

# **SW Engineering CSC648/848 Spring 2021**

## **SYNC**

### **Team 06**

|                                   |                          |                         |
|-----------------------------------|--------------------------|-------------------------|
| Team Lead                         | Rebecca Zumaeta          | rzumaeta@mail.sfsu.edu  |
| Front End Lead                    | Bryan Fetner             | bfetner@mail.sfsu.edu   |
| Back End Lead                     | Ashwini Managuli         | amanaguli@mail.sfsu.edu |
| Front End Member                  | Malcolm Angelo De Villar | mdevillar@mail.sfsu.edu |
| Front End Member                  | Hirva Patel              | hpatel11@mail.sfsu.edu  |
| Github Master and back end member | Vishakha Tyagi           | vtyagi@mail.sfsu.edu    |
| Back End Member                   | Luong Dang               | ldang2@mail.sfsu.edu    |

### **Milestone 3**

04/22/2021

### **History Table**

| Version | Date       | Notes |
|---------|------------|-------|
| M3V1    | 04/22/2021 |       |
| M2V2    | 04/08/2021 |       |
| M2V1    | 04/01/2021 |       |
| M1V2    | 03/09/2021 |       |
| M1V1    | 03/05/2021 |       |

## **Table of Contents**

|    |                                                            |    |
|----|------------------------------------------------------------|----|
| 1. | Data Definitions V3.....                                   | 3  |
| 2. | Functional Requirements V3.....                            | 6  |
| 3. | Wireframes Based on Mockup/Storyboard.....                 | 10 |
| 4. | High level database architecture and organization V2 ..... | 46 |
| 5. | High Level Diagrams V2 .....                               | 51 |
| 6. | List of contributions .....                                | 53 |

## 1. Data Definitions V2

1. **User:** a person who has a spotify account that would like to listen to music with others in real time on web application named ‘SYNC’
  - a. userId: Unique number given to each registered user on SYNC
  - b. spotifyId: A number given to registered user having spotify account
  - c. profilePic: A display picture of every registered user and they have the option to post a picture of themselves or not.
  - d. displayName: Name of the user with which they wants to be identified as on SYNC
2. **Profile:** It has the information describing the registered user.
  - a. userId: Unique number given to each registered user on SYNC
  - b. activity: It will show if the registered user is currently using SYNC or available online or not.
  - c. profilePhoto: A display picture of every registered user for which they have the option to post a picture of themselves or not.
  - d. status: It defines if the person is listening to songs in the room as a participant who joined the room or as a host who created the room for others to join.
3. **Participant:** Any registered user who is using the SYNC for listening to songs in the real time and is present in the room but did not create the room.
  - a. userId: Unique number given to each registered user on SYNC
4. **Host:** The registered user who either created a private or a public room and sends invites to others to join the room. Also, this user has more control of the room than other participants.
  - a. userId: Unique number given to each registered user on SYNC
5. **spotifyInfo:** The information of the users imported from the spotify.
  - a. authToken: key string for Spotify’s player
  - b. spotifyId: userID from Spotify’s API endpoint
  - c. userName: Name of the Spotify user.
  - d. userId: Unique number given to each registered user on SYNC
  - e. playlistListId: Every playlist on spotify has a number attached to it
6. **spotifyAPI**
  - a. Player
    - i. connectivity: If the player is properly connected to the internet and is able to play songs as per the request of registered users. Online, disconnected, or error
    - ii. currentSongTitle: The title of the song currently being played in the player.

- iii. progress : The minute at which the song in the player is playing.
- b. song
  - i. songTitle : The title of the song that can be searched or in queue.
  - ii. artist : The creator of the song associated with songTitle.
  - iii. imageUrl : The image that is associated with the album.
  - iv. genre : The category at which the song is considered in.
  - v. album : The album name and which the song belongs in.
- c. artists
  - i. artistName : The artist that is associated with the song.
  - ii. songs : The songs that are created by the artist.
  - iii. album : The album created by the artist.
- d. genres
  - i. imageUrl : Image that represents the genre.
  - ii. genreName : The name of the genre in which a song belongs to.
  - iii. description : Describes what the genre is and gives a summary of what will be expected.
  - iv. topPlaylist : Shows the top playlist in the genre.

## 7. rooms

- a. roomType : It determines the kind of room the created room is.
  - i. Public: available for all users through search and recommended results
  - ii. Private: only available to other users through the sharing of the room id
  - iii. Communal: A perpetual room.
- b. roomId : This is the unique room identifier.
- c. roomName : This is the room name in which the user set the room name to be.
- d. description : A description of the room in which the user decides to put.
- e. currentSong : It shows what the current song is playing in the room. This is also shown in the previews of the rooms.
- f. roomHost : It shows who created the room.
  - i. userId : This is the unique identifier of the user.
- g. password : This is the password set by the roomHost for private rooms.
- h. status : Room status can either be Open or Closed, depending on the activity of the room.
- i. maxMembers : This is the max limit number of users that can join in a room, which is specified by the roomHost.
- j. currentNumber : This is the current number of users that are in the room.
- 8. queue : The songs that are in queue in the room to be played and voted on.
  - a. roomId : Identifies where that song queue belongs in.

- b. songListId : This is the unique queue list identifier.
- 9. **songList** : This shows the
  - a. songTitle : This shows the title of the songs in the songList
  - b. songId : This is the unique identifier of the songs in songList.
  - c. votesId : This is the unique identifier for the votes in the songList.
- 10. **votes** : This identifies what songs the users voted for to be played next in the room.
  - a. userId : This is the unique identifier for the user who voted for songs.
- 11. **chatSection** : This is the portion of the web application where it shows the users where a user can chat with.
  - a. tabId : This is the unique identifier for the chatSection to identify who the user is talking to, or what the current active tab is.
  - b. tabStatus : This identifies the current, active, or inactive chat tabs per user.
  - c. server : This will be the server that will handle all realtime chat interactions.

## **2. Functional Requirements V3**

### **Priority 1:**

#### **1. Unregistered Users**

- 1.1. Unregistered Users shall be able to log into their Spotify Premium.
- 1.2. Unregistered Users shall be able to access the homepage of the website.
- 1.4. Unregistered Users shall be able to access the FAQ of the website.
- 1.5. Unregistered Users shall be able to access the Contact page of the website.

#### **2. Registered Users**

- 2.7. Registered Users shall have a premium Spotify account.
- 2.8. Registered Users shall be able to login into their Spotify Premium.
- 2.9. Registered Users shall be able to listen to music in real time.
- 2.10. Registered Users shall be able to access the Homepage of the website.
- 2.12. Registered Users shall be able to access the FAQ of the website.
- 2.13. Registered Users shall be able to access the Contact page of the website.
- 2.26. Registered Users shall be able to logout.
- 2.27. Registered Users that create a room shall have the status of host.
- 2.28. Registered Users as hosts shall be able to name the room.
- 2.29. Registered Users as hosts shall be able to add songs to queue
- 2.31. Registered Users as hosts shall be able to pause currently playing songs.
- 2.35. Registered Users shall be able to create a “room” public
- 2.36. Registered Users shall be able to create a “room” private.
- 2.37. Registered Users shall be able to search a public room.
- 2.38. Registered Users shall be able to search a private room.
- 2.39. Registered Users shall be able to join a public room.
- 2.40. Registered Users shall be able to join a private room.
- 2.42. Registered Users shall be able to invite people to their room.
- 2.43. Registered Users who created a room shall be able to choose what song to play.
- 2.44. Registered Users shall be able to search for songs.
- 2.45. Registered Users shall be able to choose the next song to play in the room.
- 2.47. Registered Users that create a room shall be able to close the room.
- 2.48. Registered Users shall be able to chat in all room types.
- 2.49. Registered Users shall be able to chat with people who joined in their created room.
- 2.57 Registered Users shall be able to vote on songs to be played next in queue
- 2.58. Registered Users shall be able to leave rooms
- 2.59. Registered Users shall be able to vote

### **3. Administrators**

- 3.58. Administrators shall be able to reset user passwords.
- 3.61. Administrators shall be able to join all rooms without a password.
- 3.63. Administrators shall be able to ban users.
- 3.67. Administrators shall be able to delete rooms.
- 3.65. Administrators shall be able to send messages to all users.
- 3.64. Administrators shall be able to leave messages regarding reasons for user bans.
- 3.66 Administrators shall be invisible (to other users in a room) when entering a room.

