step	<ul style="list-style-type: none"> want to play sport open app click create button 	<ul style="list-style-type: none"> Choose sport Select location Select room Select session privacy setting to public, 5 people 	Click confirm	<ul style="list-style-type: none"> Wait for people to join People join 	<ul style="list-style-type: none"> Follow the map to the location Wait for people at the location People arrive and check in 	<ul style="list-style-type: none"> Let other come scan check-out qr code with you 	<ul style="list-style-type: none"> Report toxic people who join
Touch Points	<ul style="list-style-type: none"> Smart Phone Dropin application Create page 	Session detail page	session detail page	Incoming session page	<ul style="list-style-type: none"> Selected location Random teammate Check-in page 	<ul style="list-style-type: none"> Check in page 	<ul style="list-style-type: none"> non
Experience	Don't know where to play sport and have no friend	Don't know any places but see the list of location and see the map to that destination	none	Nervous will people join?	Ready to play and meet new friend introduce myself to everyone when they come and scan check in qr code with me	Time to say good bye, everyone check out	Feel bad one of the teammate is toxic then report him
Emotion	bored	Normal	Normal	Excited	Excited	Good	Bad
Suggestion	can create button be bigger and easier to find?	will be nice if there a list of equipment the facility provide so I dont need to bring everything with me	none	It will be nice if I can see their profile	The map can be more detailed.	Why they can't check out themself	can I give feedback to the session be for me they refuse to leave the court on time

Case: Private game

Journey Phases	Creating session	Config session detail	Book session	Friend join	Go to meet with friend at the destination	Check out	Feedback
Step	<ul style="list-style-type: none"> want to play sport open app click create button 	<ul style="list-style-type: none"> Choose sport select location select room select session privacy setting to private 6 people 	Click confirm	<ul style="list-style-type: none"> Wait for friend to join Friend join and confirm session 	<ul style="list-style-type: none"> Follow the map to the destination with friend Scan qr code with the session owner for check in 	<ul style="list-style-type: none"> everyone come scan qr to check out 	non
Touch Points	<ul style="list-style-type: none"> Phone Dropln application Create page 	Session detail page	Session confirmation page	Incoming session Page	<ul style="list-style-type: none"> Map page Selected location Check in page 	<ul style="list-style-type: none"> Check in page 	non
Experience	Don't know where to play sport but have a lot of friends	Don't know any places but see the list of location and see the map to that destination	non	<ul style="list-style-type: none"> Tell friend to join Wait for friend to join 	Dont know where the location is but follow the map	Session owner put out the qr for everyone to scan	non
Emotion	normal	normal	Normal	Excited	Amazing	good	non
Suggestion	can create button be bigger and easier to find?	will be nice if there a list of equipment the facility provide so I dont need to bring everything with me	none	Can I just invite them instead	can they just not check in since its a private game	Why the session owner can't just click end session	I want to rate my friend

Case: User Joining session

Journey Phases	Filter to find the session list	select session	Book session	Meet up and check in at destination	Check-out	Feedback
Step	<ul style="list-style-type: none"> Open app Find join button Filter desired game information Scroll through list of session 	<ul style="list-style-type: none"> Click the desired session with availability See the location in the map 	Click join and confirm	<ul style="list-style-type: none"> Follow the map to the destination Find the session owner Let other come scan Check in qr code with you 	<ul style="list-style-type: none"> Go scan qr code for check-out 	<ul style="list-style-type: none"> Rate the session owner and the session
Touch Points	<ul style="list-style-type: none"> Phone Dropln application Join page 	<ul style="list-style-type: none"> Join page Search result page 	Confirmation page	<ul style="list-style-type: none"> Map page Session location Check in page 	<ul style="list-style-type: none"> Check in page qr code scanner 	<ul style="list-style-type: none"> Feedback page
Experience	Don't know where to play sport and have no friend then go to application and find session	See lot of session have to choose	Excited to go play	Can't wait to meet new people and play looking for the session owner to scan qr code for check in	Go scan check out qr code with the session owner and say goodbye	Feel good leave good review to the session owner
Emotion	Sad	Normal	Excited	Amazing	Good	Good
Suggestion	Can the join button be easier to find	It would better if results of games can be displayed in order the way I want.	none	<ul style="list-style-type: none"> Why do I have to scan can I just click check in 	<ul style="list-style-type: none"> Can I check out by myself 	<ul style="list-style-type: none"> The session before me is very nice they lent me the equipment can I rate them too

