

SENG2021

MOSH

**A social media platform focussed
on connecting people through music**

**BY: team_aace
Austin Vuong, Alvin Cheng, Cuthbert Zhong,
Elliott Yu, Luke Elsabbagh**

Problem Statement

In today's era of multimedia entertainment and streaming services, music platforms such as Spotify, Apple Music and SoundCloud have provided an easily accessible way for users to listen to their favourite tracks and artists. Each of these platforms have a set of associated advantages and disadvantages which we will explore in the following section.

Spotify

Advantages

- Large range of music and artists to choose from
- Provides recommendations of playlists and tracks to users
- Can follow artists and create personal playlists
- Can listen to radio
- Free service

Disadvantages

- Limited functionality of friends between users
 - Unable to share posts, messages or music with other users
- Can only view user's other playlists and followed artists

Apple Music

Advantages

- Paid premium service
- Can listen to podcasts
- Can search by lyrics
- Can download music tracks

Disadvantages

- Exclusivity (No free service)
- No functionality of friends or interaction between users
- Exclusive for Apple users

SoundCloud

Advantages

- Get notified when followed artists releases music
- Can be shared with anyone
- Users can record and upload their own sound files

Disadvantages

- No functionality of friends or interaction between users
- No radio or podcasts
- Users are not able to follow or comment without a SoundCloud account
- Only in mp3 format
- Lack of Offline play support
- Expensive full package fee

As shown through the advantages and disadvantages of each platform, the lack of social networking and interaction is an issue that is prevalent in music streaming platforms. Whilst some platforms such as Spotify provide a limited functionality for interaction between users, there is no fully fledged media platform that implements a holistic range of features expected from a social media platform such as the ability to add friends and interact (message / share posts) with other users. Additionally, due to the intrinsic competition between music streaming platforms, it is often difficult to share playlists and tracks from one platform to another. It is this lack of interactivity between and within music streaming platforms that Mosh attempts to address.

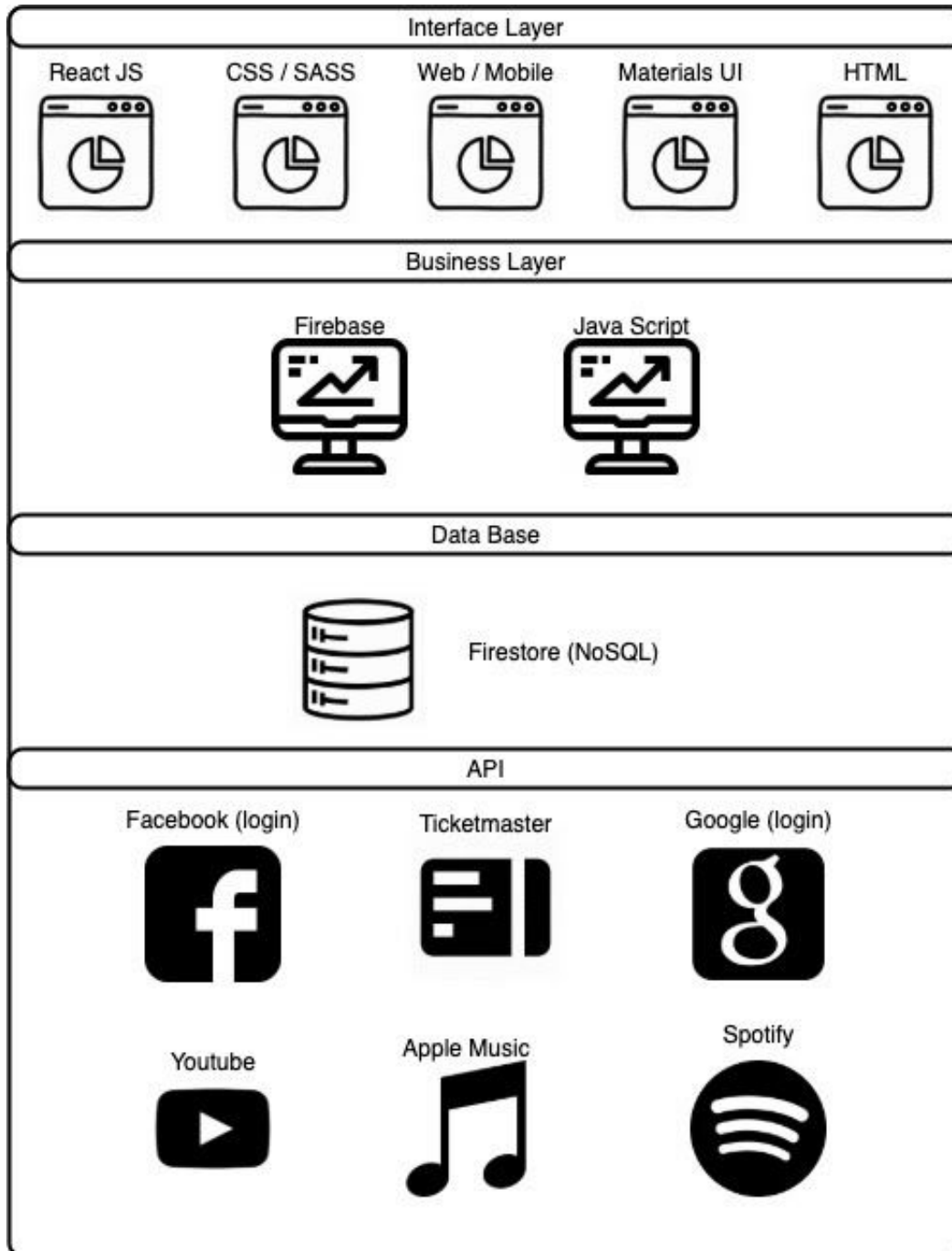
Purpose

Mosh is a social media platform aimed towards addressing the lack of social interaction features provided by existing media platforms such as Spotify. Through this, Mosh aims to create an integrated platform, where users can import their existing profile details from media platforms, connect and share music and ideas with other users of similar interests across all media platforms.