### **4. Rooms**

- 4.69. Rooms shall display the room name.
- 4.70. Rooms shall display if they are public or private.
- 4.71. Rooms shall display the hostname.
- 4.76. Rooms shall display the current song.
- 4.77. Rooms shall display the song queue.
- 4.78. Rooms shall display genre.
- 4.79. Rooms shall display chat.
- 3.80. Rooms shall display who commented in the chat.
- 4.81. Rooms shall have the voting system.
- 4.82. Rooms shall shut down if there is no host in it.
- 4.85. Rooms shall play music

### **5. Website**

- 5.82. Website shall display username.
- 5.83. Website shall display the user's profile picture.
- 5.84. Website shall let users resume activity using cookies
- 5.89 Website shall keep logged in users' info by cookies.
- 5.90 Website shall let users register.
- 5.91 Website shall let users login.
- 5.92 Website shall let users reset their password.
- 5.93. Website shall display the website's contact info.
- 5.94. Website shall give the option to continue or cancel creation of the room.
- 5.95. Website shall allow user to send invitation link to rooms
- 5.96. Website shall allow users to search for a room.
- 5.97. Website shall allow users to join a room.
- 5.98. Website shall allow users to see public rooms' info.
- 5.99. Website shall display available public rooms.
- 5.100. Website shall play music on Spotify.
- 5.101. Website shall support popular browsers

## **Priority 2:**

### **1. Unregistered Users**

### **2. Registered Users**

- 2.20. Registered Users shall be able to change status offline.
- 2.21. Registered Users shall be able to change status online.
- 2.29. Registered Users as hosts shall be able to generate playlists.
- 2.33. Registered Users as hosts shall be able to set a limit to the number of users in the room.
- 2.34. Registered Users as host of a room shall be able to kick a user out of the room.
- 2.50. Registered Users shall be able to create group chat.
- 2.51. Registered Users shall be able to text in group chat.
- 2.52. Registered Users shall be able to add friends.
- 2.53. Registered Users shall be able to DM a friend.
- 2.54. Registered Users shall be able to see their friends list.
- 2.55. Registered Users shall be able to remove friends.
- 2.56. Registered Users shall be able to block friends.
- 2.72. Rooms shall display a description of the room.
- 2.73. Rooms shall be up during its dedicated time set.
- 2.74. Rooms shall display the number of users in the room.
- 2.75. Rooms shall list all users in the room.

### **3. Administrators**

- 3.57. Administrators shall be able to change SYNC usernames.
- 3.59. Administrators shall be able to change user friends list
- 3.60. Administrators shall be able to open rooms with specific music selections.
- 3.62. Administrators shall be able to review user comments.
- 3.65. Administrators shall be able to ban songs.
- 3.66. Administrators shall be able to ban podcasts.
- 3.68. Administrators shall be able to delete accounts permanently.
- 3.69. Administrators shall be able to control the communal rooms.

### **5. Website**

- 5.96. Website shall display DMs
- 5.97. Website shall show the list of added friends.

## **Priority 3:**

### **1. Unregistered Users**

1.6. Unregistered Users shall be able to get access to technical support.

### **2. Registered users**

- 2.14. Registered Users shall be able to get access to technical support.
- 2.17. Registered Users shall be able to change their SYNC usernames.
- 2.22. Registered Users shall be able to report other users for misconduct.
- 2.23. Registered Users shall be able to share a room link though social media.
- 2.24. Registered Users shall be able to share link through Direct Message
- 2.25. Registered Users shall be able to share link through email
- 2.30. Registered Users as hosts shall be able to control the music queue.
- 2.41. Registered Users shall be able to join a random public room.
- 2.46. Registered Users shall be able to change the background theme of the room.

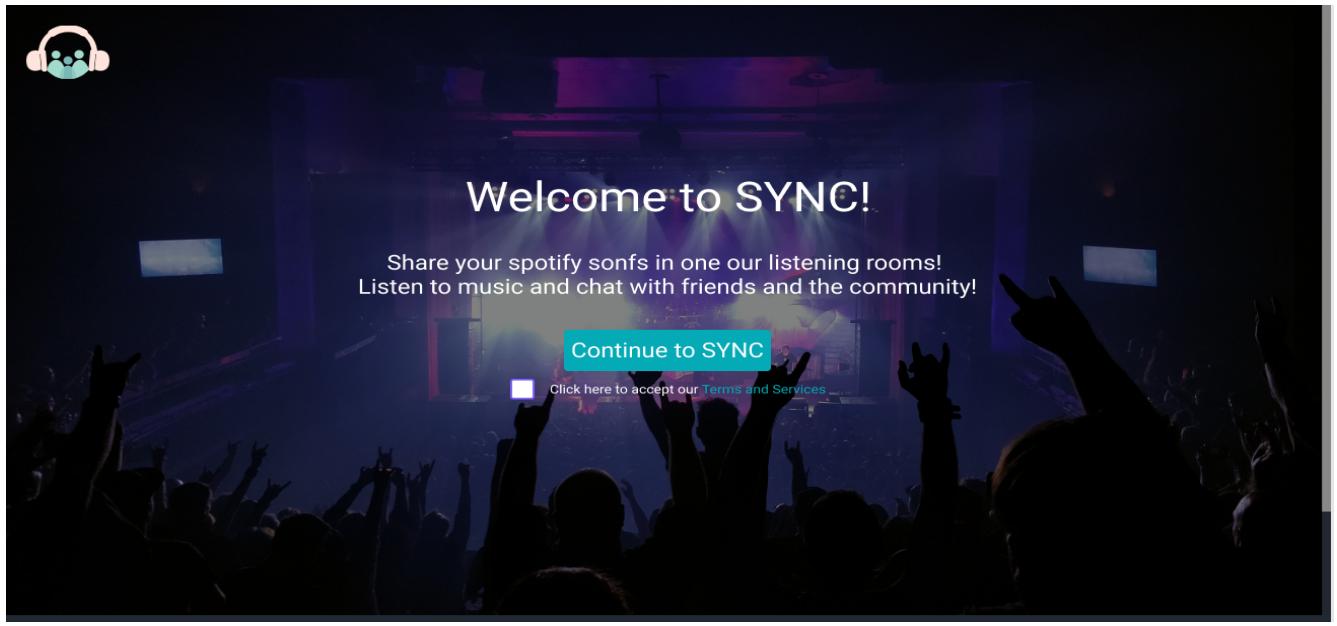
### **5. Website**

- 5.87. Website shall have a technical support page.
- 5.91. Website shall show invites.
- 5.92. Website shall send notifications
- 5.93. Website shall show the user's most preferred genres/artists.
- 5.98. Website shall display the number of SYNC friends the user has
- 5.100. Website shall show the history of rooms the users have been in

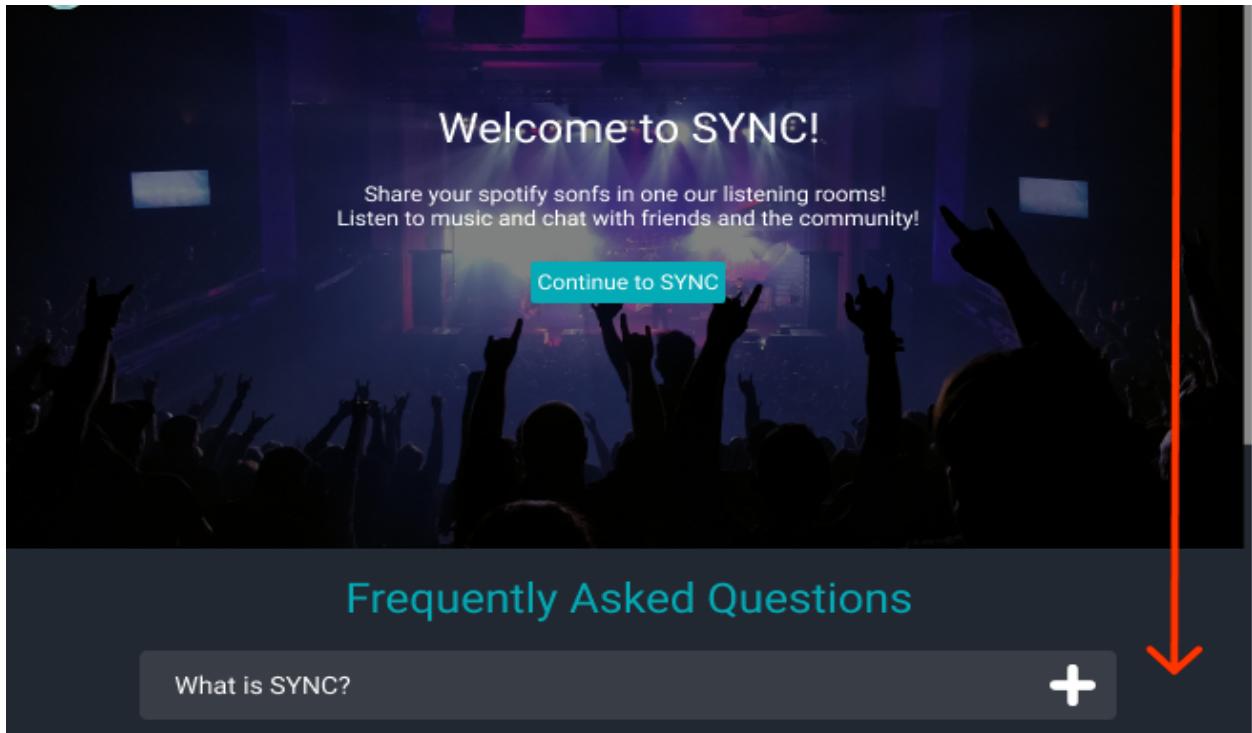
### 3. Wireframes Based on Mockups/Storyboards V2