User persona

User Persona



Manoach, "the game joiner"

Manoach: 20-year-old Computer Engineering student

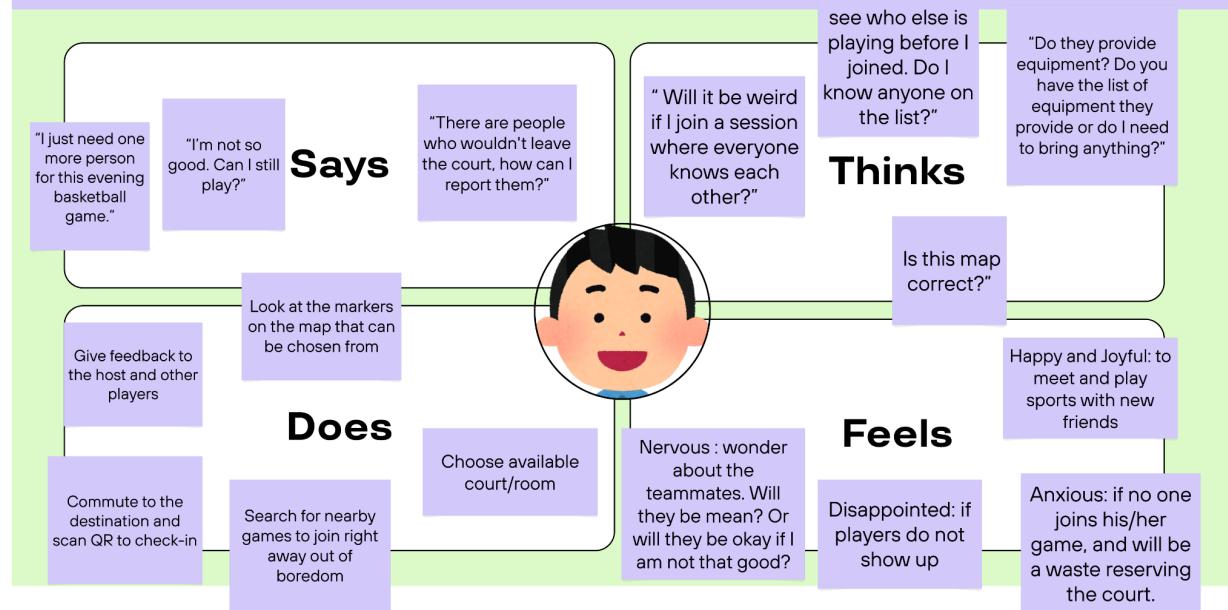
Top 3 favorite sports: Basketball, Volleyball, and Swimming

App Use Frequency: 2-3 times a week

Reliability Status: High (rarely miss check-ins/outs)

