

IM3080 Design and Innovation Project

(AY2023/2024 Semester 1)

PROJECT REPORT



Title: JamStream

Github: <https://github.com/aftanza/DIP-Group-4>

Submitted by: Group 4

Supervisor: Assoc Prof Mohammed Yakoob Siyal

Table of Contents

1. BACKGROUND AND MOTIVATION.....	3
2. OBJECTIVE.....	4
3. REVIEW OF LITERATURE/TECHNOLOGY.....	6
3.1. Project Management.....	6
3.2. Application Design.....	6
4. DESIGN AND IMPLEMENTATION.....	9
4.1. Interface Design.....	9
4.2. System Design.....	11
4.3. Implementation.....	12
5. CONCLUSION AND RECOMMENDATION.....	14
5.1. Conclusion.....	14
5.2. Recommendation for Future Works.....	15
6. BIBLIOGRAPHY.....	16

1. BACKGROUND AND MOTIVATION

The music consumption landscape has transformed vastly with the emergence of music streaming services revolutionising the way consumers engage with and derive enjoyment from music. Music streaming services provide convenience and access to diverse content, replacing the traditional approach of purchasing physical copies or individual digital tracks and opening up new opportunities for musicians and music lovers.

In Singapore, the music streaming service industry is dominated by a few apps, namely Spotify, Apple Music and Youtube Music (Wong, 2023). The table lists their value proposition, including some of their well known features.

Spotify	Apple Music	YouTube Music
<ul style="list-style-type: none">○ Personalised playlists based on user preferences○ Collaborative playlists created with friends	<ul style="list-style-type: none">○ Integration with the Apple ecosystem	<ul style="list-style-type: none">○ Integration with the YouTube platform and recommendations based on ‘videos watched’ history○ Premium accounts gain access to videos linked to the music played

However, we discovered that there was an absence of a dedicated feature where users can actively share their music taste and experience with the broader online community. Thus, besides offering an extensive catalog of songs, we strive to develop an app that will enhance the whole music discovery and listening experience. In a landscape driven by algorithm-driven content recommendations, we seek to empower users with greater autonomy in customising their personal musical experience which accommodates a wide range of tastes and preferences.

2. OBJECTIVE

a. Enhance virtual socialisation

On consuming music streaming services, ‘identity salience effects have been found on frequency of identity-related behaviours in music sharing’ and there are connections between ‘self-identity factors’ and ‘entertainment streaming applications (ESA) attachment, value or use’ (Oyedele, 2018). In short, consumers give these ESAs a perceived social value and consume music streaming services as it aids in building an aspect of their social identity.

Our streaming rooms feature a chatroom, which provides an alternative to seamlessly share the user’s music experience in real time, without relying on an external messaging app. This creates a digital community where like-minded individuals can gather, exchange and bond over their music taste. The music played in the room will be synchronised across all users’ devices, enhancing the shared experience and providing a space for users to socialise with others in the online community.

b. Stimulate music discovery

The common music streaming app attracts consumers with their diverse collection of tracks and albums, allowing users to explore and discover new music. These apps deliver personalised playlist, artist and song recommendations based on user data, linked to other applications such as YouTube or Google’s search history.

JamStream empowers users with greater autonomy, enabling them the freedom to choose from a list of available streaming rooms with its own vibe and playlist, decided by the users in the room. Users can also browse through the songs in the queue to discover songs which belong to the room’s genre, vibe or the music taste of others. Users have the flexibility of

either creating public or private rooms, which are complete with a personalised touch. They can choose a display picture and description that aligns with their mood or vibe for the day.

3. REVIEW OF LITERATURE/TECHNOLOGY

3.1. Project Management

a. Jira

Developed by Atlassian, Jira offers a centralised platform for the team to distribute, manage and track assigned tasks. This tool's transparency and real-time updates also promotes collaboration among team members.

b. Github

Github is a widely used version control system used for software development, allowing the team to track changes to the code while enabling everyone to work on the project simultaneously in their own branches.