#### 1. View Frequently Asked Questions

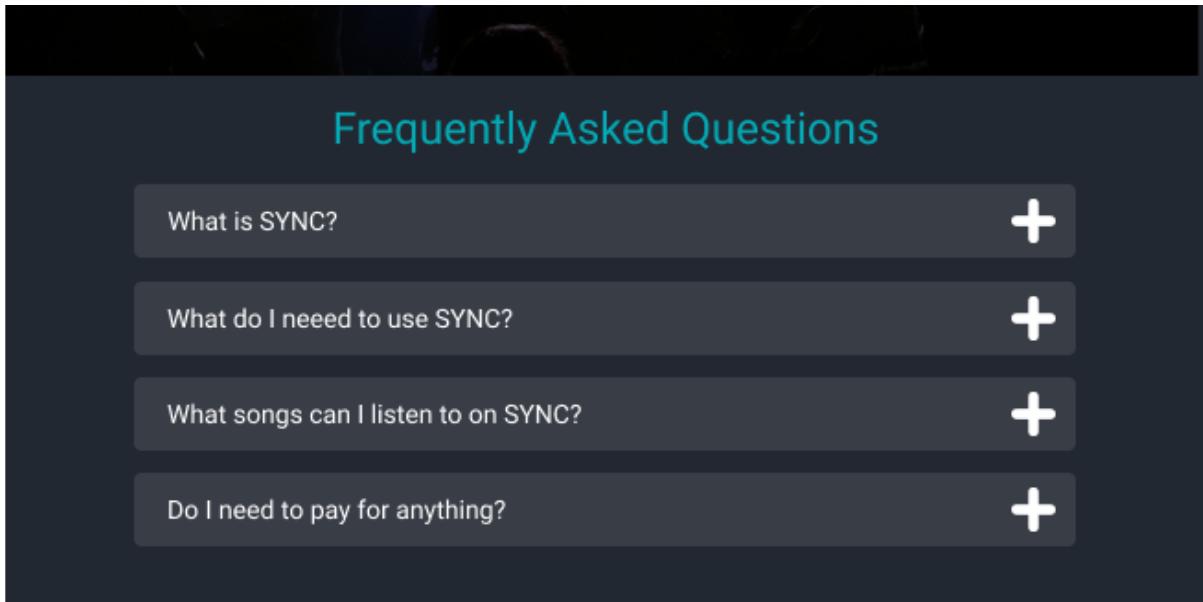
- Navigate to <http://3.17.66.0:3000/> it will bring you to this page



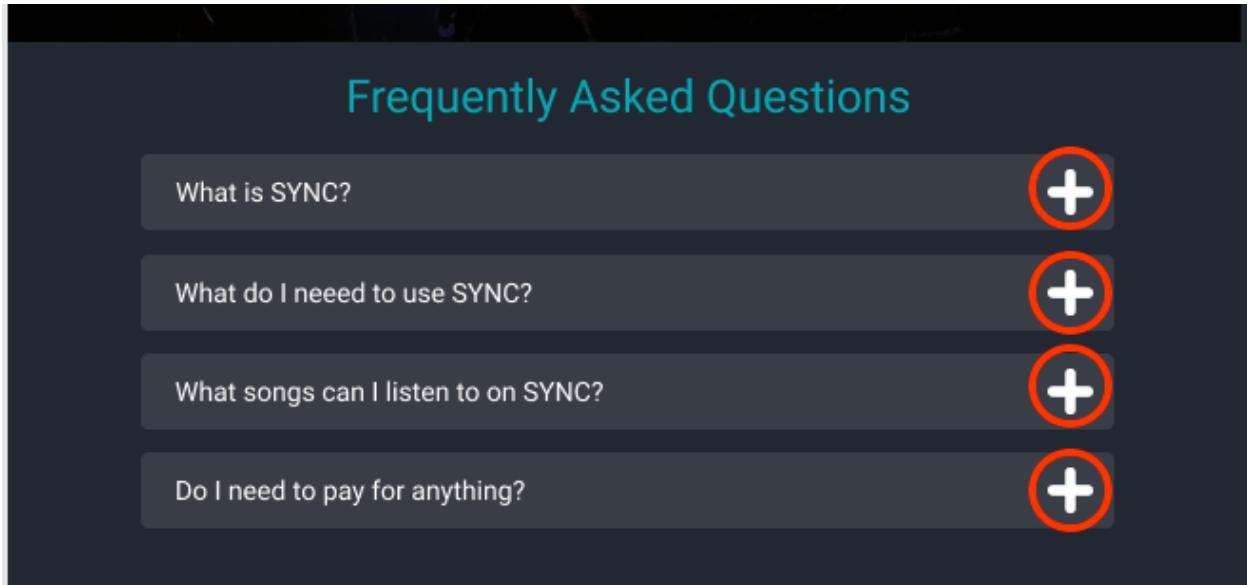
- Scroll down the landing page



- c. You will see a list of Frequently Asked Questions it will look like this



- d. Select any of the '+' plus icons to view the answers the questions depicted on the left



- e. Once one of the ‘+’ plus icons are selected you should be able to see the answer just as it is depicted below.

## Frequently Asked Questions

What is SYNC? X

SYNC is a music streaming platform linked with Spotify. You can listen and talk to friends and/ or the public to songs