Mosh will implement features such as:

- The ability to import your existing details from music platforms such as Spotify
- Sharing your thoughts/ posts with other users of Mosh
- Suggest users to connect with based on your mutual interests and likes
- Suggest music/playlists based on what your connections are listening to
- Share music with other users of Mosh regardless of music platform
- List upcoming and relevant concerts around you based on your existing interests

Software Architecture



- **Interface Layer**

HTML is the standard markup language in web browsers. Web browsers can receive HTML documents from a web server and render such documents. It is supported by technologies such as CSS for styling as well as JavaScript for scripting and creating interactive web pages.

ReactJS is a modern JavaScript library ideal for web application development, enabling for efficient and effective design and development. Its use of class based components allows for easy reusability between web pages. Additionally, ReactJS allows for more interaction in a webpage through maintaining a state and updates to the DOM, depending on changes to the state of a ReactJS component.

Material-UI is a user interface framework built using ReactJS, which provides a variety of pre-rendered components such as lists, drawers, application bars and grids. This allows for easier web development as many popular user interface designs are already provided for us. Additionally, Material-UI provides CSS and Javascript support for responsive web pages, allowing for our application to be easily accessible across a range of devices.

Due to ubiquitous nature of JavaScript and HTML in web development, our technology stack allows for Mosh to run on a variety of web browsers such as Google Chrome and Safari on virtually any operating system. Through the use of web development frameworks such as Material-UI, our application will support a variety of devices from desktop to mobile browsers.

- **Business Layer**

JavaScript is the core scripting language used in web development, allowing for interactive web pages and applications. It has many supported frameworks such as ReactJS and Material-UI which allow for easy and more intuitive web development. It supports functionality of HTTP requests allowing our application to communicate with third party API's as well as our backend server built upon Firebase.

- **Database**

Firestore is a technology provided within the Firebase system, providing a NoSQL database infrastructure. This allows for efficient storing and retrieving of data through use of JSON documents. This allows for more efficient parsing and processing of data. Further, NoSQL databases have the flexibility to change over time. This is useful in the cases where system changes occur, allowing for changes to be made without a lot of refactoring.

Specifically, Firestore was chosen due to its ease of use and quick learning curve. Predominantly accessed/configured through a GUI and used through calls from the front-end, it allows for the database infrastructure to be developed and running quickly in order to support other functionalities of the project at an early stage.

Also, Firestore provides automatic and robust scaling for queries, performance, storage and synchronisation. This is ideal in this project as it allows for greater focus on

functionality and user experience rather than focus on the intricacies of the database infrastructure through development.

- **API**

- Google / Facebook**

- The use of these APIs will streamline the authentication process of the users of our platform by using Google and Facebook's inbuilt supported functionality of their respective accounts to log into third party applications. Additionally, Firebase allows for easy integration of Google and Facebook authentication.

- Apple Music / Spotify**

- The use of these APIs will allow users to search music and access recommendations or playlists. These will be used by our platform to fetch albums, artists, and songs.
 - Additionally, when a user links their Spotify or Apple Music, this API will be able to fetch personalised data about the user's listening habits, allowing for a more personalised experience.

- Ticketmaster**

- The use of this API will feed our platform with events for a particular artist or venue in a specific area in relation to the user's interests.
 - The API will provide details of the event and link to purchase tickets.

- Youtube**

- The use of this API will communicate with platforms, search for appropriate videos and provide links for users.
 - The use of the API will also allow users to integrate their interests within the platform based on their subscriptions.

Initial Software Design

Features

Requirement : A social media platform that is catered towards music interests.

Feature: Sign Up

As a user,
So that my data can persist on the platform
I want to be able to sign up.

GIVEN I am on the landing page
AND I do not already have a account
WHEN I click “sign up”
THEN I am shown a sign up page where I can enter my account details
WHEN I click ‘add’
THEN an account and profile are created
AND I am directed to my homepage.

Feature: Log In

As a user,
So that I can access my profile and account
I want to be able to log into the platform.

GIVEN I am on the landing page
AND I have an account
WHEN I click “sign in”
THEN I am shown a login page
WHEN I enter my valid account details
AND press enter
THEN I should be able to access the service
AND my homepage is shown.

Feature: Log out

As a user,
So that my account remains secure from outside access
I want to be able to log out of my account.

GIVEN that I am logged into my account
WHEN I am on any page
THEN I should be able to click “log out”
AND log out of my account

Feature: Edit Account Settings

As a user,
So that my account can better fit my needs
I want to be able to edit my account settings.