3.2. Application Design

a. Canva

Canva is an online graphic design platform that is used to create social media graphics and presentations. Its user-friendly interface and extensive design tools and assets empowered us to create our logos and promotional posters with ease, which helped us bring a visually appealing and professional touch to the app's branding.

b. Figma

Figma is a collaborative web application for interface design, with a simple layout for mobile application design and real-time collaboration features. Our group decided to use Figma to create our initial JamStream design and prototype before starting on implementation, as it helped us greatly in providing a clear and helpful guide during implementation, and keeping our implementations consistent in terms of design and navigation.

c. React Native

React Native is an open-source UI software framework, which utilises the JavaScript programming language to enable developers to use the React framework along with native platform capabilities. Its extensive libraries and myriad of features helped us greatly in implementing the features we wanted to add to our JamStream app, such as accessing our device's files to upload images. Its capability for cross-platform application development also attracted us to utilise React Native as opposed to other platforms such as Android Studio.

d. Expo

Expo is a flexible, robust and convenient framework used to build React Native apps, providing developers with a list of tools that simplify the creation and testing of React Native applications (Borozenets, 2022). It serves as our preferred addition to React Native for our application development, as it provides additional features such as real-time remote application testing and a large set of libraries that help simplify our JavaScript codes yet not compromise on the quality of our application.

e. Firebase

Firebase is a set of backend cloud computing services and application development platforms provided by Google, enabling developers to host databases, services, authentication, and integration for mobile and web applications. We utilised Firebase for our backend database due to its extensive real-time synchronisation capabilities and ability for collaboration across multiple simultaneous devices.

f. Spotify API

The Spotify Web API is a restful API with different endpoints which return JSON metadata about music artists, albums, and tracks, directly from the Spotify Data Catalogue (*API Calls | Spotify for Developers*, n.d.). By implementing an authorisation flow through a redirect link, our application utilises the Client ID and Client Secret requirement to request access tokens, which are strings that contain the credentials and permissions that can be used to access a given resource (e.g artists, albums or tracks) or user's data (e.g a user's profile or personal playlists).

We have utilised the Spotify API to acquire essential initial data for the application's functionality. This includes retrieving the user's profile name and display image, compiling a list of the tracks in a user's current queue and accessing their personal playlists. This information will then be displayed on the home page after the user logs in with their Spotify account. For the song search feature, API calls were made to fetch an array of relevant tracks based on the entered search term, which will play clips of the songs upon user input.

4. DESIGN AND IMPLEMENTATION

4.1. Interface Design

a. Colour Scheme

Inspired by the concept of remote group study and gaming sessions, our colour palette has vibrant neon teal as the primary colour with tropical indigo and royal blue as secondary colours. The primary colour represents creativity, enjoyment and the engaging nature of the app whereas the different shades of blue convey a sense of serenity, reliability and trust. Together, these colours encourage users to forge a connection not only with the music played but also among other users, crafting a memorable experience for all.

We opted for platinum and rich black instead of pure white and black. This mitigates the sharp contrast between light and dark colours, reducing eye strain for a more comfortable user experience.

b. User Interface (UI)

With Spotify as our reference point, we aim to provide users with an intuitive and seamless user experience. According to Jakob's Law, we considered users' existing mental models and in order to meet users' expectations, we prioritised a familiar UI that is found in most music streaming platforms (Yablonski, n.d.).

By incorporating colours from our chosen colour palette, we established visual hierarchy on each page, ensuring users can easily navigate through the app. We also made sure that users are informed about the system status by providing immediate and responsive visual feedback for every user action, creating an engaging user experience (*10 Usability Heuristics for User Interface Design*, n.d.).

c. User Experience Flow

The user experience flow of the room feature, which begins with creating a room or joining a room, needs to be intuitive and logical. To ensure the cohesiveness of the navigation, we had to create coherent and smooth transitions to the other pages, such as the create queue page and chatroom page.

When a user joins a room, the queue shifts to align with the synchronised music played in the room. This introduces a need for the effective management between the user's personal queue and any potential room-specific queue that the user might adopt. Our approach involves utilising the Firebase database, which compelled us to purposefully decide on the optimal instances – be it a button press or navigation to a specific page – when the respective queues should be saved, retrieved or deleted. This extends to the addition and removal of songs from the queue.

4.2. System Design

a. Use case Diagrams

We drafted out a use case diagram to visually represent the various interactions and functionalities within the app. Each use case captures a common user scenario and its corresponding system behaviour. For a breakdown of these use cases and its details, refer to Appendix A.

b. Sequence Diagrams

The sequence diagrams provide a comprehensive view into each user action, serving as a visual representation of the chronological order of activities. Additionally, it depicts the flow of information to and from different sources. For a breakdown of these sequence diagrams and its details, refer to Appendix A.