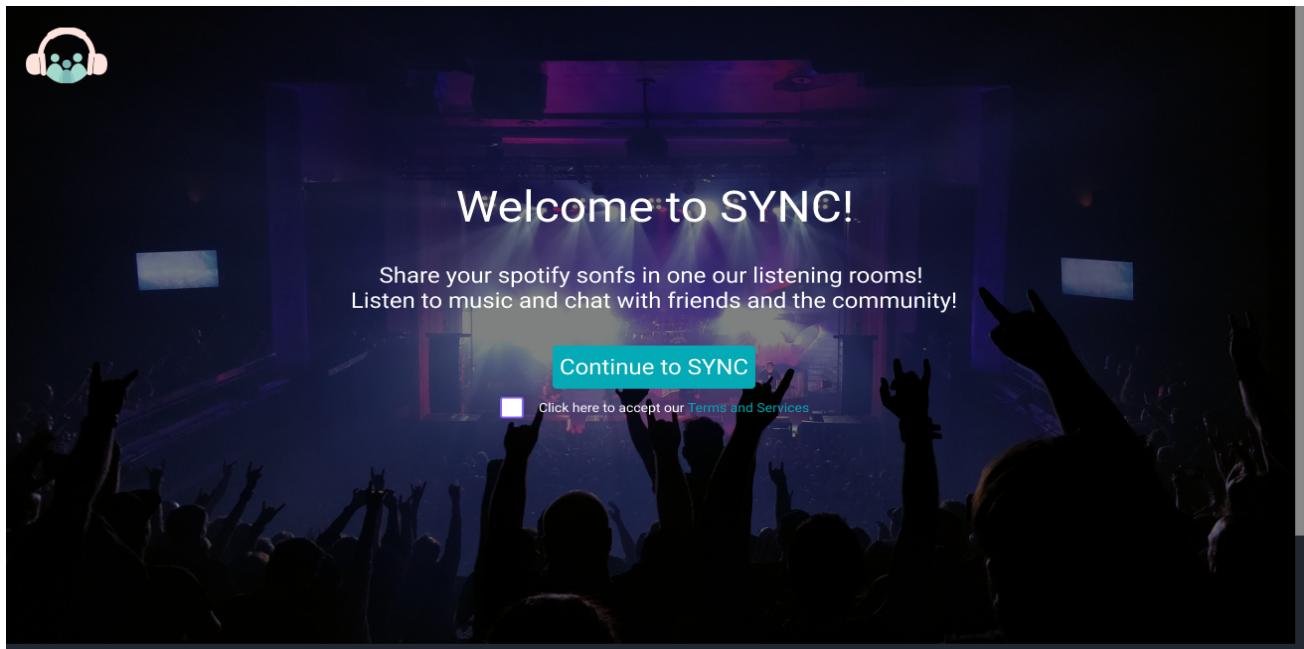
What do I need to use SYNC? +

What songs can I listen to on SYNC? +

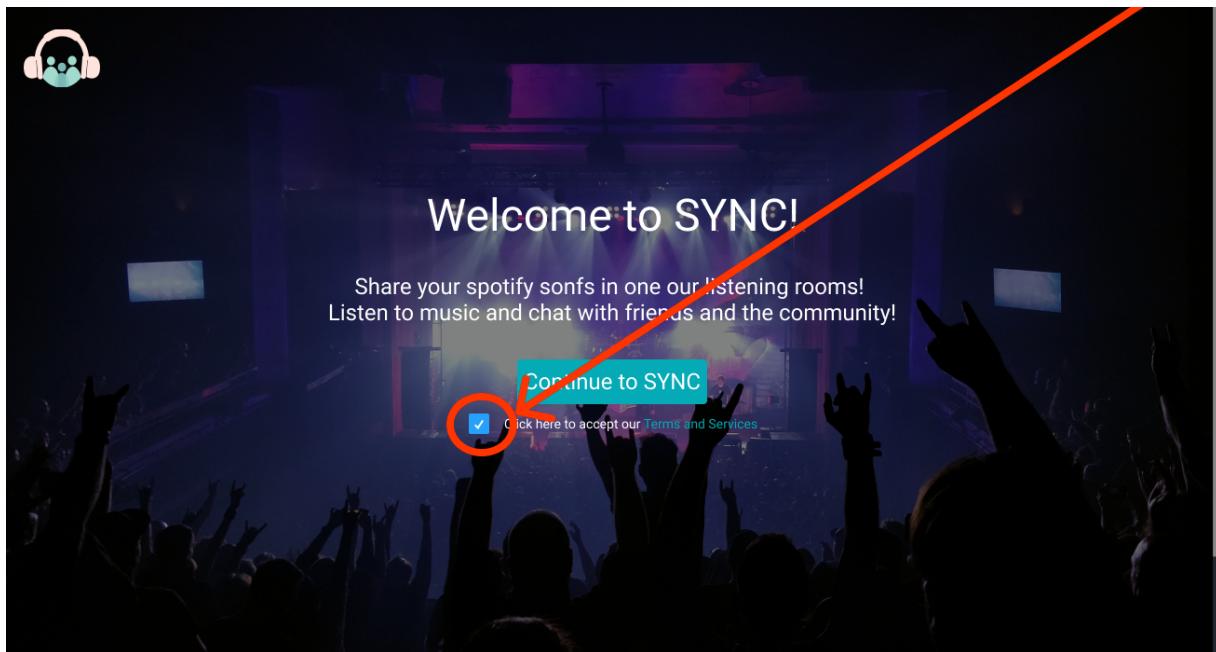
Do I need to pay for anything? +

## 2. Continue into SYNC and check off Terms of Services

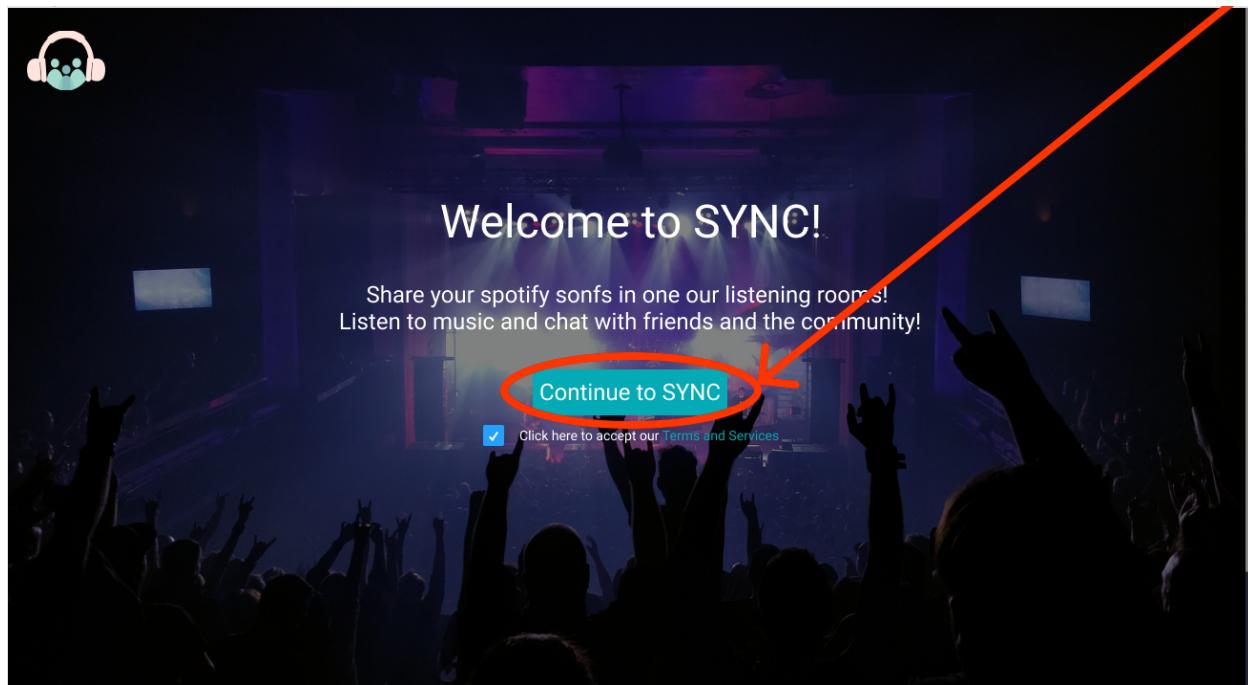
- a. Navigate to <http://3.17.66.0:3000/> it will bring you to this page



- b. To continue to the home page you must check off the terms and services. To do so you must check off the check box button located below the 'Continue to SYNC' button



- c. Once that is checked select the ‘Continue to SYNC’ button to be redirected to SYNC’s home page.

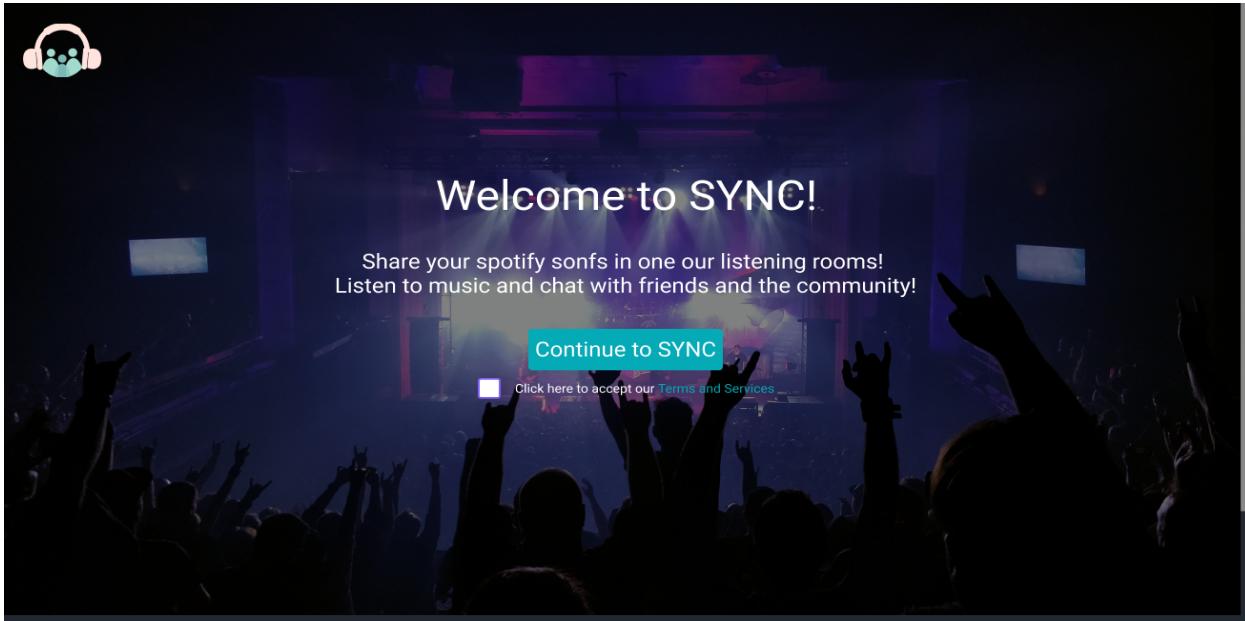


- d. Directly after pressing the ‘Continue to SYNC’ button you will be redirected to the home page of SYNC!

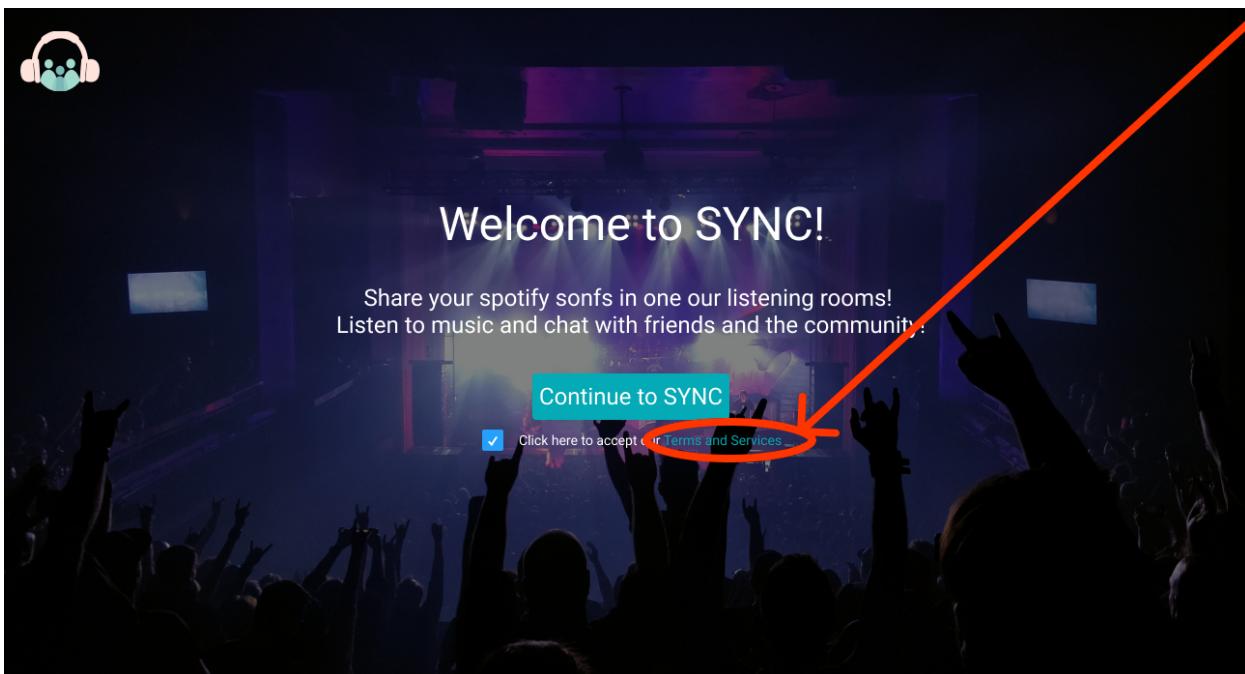
A screenshot of the SYNC! website's home page. At the top, there is a navigation bar with the SYNC! logo, "create", and "join" buttons. On the right, it shows "signed in as, User Name v". The main area has a dark background with white text. It says "Search for a room to join, or create your own room!" and features two blue buttons: "Join a Room" and "Create a Room". Below this is a section titled "Recommended Rooms" with four cards. From left to right: 1. A photo of a dessert (ice cream) with the caption "Room Name: Room 1" and "Category: Dessert". 2. An illustration of cherries with the caption "Room Name: Room 2" and "Category: Classical". 3. A photo of a vertical garden with the caption "Room Name: Room 3" and "Category: Plants". 4. A photo of a jellyfish with the caption "Room Name: Mag's Room" and "Category: Dark".