GIVEN I am on my profile page
WHEN I click “edit profile settings”
THEN I am shown a page of settings for my profile

GIVEN I am on my profile settings page
WHEN I click “change privacy”
THEN I am given a range of privacy settings
WHEN I click on a privacy settings
THEN my account privacy is set to that option if not already.

GIVEN I am on my profile settings page
WHEN I click “update avatar”
THEN I can either use a preregistered avatar or upload my own image so that
WHEN I click ‘update’
THEN my avatar is changed to that image.

GIVEN that I am on the “Profile Settings page”
WHEN I press the “deactivate account button”
AND press “yes” on the are you sure prompt
THEN my account should be deleted from the site

Feature: Import details from other media platforms

As a user,
So that I can have a more personalised experience,
I want to be able to import my existing details from other media platforms.

GIVEN I am on my profile
OR signing up
THEN I am shown a list of possible media platform connections
WHEN I click on a media platform button
THEN I should be able to login to a media platform
AND my account will become linked to my media platform account
AND my data from the other platform will be loaded into my profile.

Requirement: Recommend people, music and other features to me based on my personal preferences.

Feature: Discover people

As a user,
So that I can meet people of similar interests
I want to be recommended their profiles

GIVEN that I am on my home page
WHEN I click the “discover” tab
THEN I should be on the discover page
WHEN I click the ‘people’ side tab
THEN I am shown a list of people recommended to connect with based on our mutual interests.

Feature: Discover Concerts

As a user,
So that do not miss out on concerts that interest me
I want to be able to keep track of relevant local concerts around me.

GIVEN that I am on the discover page
WHEN I click on the “concerts” side tab
THEN I am shown a list of concerts happening around me based on my interests.

Feature: Discover Music

As a user,
So that I can broaden my horizons
I want to be able to discover new recommendations regarding music

GIVEN I have a music account linked to my profile
AND I am on the discover tab
WHEN I click on the music tab on the side bar
THEN I am shown a list of recommended tracks on the discover page
WHEN I click a track
THEN I am redirected to a media platform that plays the selected media

Requirement: Ability to interact with other users of the social media platform

Feature: Connect with other users

As a user,

So that I can build up my social network,

I want to be able to connect with other users of the social media platform.

GIVEN I am on the the “discover” page

AND have selected the “people” sidebar option

WHEN I click on a user’s profile

THEN I am shown that user’s profile

WHEN I click on the connect button

THEN I am able to add that user to my connections.

Feature: View existing connections

As a user,

So that I can keep track of my social network,

I want to be able to view my existing connections with other users.

GIVEN I am on the “discover” page,

WHEN I click on the “existing connections” sidebar button

THEN I am shown a list of existing connections with other users.

Feature: Remove connections

As a user,

So that I can modify my social network

I want to be able to unconnect with users that I have connected with in the past.

GIVEN I am on another user’s profile page

AND I am already connected with them

WHEN I click the disconnect button

THEN I am no longer connected with them.

Feature: Share thoughts

As a user,

So that others can be updated on my status,

I want to be able share my thoughts on the feed.

GIVEN I am on my homepage,

THEN I am shown a feed of other users’ posts

WHERE I can filter relevant posts by tag

OR view my own posts.

GIVEN that i am on the homepage
WHEN I click “post update”
THEN I should be able to post an update
AND attach relevant tags.

Requirement: Social Media platform lacks a way to send tracks to other users on a variety of music platforms seamlessly.

Feature: Map music tracks

As a user,
So that I can more easily listen to music
I want to be able to map music tracks from one platform to another.

GIVEN I am logged in
AND have selected a music track
WHEN I click on that track
THEN I am shown links to open that track on various media platforms.

Feature: Send music tracks to other users

As a user,
So that I can share my interests with other people
I want to be able to send music tracks to other users.

GIVEN I am logged in
AND on the “share music” tab
WHEN I click the “send track” button
THEN I am able to specify a music track
AND send it to a particular user who I have connected with.

Feature: Receive music tracks from other users

As a user,
So that I can experience what others are listening to
I want to be able to receive music tracks sent to me by other users.

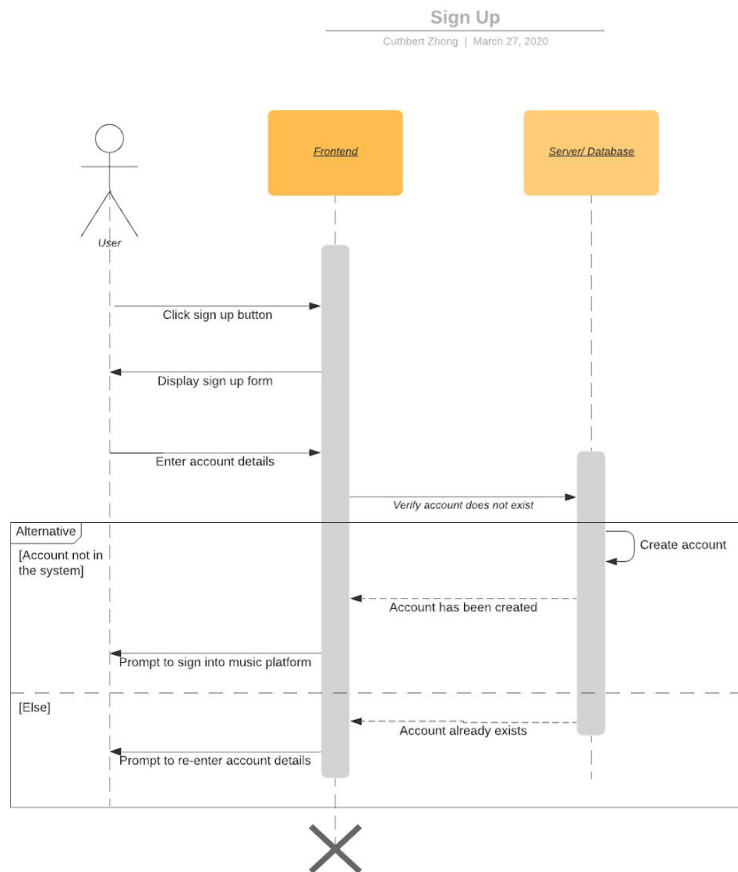
GIVEN I am on the “discover” page
WHEN I click on the “shared with me” page
THEN I am shown a list of music tracks sent to me by other users.