Empathy map

Empathy Map



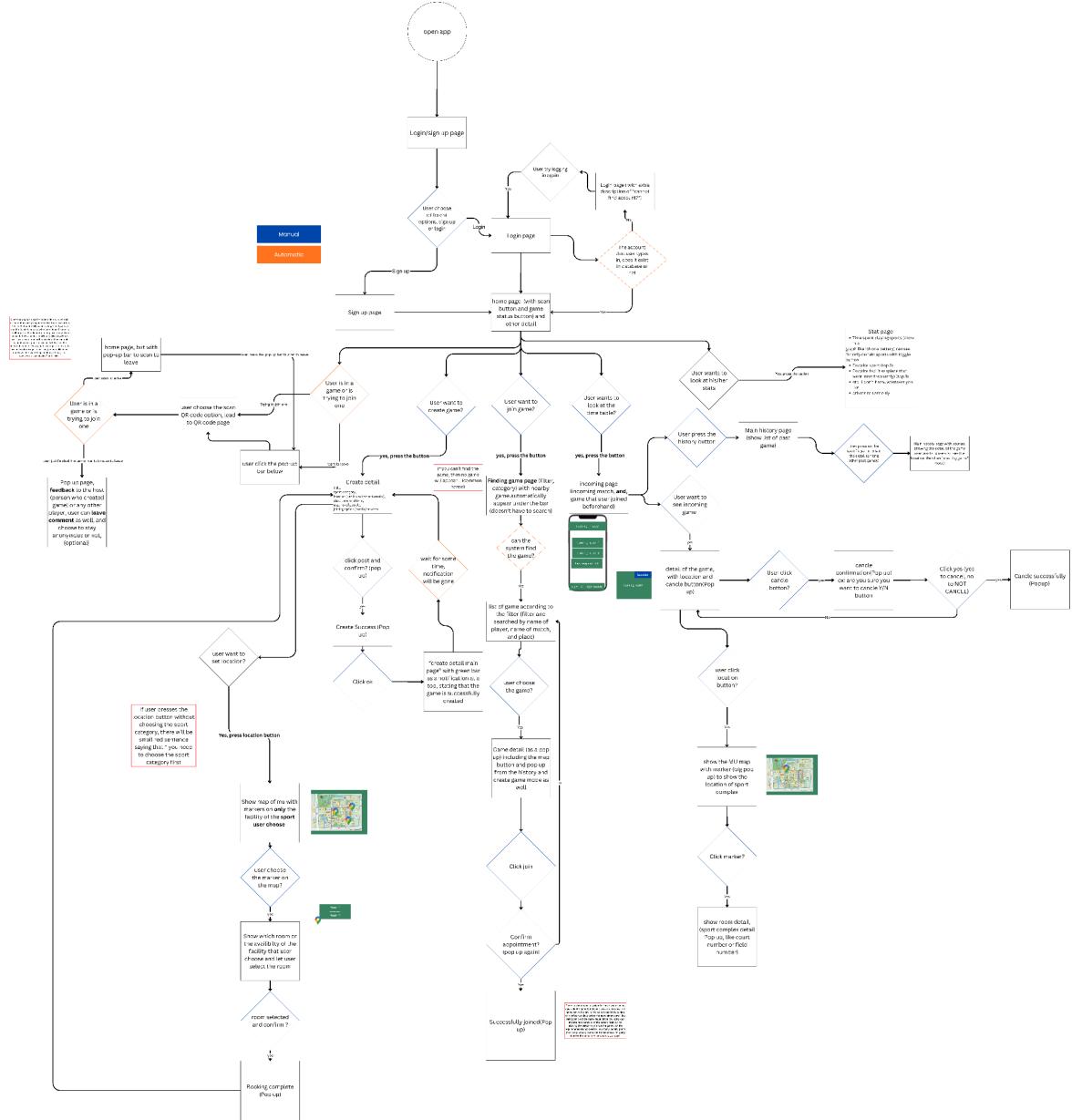
Pain

- Game creators (hosts) have to wait for others to join, and are not guaranteed that the players will show up.
- Users may suffer from failure of scanning QR code, especially when they are late and the lightning is poor.
- There is no way to know if a stranger joining is compatible, or friendly.