If you would like to read the Terms and Services, you can select the link provided on the landing page.

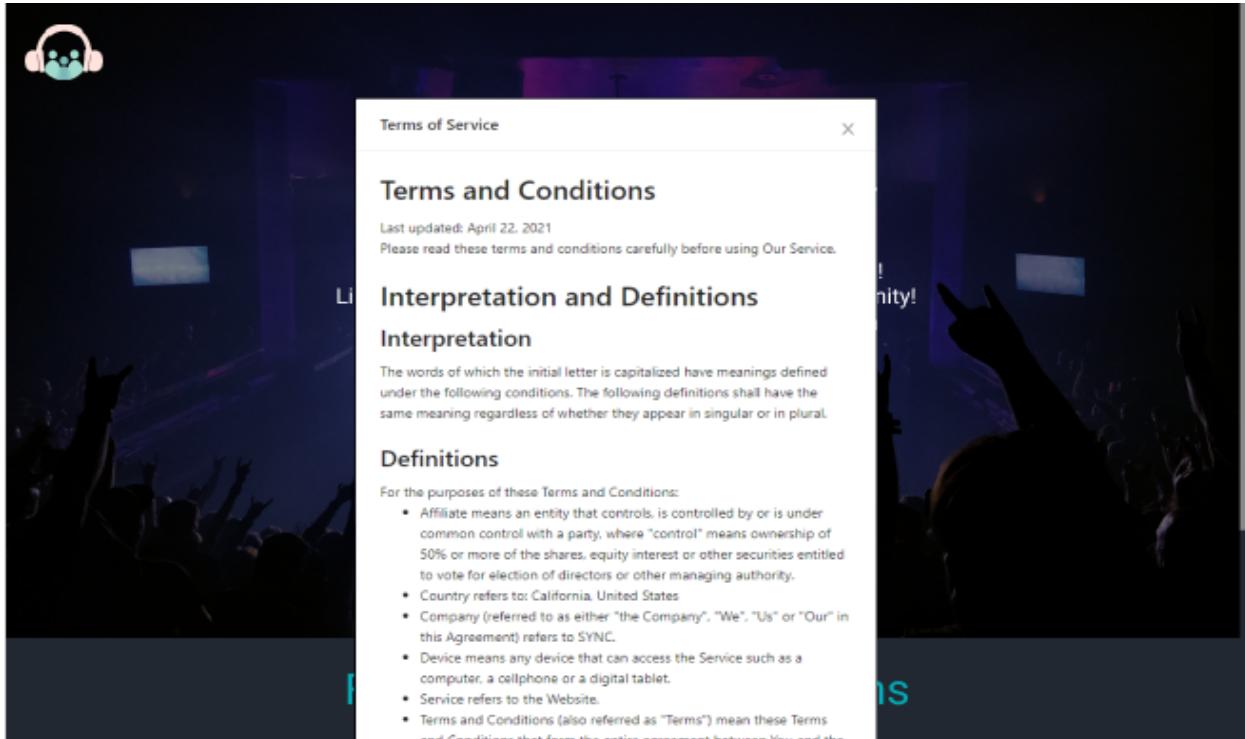
- e. Navigate to <http://3.17.66.0:3000/> it will bring you to this page



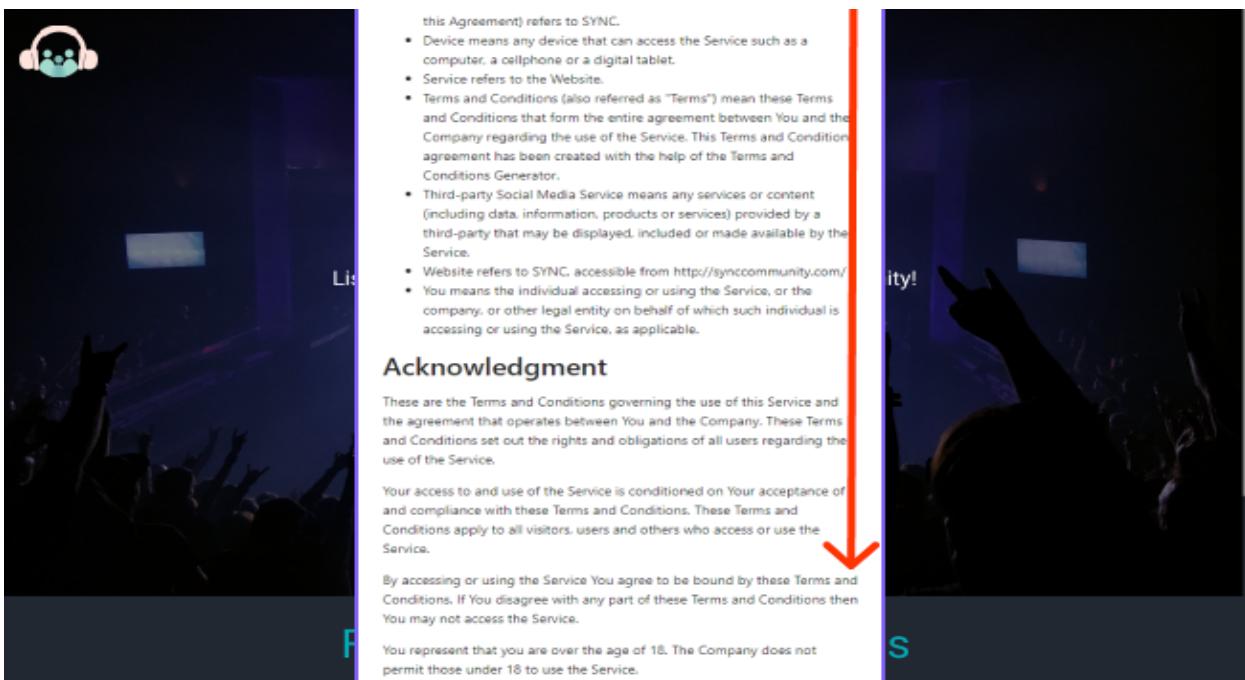
- f. Select the 'Terms and Services' link located under the 'Contine to SYNC' button



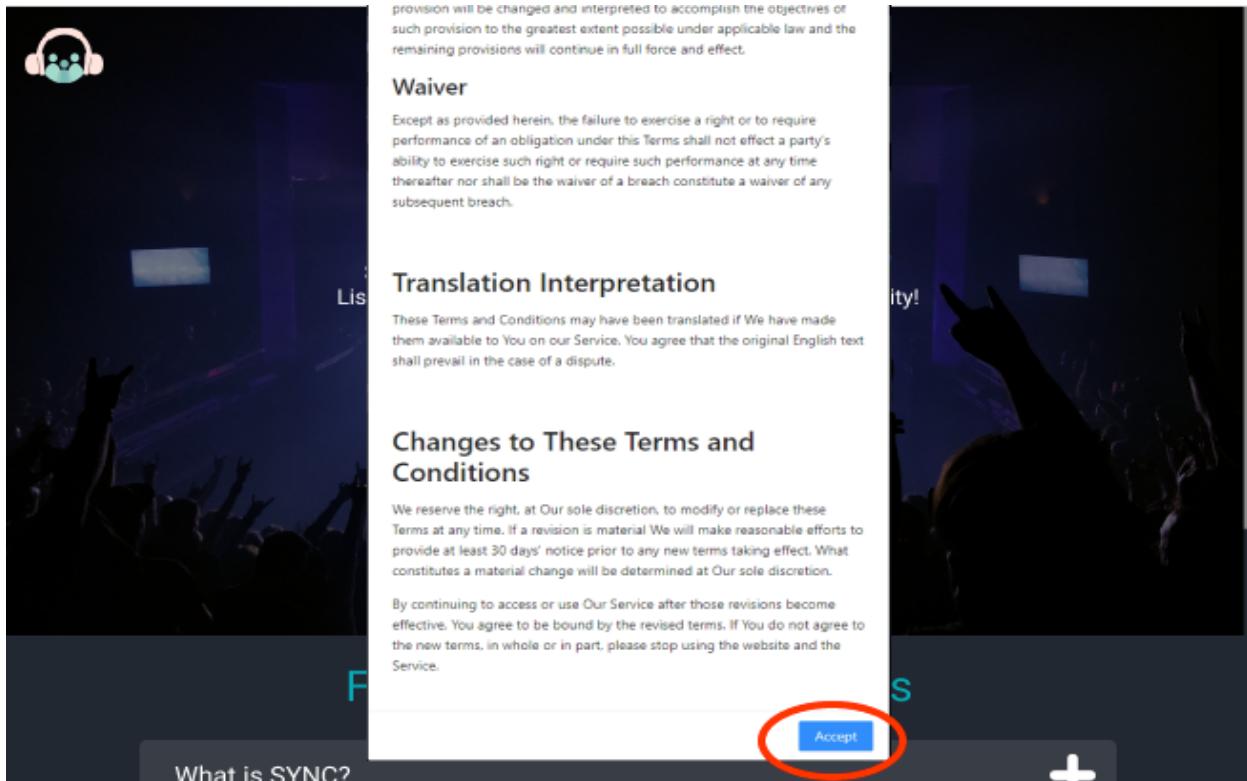
g. selected a pop up window will appear. It will look like this:



