# **Discotheque**

**An appified dancefloor.**

# **DEFINITION**

## **Name**

Discotheque

<https://github.com/CSUMB-SP17-CST438/project3-h1-jocruz-tha-pking-cyates>

## **Team**

* Joshua **Ryan** Cruz
  + Email: [jocruz@csumb.edu](mailto:jocruz@csumb.edu)
  + Github: <https://github.com/jcrzry>
  + Slack: @jocruz
* Tommy Ha
* Email: [tha@csumb.edu](mailto:tha@csumb.edu)
  + Github: <https://github.com/>
  + Slack: @tvanha01
  + Carsen Yates
  + Email: [cyates@csumb.edu](mailto:cyates@csumb.edu)
  + Github:<https://github.com/Fatburger3>
  + Slack: @cyates
* Peter King
  + Email: [pking@csumb.edu](mailto:pking@csumb.edu)
  + Github: https://github.com/
  + Slack: @pking

## **Deliverables**

We aim to deliver a real-time silent disco Android application, where users will have the ability to start a silent disco with their friends, or even join random silent discos already occurring. The application will allow users to authenticate via [SoundCloud](https://soundcloud.com/), and play music in a turn-based cycle via SoundCloud’s API right in the application. In addition, users will be able to share their thoughts in a real-time integrated chat.

**Motivation**

We have all listened to music, but not all of us have participated in a silent disco. This application will allow everyone to join in on the fun and be in sync with one another. There aren’t a lot of apps out there that are on the forefront of silent disco, so we’re going to give it a try.

## **Approach**

We’ll be building an Android application as the main client. For the backend we will continue to use Flask as a server and Postgresql for persistence. To communicate with the client we will be using [socket-io-client-android](https://github.com/socketio/socket.io-client-java) in conjunction with Flask-socketio to build upon what we have already learned to implement. In order to stream music, we will be using the Android Media Player API in conjunction with audio sourced via the SoundCloud API. The app will also allow users to authenticate via SoundCloud, Facebook, Google or create a new local account.

**Dependencies and risks**

Client/Server interactions are new to most of us, as we have always seemingly focused on client side coding in our classes. Our app will depend highly on our server host-reliability (heroku? Splash page and server?). In addition, the application will be highly dependent on the music available on SoundCloud.

# **USER STORIES**

## **As a user, I should be able to invite friends to join my silent disco floor.**

* Invite button and share buttons are always on the screen and visible when on a listening floor.
* When I tap the invite button, a pop up allows me to choose which friends to invite.
* When I press ‘send invite’, pop-up window disappears and invite message is sent to selected friends.
* A ‘toast’ appears notifying me that the invitation has been sent.

## **As a user, I should be able to use the app to send users on the same floor messages.**

* I should be able to click on the user list and select a user to send a message to.
* I should only be allowed to send messages to users in the room
* When i tap on ‘send message’ the text from the text field should disappear.
* âA ‘toast’ appears notifying me that the message has been sent.

## **As a user, I should be able to create my own silent disco floor.**

* At home screen there is a ‘+’ button that navigates to a new window
* The new window allows me to name my floor
* There is also an option to select a genre for the floor
* There is an option to make the floor public
* Before launching the floor there is a confirmation message
* A ‘toast’ notifies me that the floor has been created

## **As a user, I should be able to pick a song to play when my turn comes up.**

* I am notified that it is my turn to pick a song before the current song ends.
* There is a list of song titles to pick from.
* There is an option to auto-select a song if i don’t feel like picking a song.
* The song i selected is added to the queue and played next.
* A ‘toast’ notifies me that the song has been added to the queue.

## **As a user, I should always be able to view the player and track information when on a silent disco floor.**

* I should be able to see the player at all times when on a silent disco floor
* Track information should always be present and visible when i am listening.

## **As a user, I should be able to have music playing seamlessly.**

* Music should continue playing in the background and when the screen is off just like a standard music player.
* When a song ends, another song should begin, whether or not a member of the floor has chosen a song or not.

## **As a user, I should be able to add other users to my friends list.**

* There should be an ‘add listener’ to friends button
* After tapping the button the user should be added to my friends list
* I get a notification informing me that I have added the user to my friends list.
* I get a notification informing me that a user has added me to their friends list.

## **As a user, I should be able to view profile info of myself and my friends.**

* When I tap on a user’s photo I should go to their profile.
* A profile should include the user’s channel, their favorite genres, and their friends list.

## **As a user, I should be able to edit my profile.**

* My profile should have a button that allows me to edit my profile info.
* I should be able to select what Genres I am interested in.
* I should be able to select what profile info my friends can see.

## **As a user, I should be able to remove a user from my friends list.**

* I should be able to click on the user list and select a user.
* After a long press on the user name, an alert box should appear asking if I want to remove the user from my friends list.
* A ‘toast’ notifies me that the user has been removed from my friends list after clicking ‘yes’.