Gain

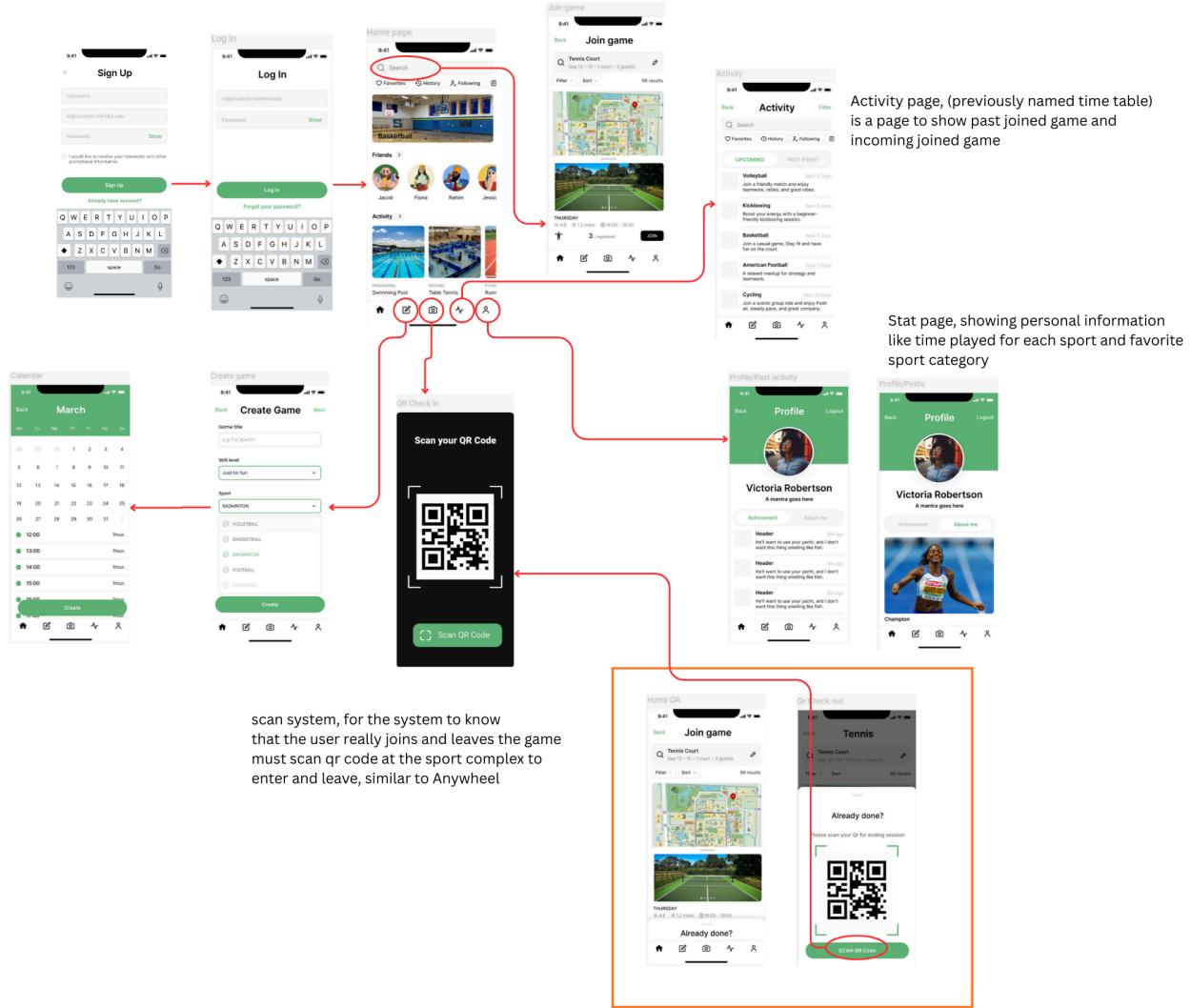
- Every sports complex in MU can be reserved all in one application.
- Easy for freshmen who have no idea where the building is since we have a map and location detail ready to use.
- Not everyone has lots of friends and some sports require more than one person so to be able to join other sessions even though you have no company is convenient.
- Users are all verified MU students. In case of any incident or emergency, the records of who, when, and where can be tracked. Help is easier and faster to seek for.

User-flow diagram



Ui and explanation.

join game page, to find a game to join. The game showing without searching is a game that is happening and near user's location
Can do searching with filter



Activity page, (previously named time table) is a page to show past joined game and incoming joined game

Stat page, showing personal information like time played for each sport and favorite sport category

Home page and pop-up “already done?” that is always there when user is in the game user can scan to leave

Ui : clean

Sign Up

Log In

Home page

Join game

Activity

Calendar

Create game

QR Check in

Profile/Past activity

Profile/Posts

Home QR

Qr Check out