You will need to read and scroll through the Terms and Conditions to view all its content

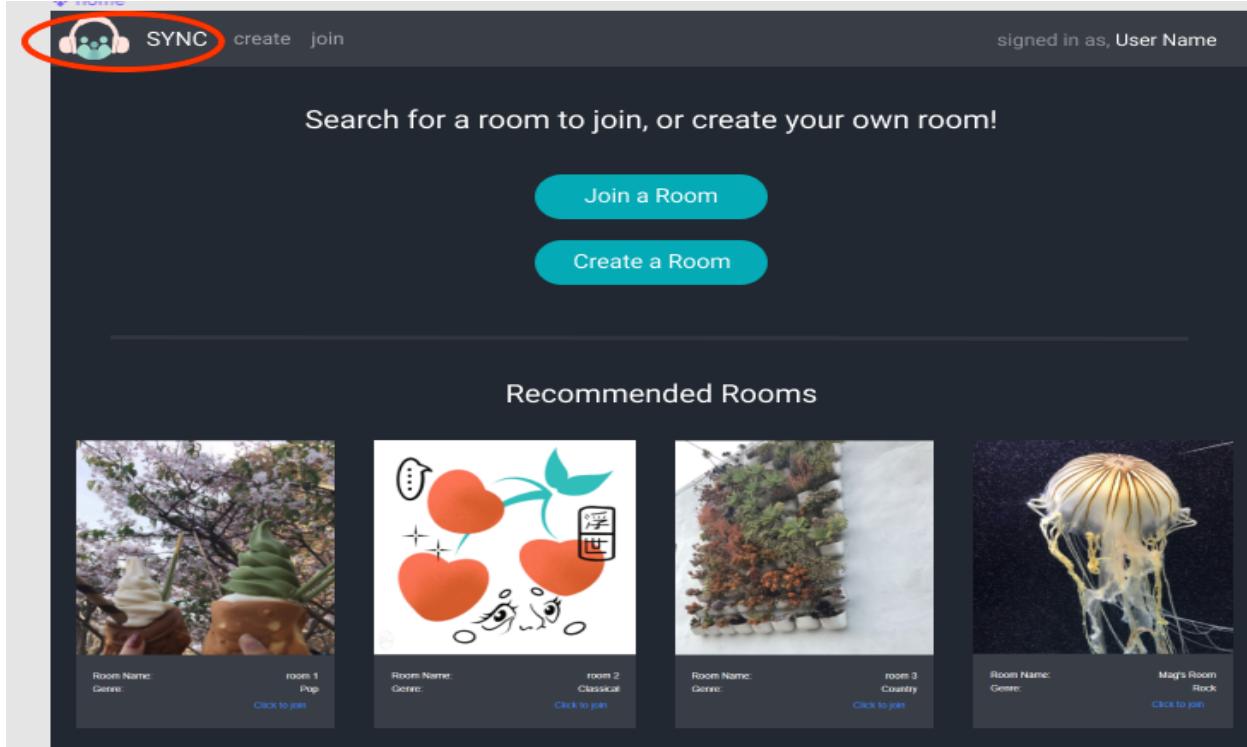


- h. After scrolling and reading all the terms and conditions, select the accept button

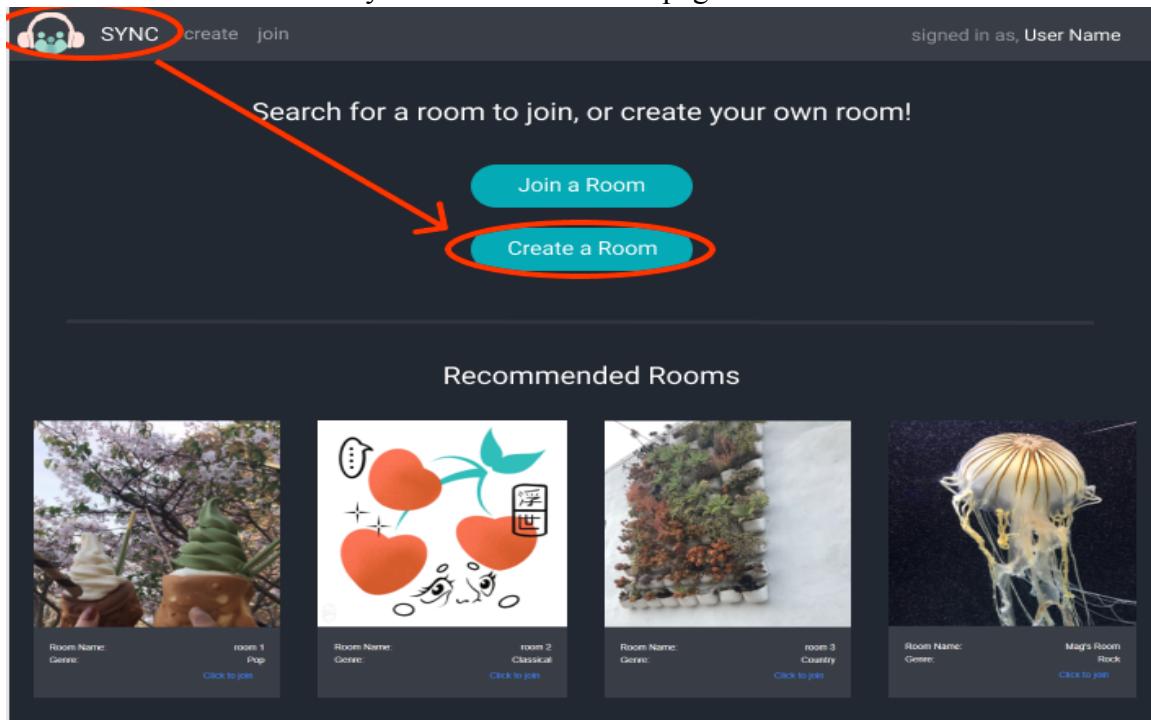


### 3. Create a room

- a. There are a couple places to create rooms. I will guide you through both locations
  - i. Navigate to <http://3.17.66.0:3000/Home> by clicking this link or if already within site, select the SYNC logo at the top left



- ii. Once you arrive to the home page select the 'create a room' button



iii. A drop down will appear with the necessary options to create a room.

The screenshot shows the SYNC application interface. At the top, there is a user icon, the word "SYNC", and navigation links for "create" and "join". On the right, it says "signed in as, User Name". Below this, a header reads "Search for a room to join, or create your own room!". There are two buttons: "Join a Room" (in white) and "Create a Room" (highlighted in teal). A large red oval encircles the central "Create a Room" form. This form contains fields for "Roomname" (with placeholder "e.g. Bill's Room of Splendor") and "Genre" (with placeholder "Select genre"). It also includes radio buttons for "Public Room" (selected) and "Private Room", and a checkbox for accepting "Terms of Services". A "Submit" button is at the bottom. Below this section, the heading "Recommended Rooms" is displayed, followed by four cards, each showing a thumbnail image and room details:

- room 1** Pop  
Room Name:  
Genre:  
[Click to join](#)
- room 2** Classical  
Room Name:  
Genre:  
[Click to join](#)
- room 3** Country  
Room Name:  
Genre:  
[Click to join](#)
- Maggie's Room** Rock  
Room Name:  
Genre:  
[Click to join](#)

The **second** way to create a room is as such:

I. Navigate to <http://3.17.66.0:3000/Createpage> by selecting provided a link or ( if already in site) selecting the ‘create’ button in the nav bar located at the top of every page within the site. It will take you to the crea a room page

Search for a room to join, or create your own room!

Join a Room

Create a Room

Recommended Rooms

Room Name: Generic room 1 Pop Click to join

Room Name: Generic room 2 Classical Click to join

Room Name: Generic room 3 Country Click to join

Room Name: Generic room 4 Rock Click to join

Roomname: e.g. Bill's Room of Splendor

Genre: Select genre

Public Room Private Room

Click here to accept our Terms of Services

Submit

- b. The next thing is to fill the text boxes.
  - i. Type in a name for the room you are creating. This is uniquely yours.

Create your own room!

\* Roomname: My Cool Room!

\* Genre: Select genre

Public Room Private Room

Click here to accept our Terms of Services

Submit

- ii. Pick a genre: once you select the text box a drop down menu appears, allowing you to select genres such as pop or classical.

SYNC create join signed in as, User Name

Create your own room!

\* Roomname: My Cool Room!

\* Genre: Pop

Rock  
Pop  
Classical  
Country

Submit

- iii. Select 'Public' or 'Private' for the room type. This allows you to chose if all users can see the room or only to those you allow to see

SYNC create join signed in as, User Name

Create your own room!

\* Roomname: My Cool Room!