4.3. Implementation

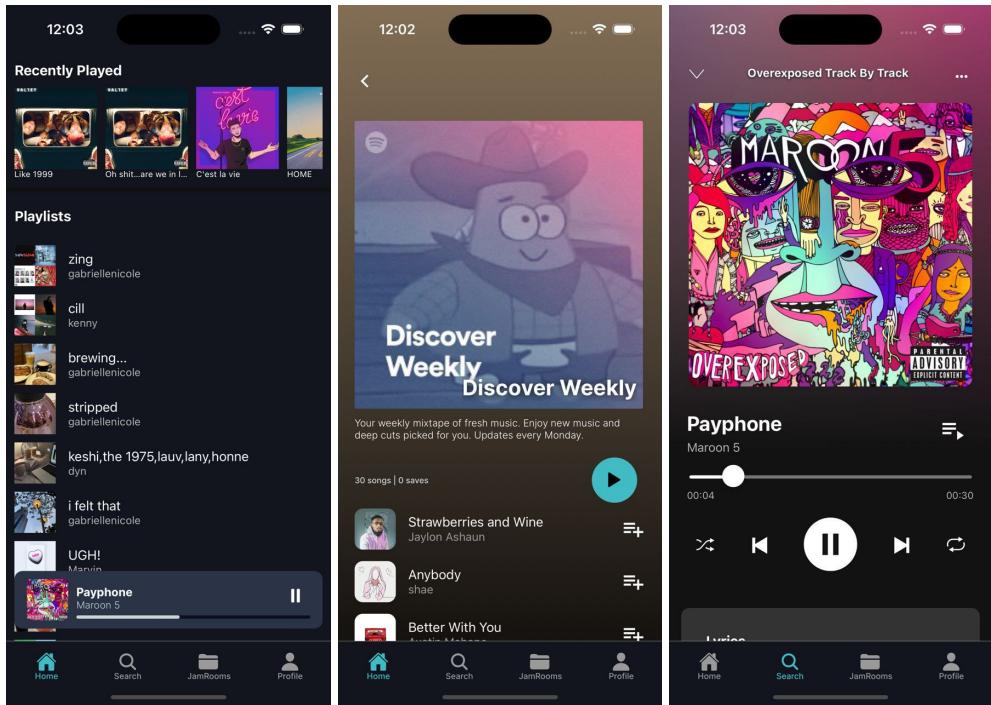


Fig. 1: Home page, Playlist page, Track page

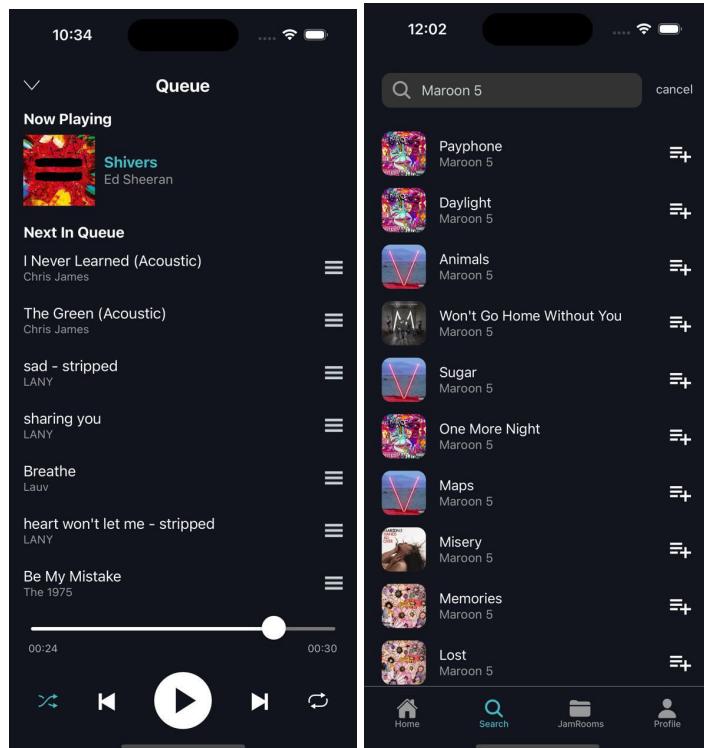


Fig. 2: Queue page, Song Search page

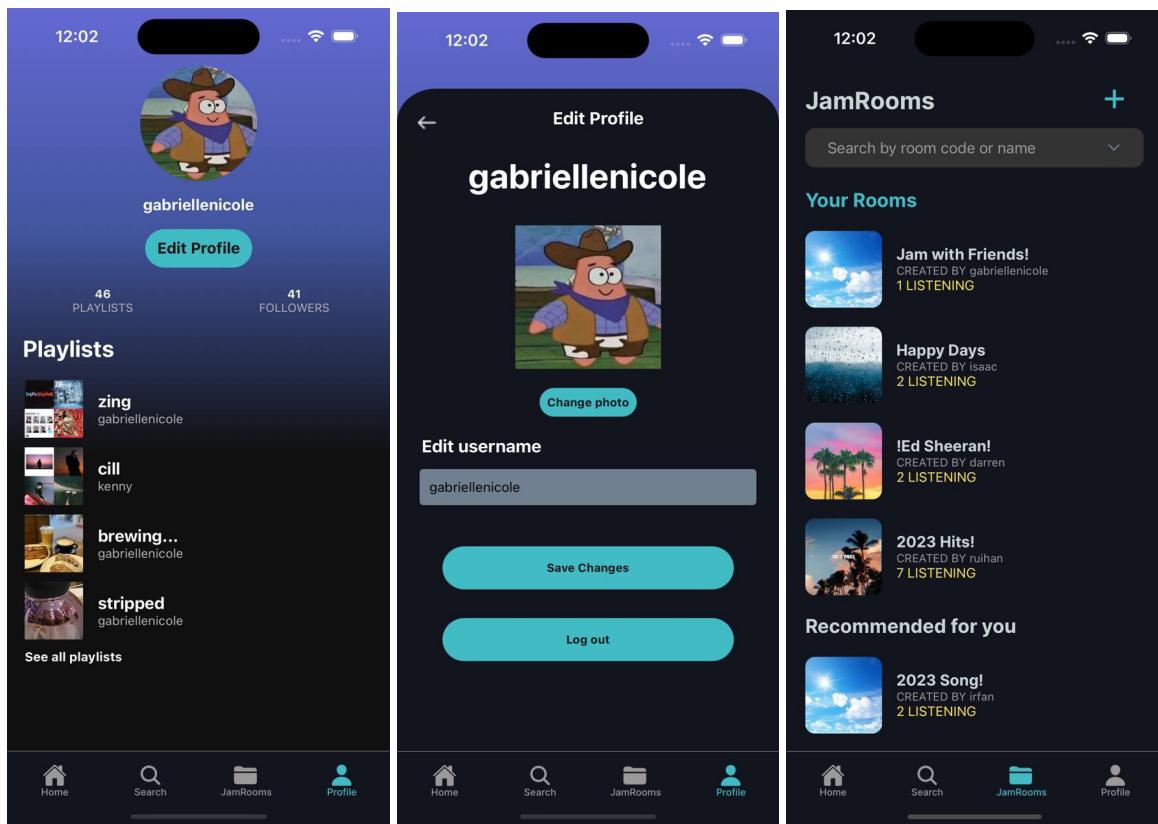


Fig. 3: Profile page, Edit Profile page, JamRooms list page

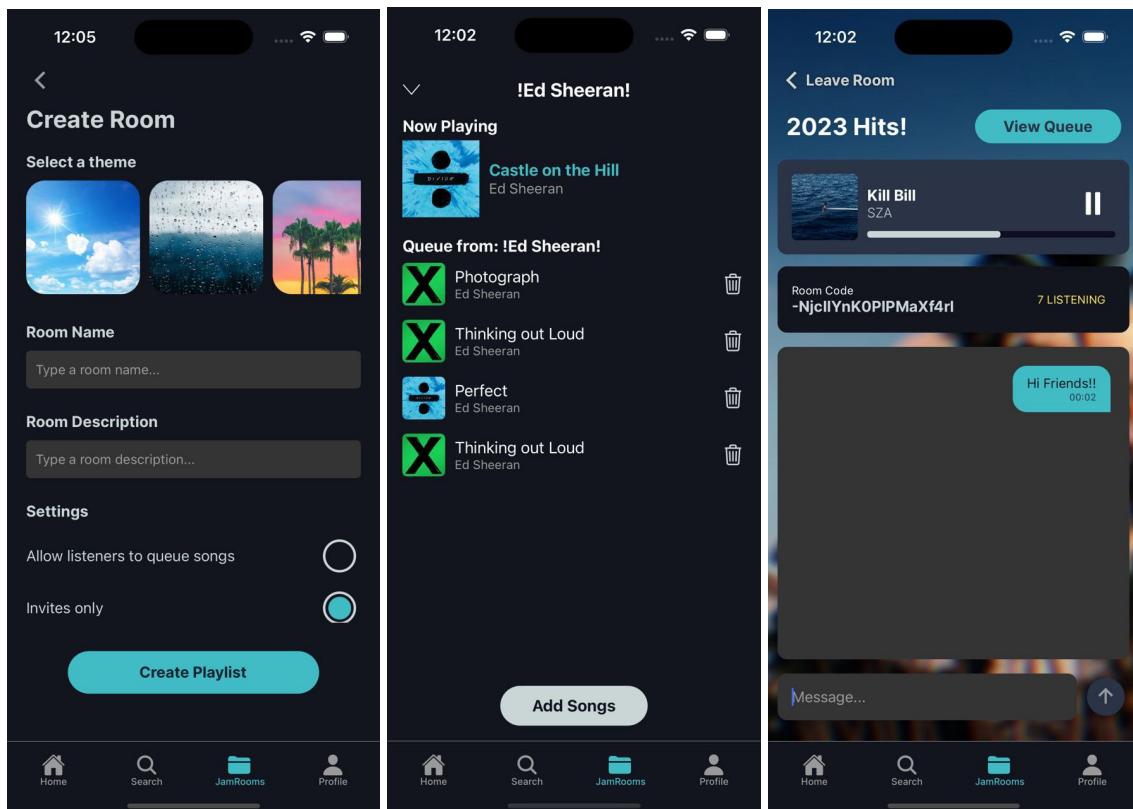


Fig. 4: Create Room page, Create Playlist/Queue page, Chatroom page

5. CONCLUSION AND RECOMMENDATION

5.1. Conclusion

The journey of developing JamStream has been an enriching experience, offering valuable insights into software development projects. The project began with a thorough exploration into existing music streaming apps, identifying areas for improvement and envisioning unique features that would address specific user needs. Along the development process, we took into consideration the user experience from the user perspective, as well as the technical requirements for the app to realise. In the end, we navigated through complexities, learned to appreciate different perspectives and recognised that successful projects are seldom straightforward.