Sequence Diagrams

Sign Up

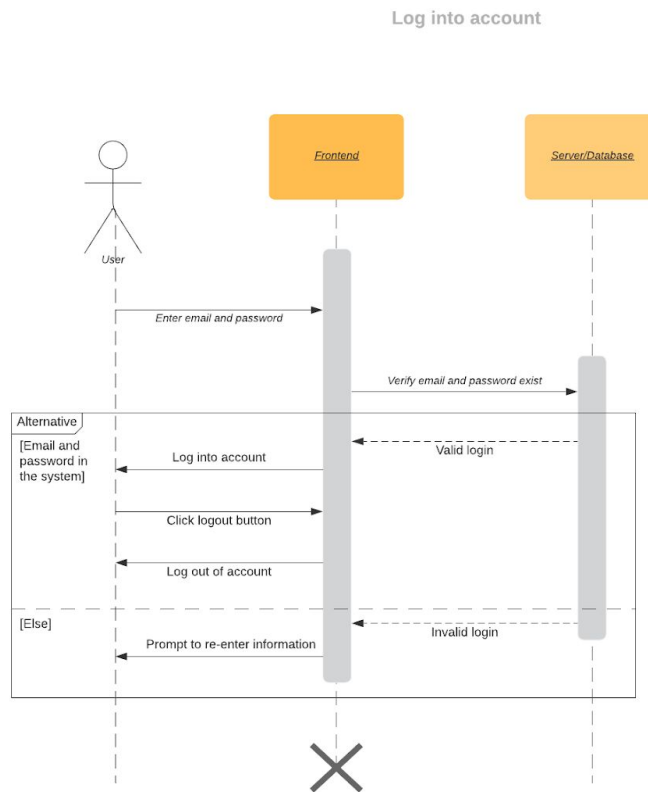
A user without an account clicks sign up

- They enter the email and password they want to use for their account
- The server will then check that the details the user is trying to make an account with is valid and an account doesn't already exist with that information



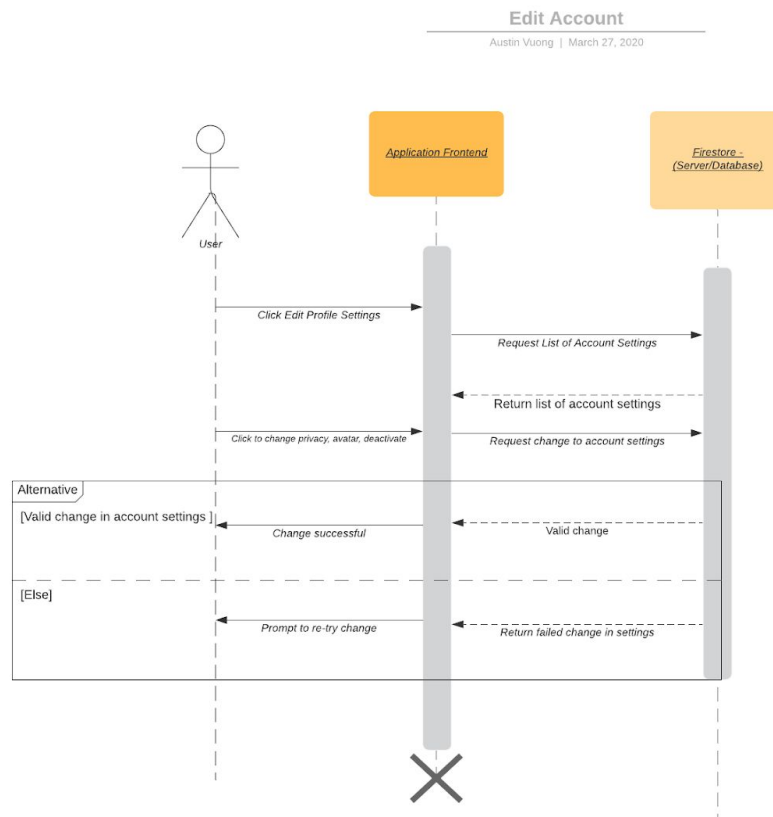
Log in / Out

- A user on the login screen will enter the details to their account
- The server will then ensure the login is valid for an existing account
 - If the login is valid the user will be logged in
 - The user can then log out of their account
 - If the login is invalid the user will be informed and can try to re-enter their details