\* Genre: Pop

Public Room  Private Room

Click here to accept our Terms of Services

Submit

- iv. Check off ‘Accept Terms of Services’ by selecting the box beside the text that reads ‘Click here to accept our Terms of Services’

The screenshot shows a dark-themed interface for creating a room. At the top, there's a navigation bar with a user icon, the word 'SYNC', and links for 'create' and 'join'. On the right, it says 'signed in as, User Name'. Below this is a section titled 'Create your own room!'. A large, semi-transparent rectangular overlay covers the middle of the screen. Inside this overlay, there are several input fields and controls:

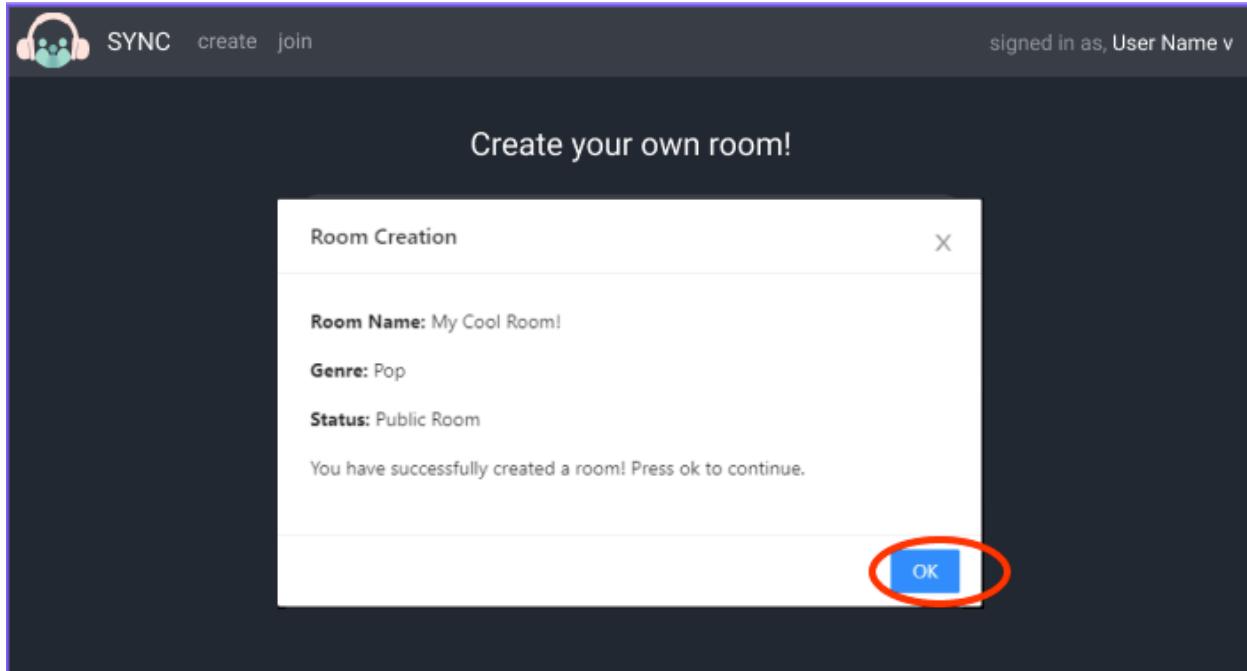
- A text input field labeled 'Roomname:' containing 'My Cool Room!'
- A text input field labeled 'Genre:' containing 'Pop'
- A radio button group for 'Room Type': 'Public Room' (selected, indicated by a blue outline) and 'Private Room' (unselected, indicated by a grey outline)
- A checkbox labeled 'Click here to accept our Terms of Services' which is checked (indicated by a blue checkmark)
- A teal-colored 'Submit' button at the bottom of the overlay

A thick red arrow points from the left towards the 'Click here to accept our Terms of Services' checkbox.

- v. Select the ‘Submit’ button when you are satisfied with your selection

This screenshot is similar to the previous one, showing the 'Create your own room!' form. The overlay contains the same fields: Roomname ('My Cool Room!'), Genre ('Pop'), Room Type ('Public Room' selected), and the 'Click here to accept our Terms of Services' checkbox (checked). The 'Submit' button is at the bottom. A thick red arrow points from the bottom-left towards the 'Submit' button.

- c. Select the 'ok' button the view room



- d. You are now in your created room!

The screenshot shows the main interface of the room 'My Cool Room!'. It includes:

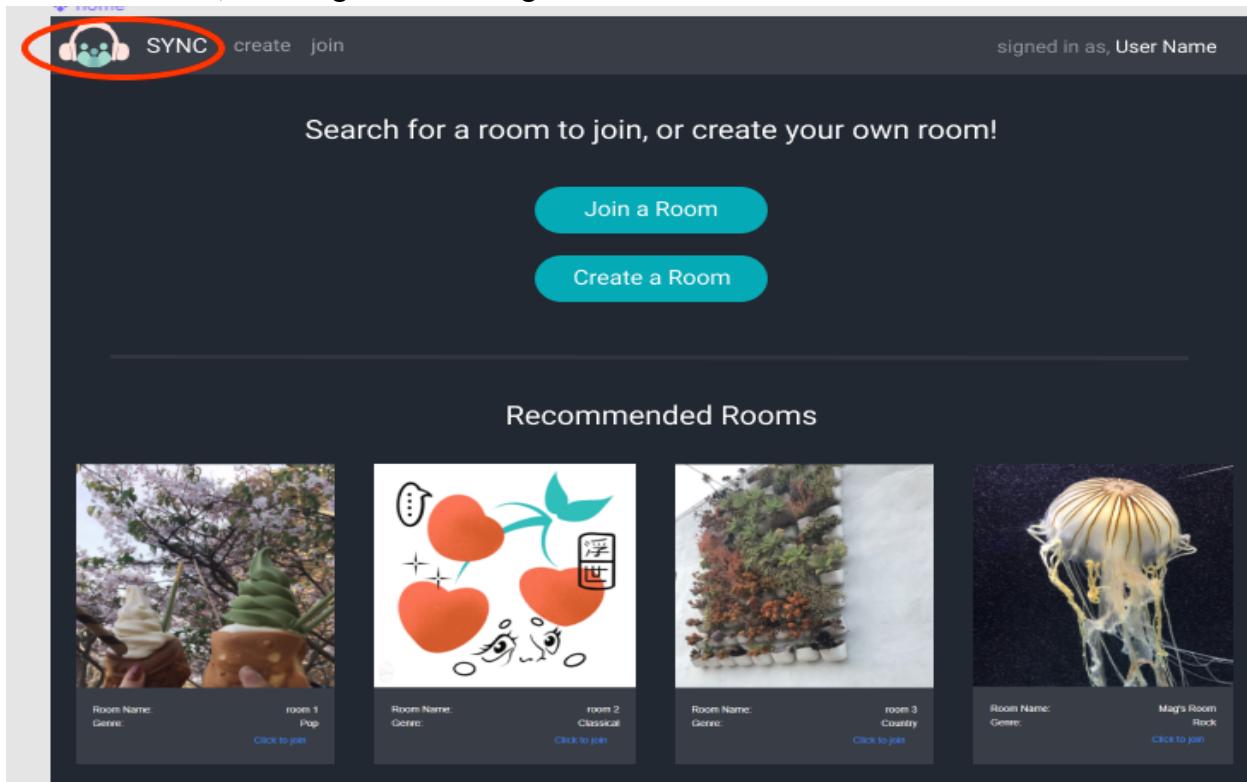
- Queue:**

| Queue      | Votes                      |
|------------|----------------------------|
| no love    | 3 <input type="checkbox"/> |
| crime pays | 1 <input type="checkbox"/> |
| trying     | 0 <input type="checkbox"/> |
- Player Controls:** Shows a thumbnail of a silver car parked next to orange flowers, with playback controls: 0:06 / 0:42, a progress bar, and volume/mute buttons.
- Chat Room:** A sidebar titled 'The My Cool Room! Chat Room' containing messages from various users:
  - Rebecca: If it wasn't for the beetles, music wouldn't even exist anymore!
  - Howard: what do you mean by that?
  - Luong: I need something more upbeat
  - Malcolm: Anyone know the name of this tune?
  - George: Let's not choose this song again
  - Jane: This is the best room I've been in
  - George: Now I'm confused.
  - Bryan: Now I'm confused.
  - Bill: Let's not choose this song again
  - Frank: I need something more upbeat
  - Frank: Now I'm confused.
  - Bryan: I need something more upbeat
 A 'Send A Message' input field and a 'Submit' button are also visible.
- Search Function:** A 'Search for a Song' input field at the bottom left.

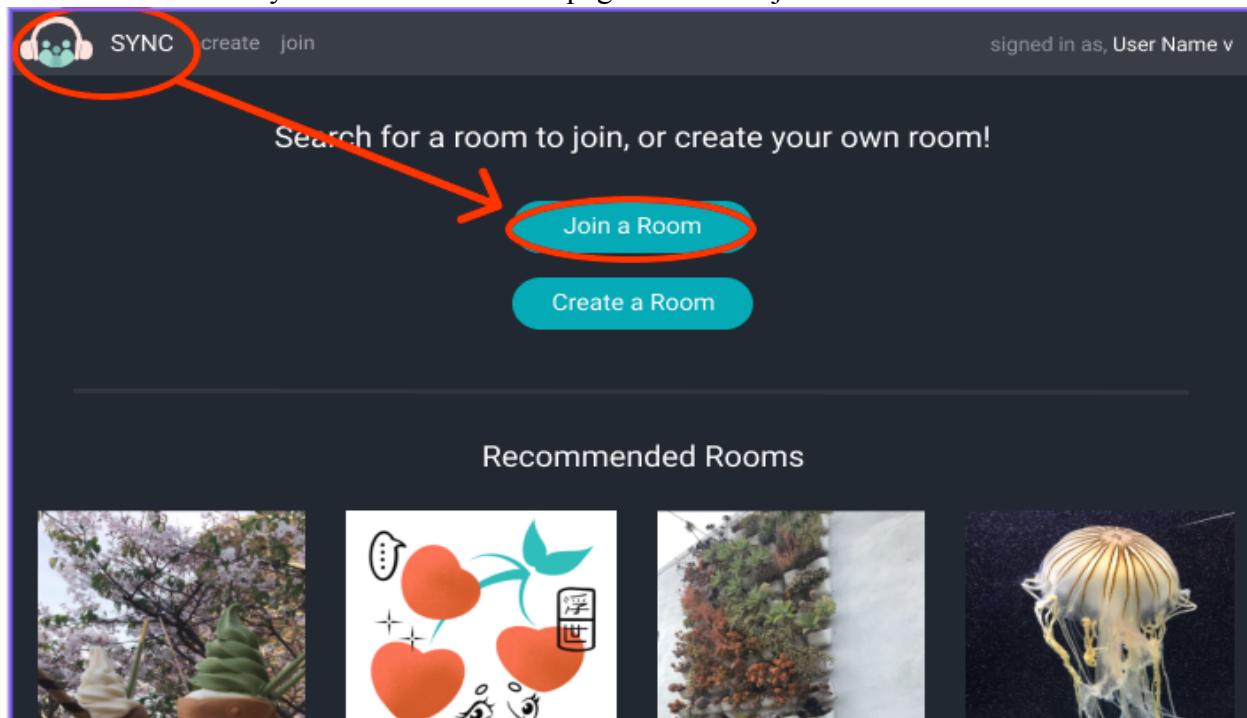
#### 4. Search and Join a room

There are a couple places to join a room. I will guide you through both locations