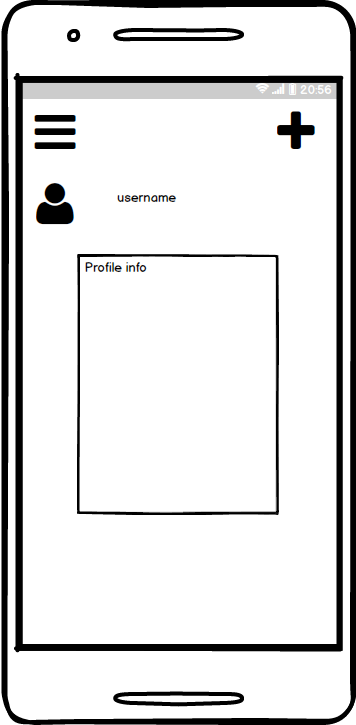
# 

# 

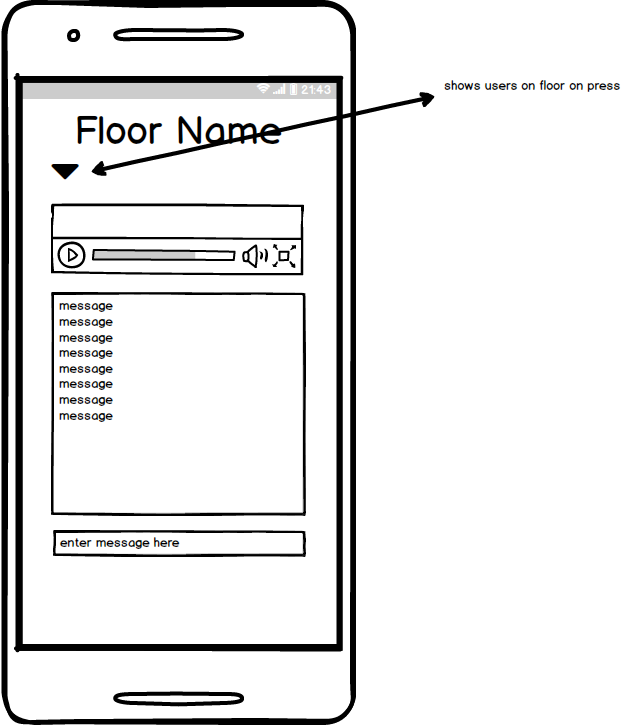
# 

# **MOCKS**

## **Main view with Discotheque Profile/start page**

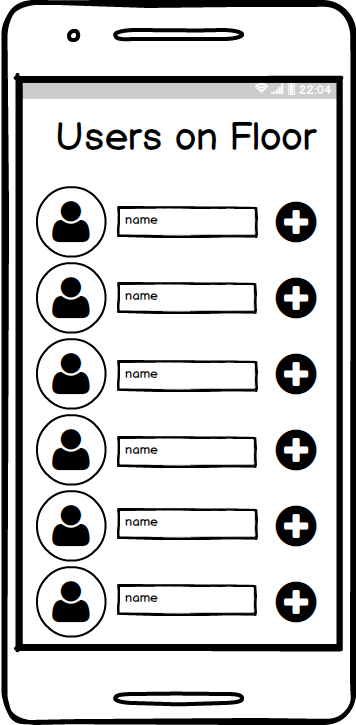


## **Floor view with player and chat window**



# 

## **Floor view with list of users with add buttons**



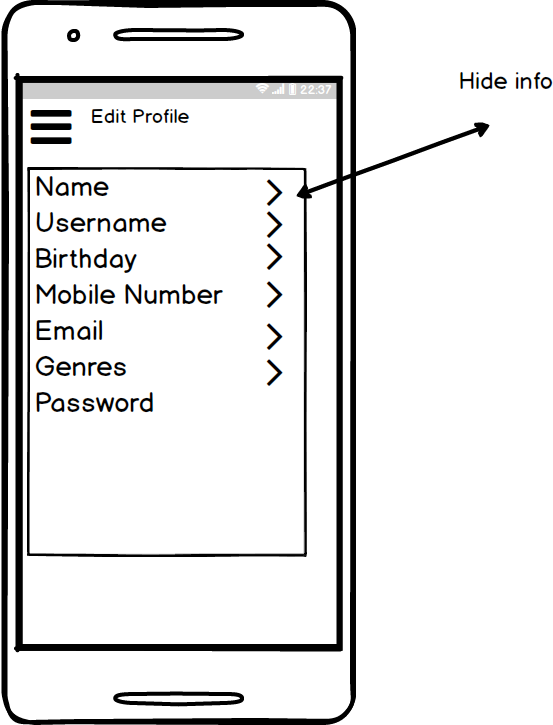
## 

## 

## 

## 

## **Edit profile view**



# **HANDIN MILESTONES**

## **Handin 1, due March 27th at 11:55pm**

The base of this app is the core streaming functionality, where users are able stream music in a room or ‘floor’ as we’ve been calling them. The floor should then stream the same music to other users also on the same floor. This will be the initial milestone and base functionality of the app. We would also like to have the basis of chats for each floor started. Since we have had no experience with streaming music this will be the priority, styling will be minimal.

## **Handin 2, due April 10th at 11:55pm**

Building on the streaming/broadcasting we completed in handin one, we’d like to to add styling to the application to make it look like the mocks. We would also like to add a social login feature, where users are given the ability to login with either Google, Facebook, or SoundCloud credentials or allow them to create a new account altogether. We’d also like to add persistence, basic user profiles, and the ability to add friends to a friends list.

## **Handin 3, due April 24th at 11:55pm**

To wrap things up, we want to focus on cleaning up any interaction bugs that are prominent. We would also like to finish styling the application for it to be more in line with our mocks and if time permits we would also like to create a splash page for our app.