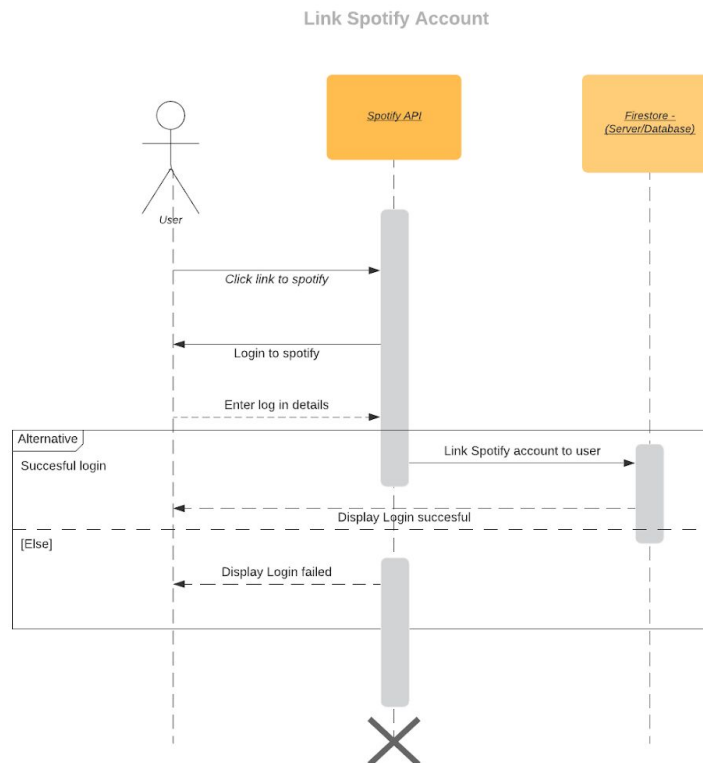
Edit account settings

- The user will access account settings by clicking on the respective button
- A list of options to change (Eg. privacy, avatar and deactivation) will be provided
- The user will then input their change to the desired setting. This request will be passed to the Database of users.
- This will involve the necessary error checking to approve or disprove the change.
- If successful, the change will be made and the user notified of the successful change.
- If unsuccessful, the change will not occur and the user notified of the failure in change and asked to retry.



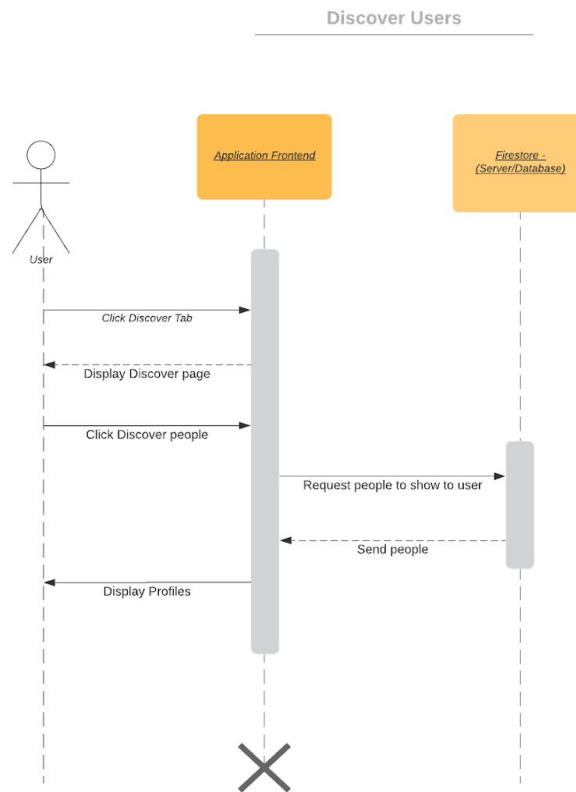
Import details from other platforms

- User clicks “link to platform” and application sends a request to link to platform api
- The platform api returns a request to login and user is prompted by application to enter login details
- User inputs login details which the application sends to the platform api
- The platform API:
 - On success, sends a link request to firebase database which is then accepted and firebase returns a login successful to the application which is displayed to the user
 - On failure, returns a login failed to the application which is displayed to the user