We also realised that effective communication is key to teamwork, especially given the need to split into subgroups with their areas of specialisations. Open and transparent communication is indispensable in order to understand each other's motivations, ideas and to overcome challenges.

5.2. Recommendation for Future Works

While we have successfully accomplished our project goals, the limited timeframe of 13 weeks with a 10-member team proved insufficient for the complete realisation of the app. Given more time, we have considered implementing additional features to enhance the user experience.

These improvements include adding advanced user profile customisation options, which provides users with a broader range of personalisation choices for their profiles. Additionally, we also started a discussion about introducing gamification features within the JamRooms, enhancing the quality of interactions among users in the community and increasing user dwell time. These enhancements would contribute to a more engaging platform, with alignment to our project objectives, as an expansion to JamStream capabilities.

6. BIBLIOGRAPHY

- a. Wong, B. (2023, August 10). *Spotify VS Apple Music VS YouTube Music VS Tidal: Which music streaming service is worth paying for?* DollarsAndSense.sg.
<https://dollarsandsense.sg/music-streaming-service-worth-paying/>
- b. Oyedele, A., & Simpson, P. M. (2018). Streaming apps: What consumers value. *Journal of Retailing and Consumer Services*, 41, 296–304. <https://doi.org/10.1016/j.jretconser.2017.04.006>
- c. Borozenets, M. (2022, October 13). React Native Init vs Expo 2022: What Are the Differences? *Fulcrum*. <https://fulcrum.rocks/blog/react-native-init-vs-expo#what-is-expo>
- d. *API calls | Spotify for Developers*. (n.d.).
<https://developer.spotify.com/documentation/web-api/concepts/api-calls>
- e. Yablonski, J. (n.d.). *Jakob's Law | Laws of UX*. Laws of UX.
<https://lawsofux.com/jakobs-law>
- f. *10 Usability heuristics for user interface design*. (n.d.). Nielsen Norman Group.
<https://www.nngroup.com/articles/ten-usability-heuristics>