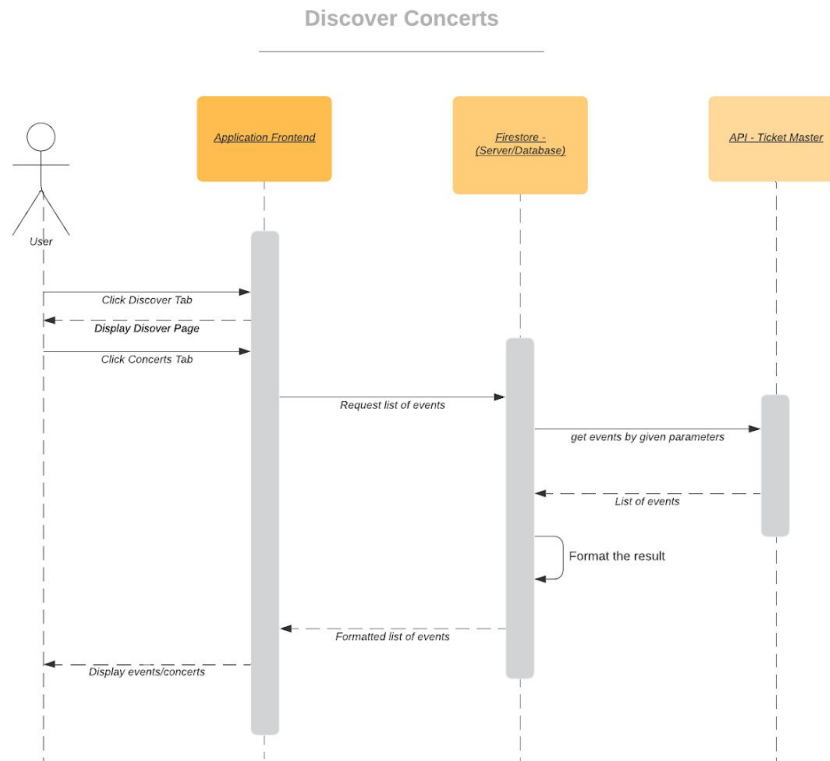
Discover People

- From anywhere on the site a user can press the discover tab
- The user will then see the discover page
- On the discover page the user can select to specifically discover people
- The server will then fetch other users from the database which will match well for the user 'discovering'
- These profiles will be displayed and the user can select them



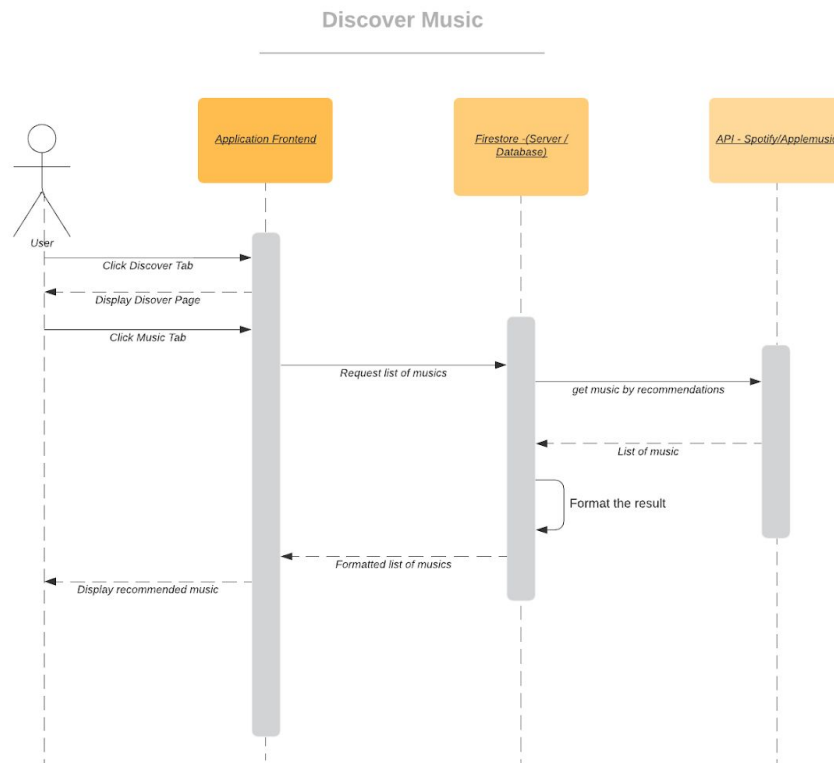
Discover Concerts

- From anywhere on the site a user can press the discover tab
- The user will then see the discover page
- On the discover page the user can select to specifically discover concerts
- The server will then request list of concerts from ticketmaster api
- These concerts then will be formatted and it will be fetched by the server
- These concerts will be displayed and the user can select them



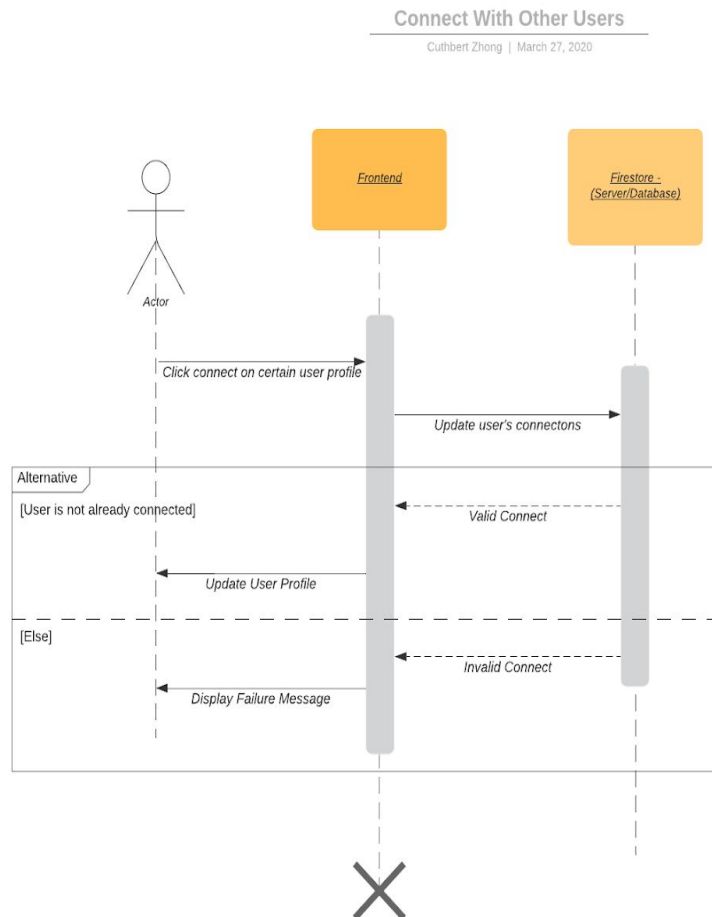
Discover Music

- From anywhere on the site a user can press the discover tab
- The user will then see the discover page
- On the discover page the user can select to specifically discover music
- The server will then request list of music from other music platform api
- These music tracks then will be formatted and it will be fetched by the server
- These music tracks will be displayed and the user can select them



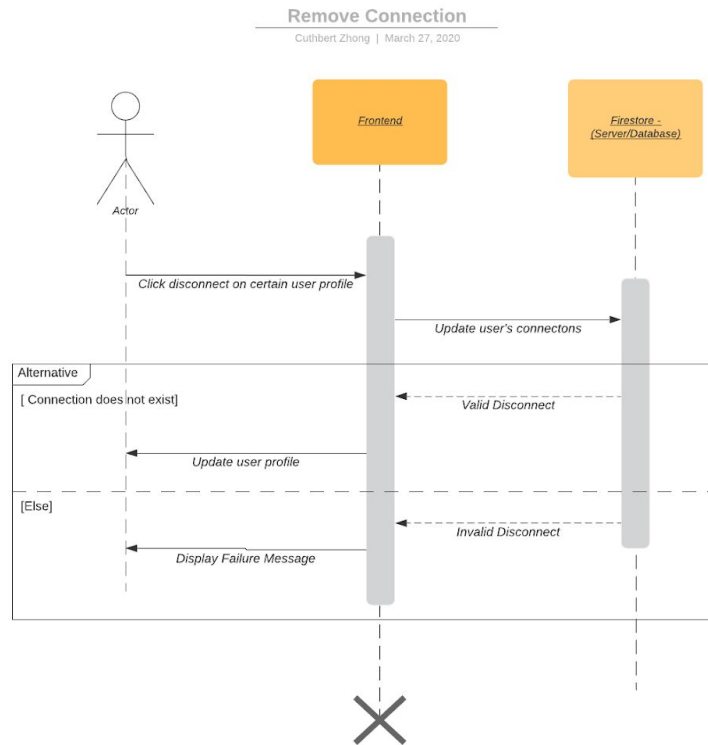
Connect with other users

- When viewing another user's profile, a user can click a button to connect with that user. When that button is clicked, the frontend sends a query to validate the new connection.
- If that connection is valid, then the user is added to the list of connections and if not valid, an error prompt is displayed.
 - Unsuccessful connects may occur from concurrency issues when the webpage not automatically updating with changes in the backend server



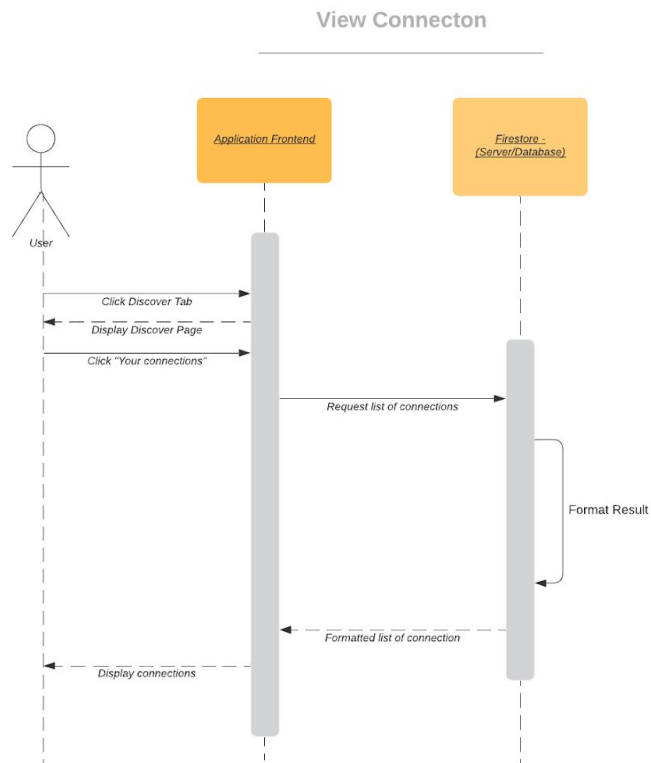
Remove connections

- When viewing a user's profile that is a connection, a user can click the disconnect button.
- When clicked, the frontend validates the disconnect request with the backend server and if successful, the user is removed from the list of users' connections.
 - Unsuccessful requests may arise from concurrency issues where the website does not automatically react to changes in the backend database.