a. Design Diagrams

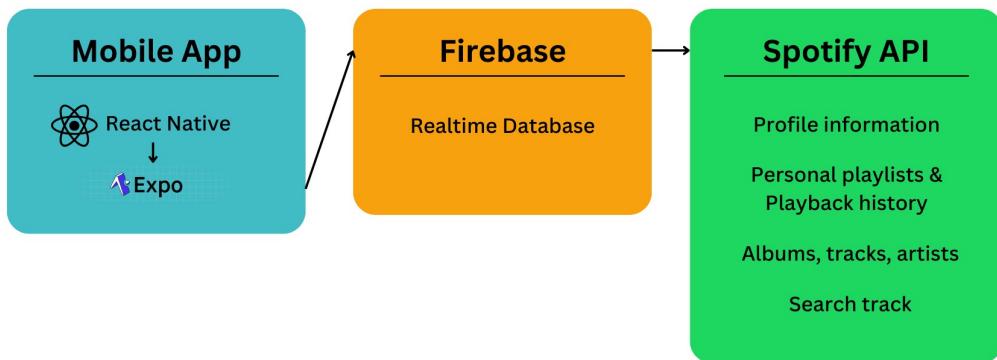


Fig. 5: Architecture Diagram

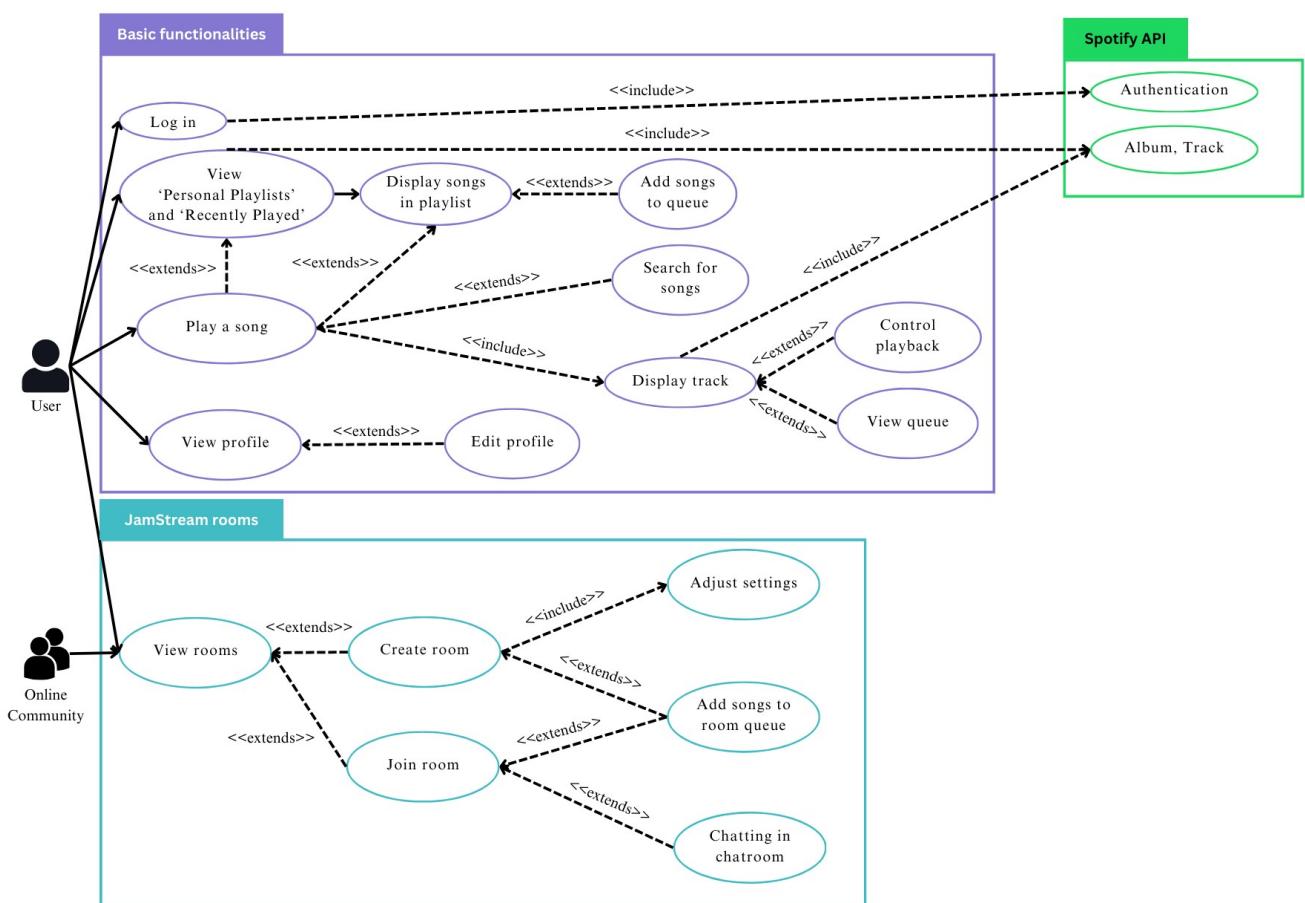


Fig. 6: Use-case Diagram

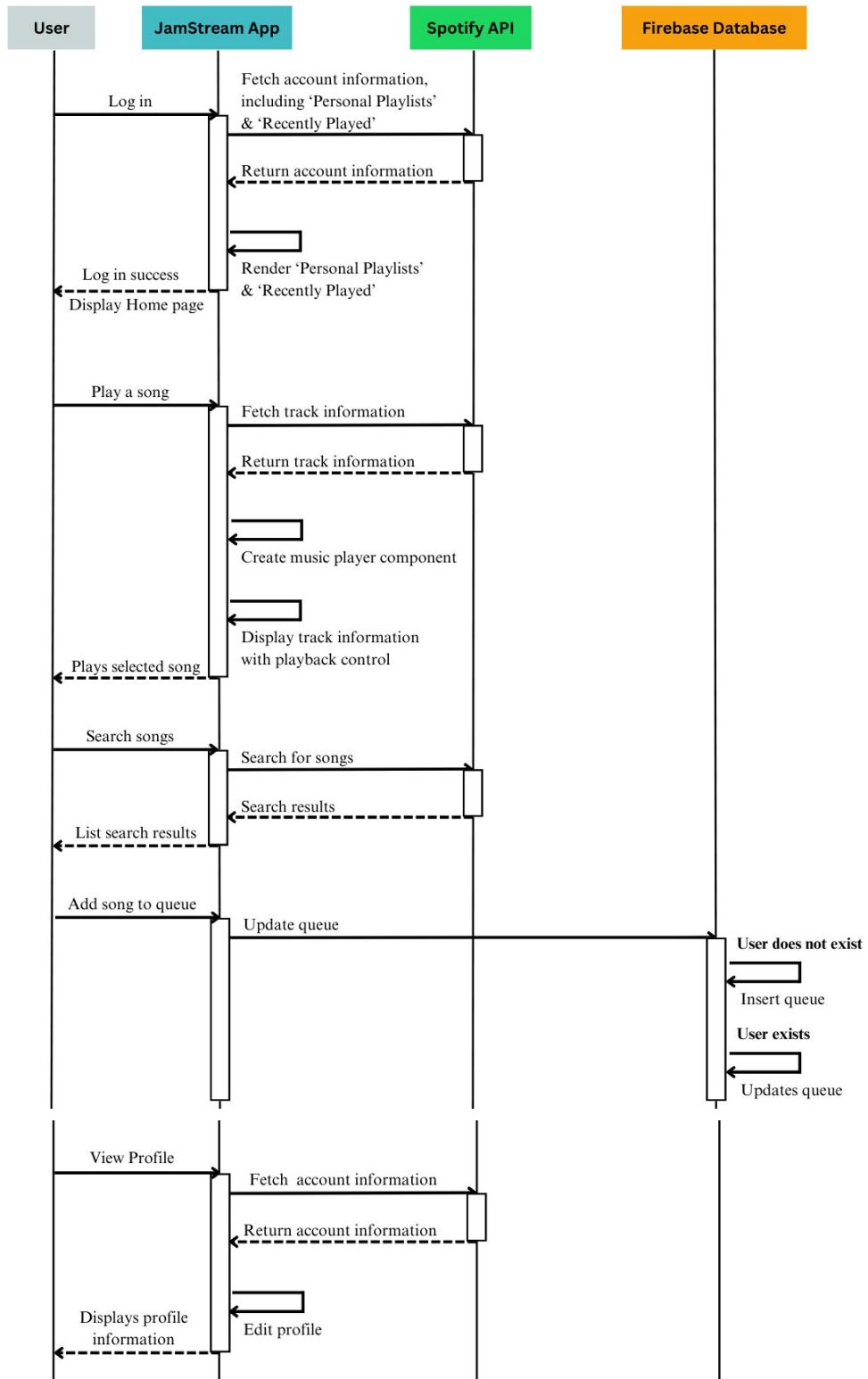