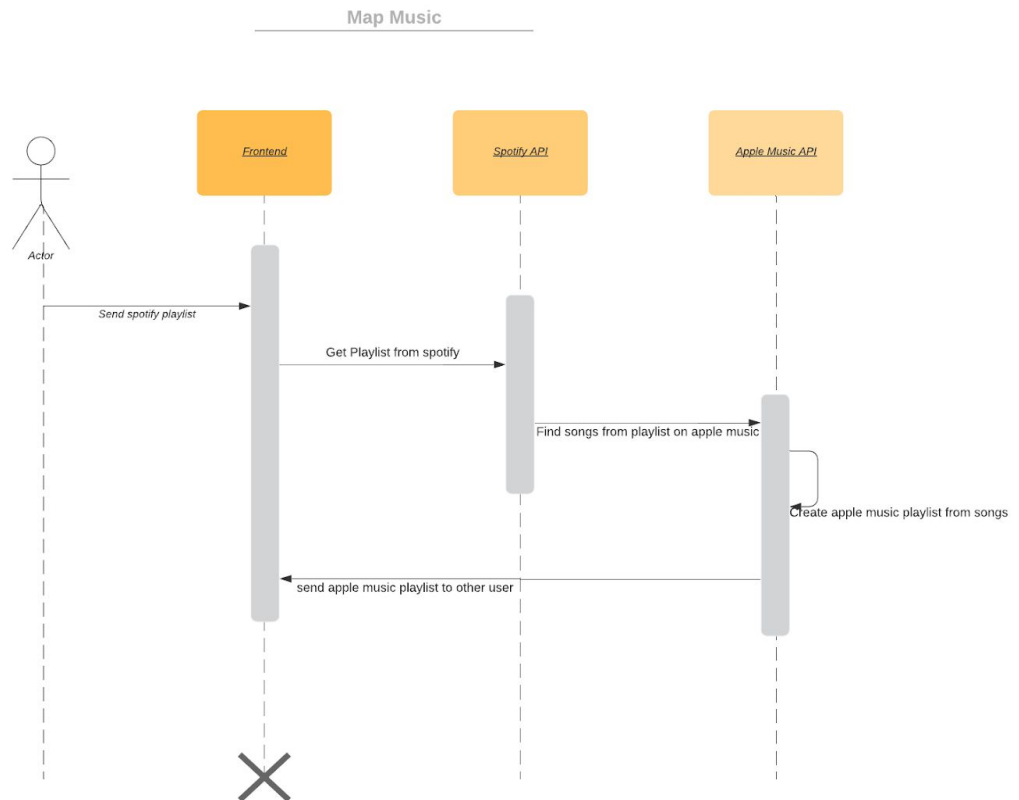
View existing connections

- From anywhere on the site a user can press the discover tab
- The user will then see the discover page
- On the discover page the user can select to specifically “Your Connection”
- The server will then fetch connected users from the database
- These profiles will be displayed and the user can select them



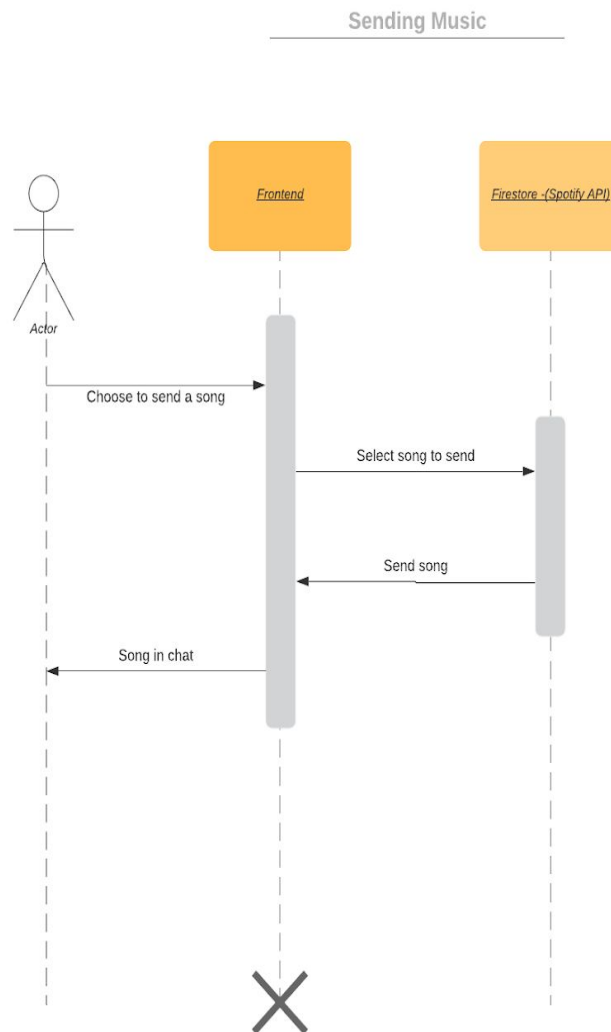
Map music tracks

- When in a chat with someone a user can send a playlist from a music platform (eg spotify) and ask the application to 'map' it
- The spotify API will then give that playlist to the server
- Using the apple music API the server will create a playlist in apple music with the same songs (where possible)



Send Tracks

- When in chat with someone a user can send a track
- They will choose the track from a music application API
- The API will then add the song to the chat and the other user will be able to listen to it



Receive music tracks

- From anywhere on the site a user can press the discover tab
- The user will then see the discover page
- On the discover page the user can select to specifically “shared with me”
- The server will then fetch music tracks that have been shared to the user's account from the database.
- These music tracks will be displayed and the user can select them