Fig. 7: Sequence Diagram for Basic Functionalities

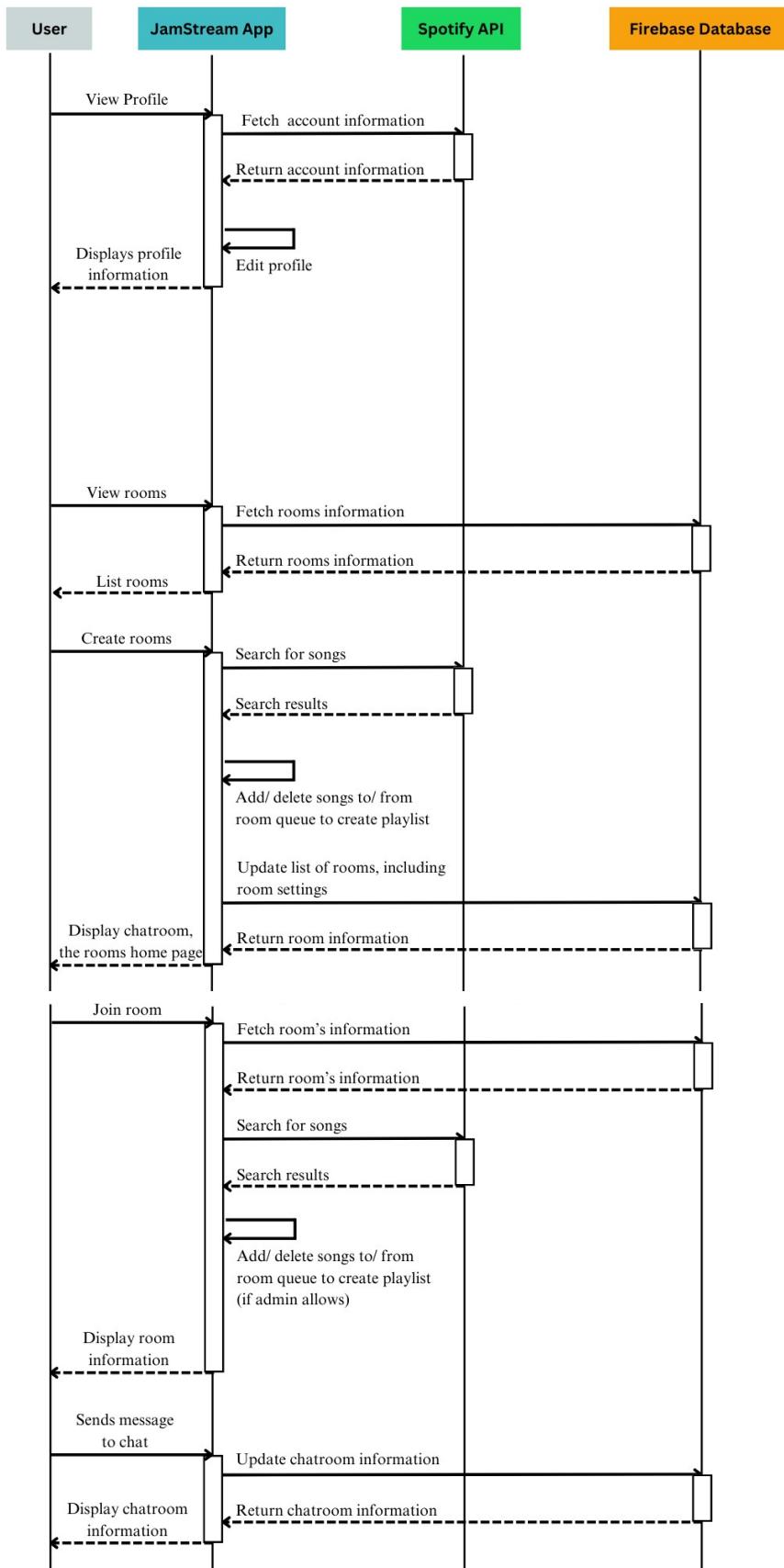


Fig. 8: Sequence Diagram for Rooms

b. Guides & Source Code



Fig. 9: User guide

a. Maintenance guide

For more information about connecting to the Spotify API, please refer to the [link](#) here.

b. Source code

<https://github.com/aftanza/DIP-Group-4>