- Navigate to <http://3.17.66.0:3000/Home> by clicking this link or if already logged in, selecting the SYNC logo

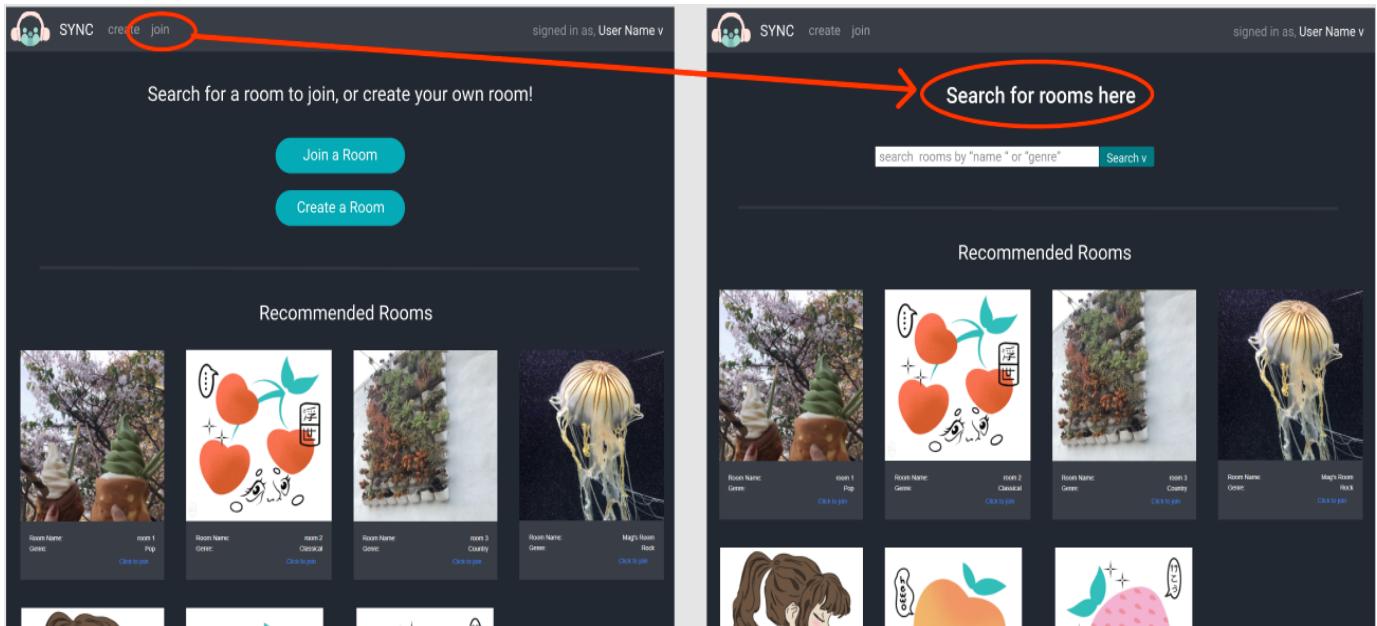


- Once you arrive to the home page select the 'join a room' button

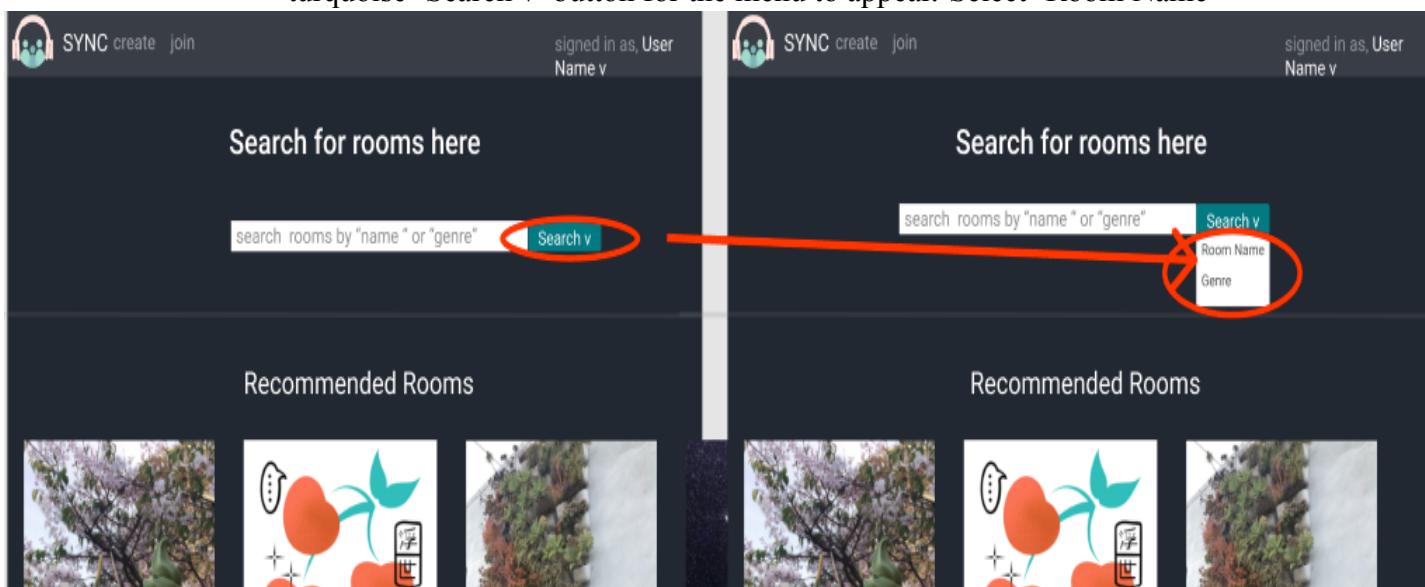


The **second** way to join a room is as such:

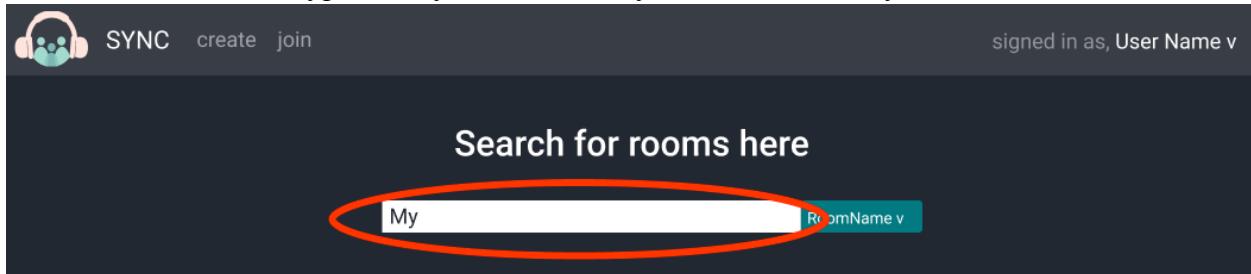
- Navigate to <http://3.17.66.0:3000/Join> by selecting provided a link or ( if already in site) selecting the join button in the nav bar located at the top of every page within the site



- The next thing is to search for something! You can search the room you have created!
  - You may filter using the drop down menu beside the search bar. Select the turquoise 'Search v' button for the menu to appear. Select 'Room Name'



ii. Type in ‘My’, or whatever you’d like to name your room



iii. Your room named ‘My Cool Room!’ will appear under the ‘Searched Rooms by “My” section. It should look like this:

A screenshot of a search results page. At the top, it shows the same header and search bar as the previous screenshot, with "My" typed into the search bar. Below the search bar, the text "Searched Rooms by “My”" is displayed in blue. Underneath this, a room is listed in a box with the following details:

|                 |                            |
|-----------------|----------------------------|
| Room Name:      | My Cool Room!              |
| Genre:          | Pop                        |
| No. of members: | 1                          |
| Link to join :  | <a href="#">Click here</a> |

At the bottom of the page, there's a section titled "All Rooms" with several small thumbnail images of other rooms.

- d. On the room you would like to join there is a link to join the room. Select the blue link that reads ‘ Click here’ on the room you would like to join

The screenshot shows a search interface with a search bar containing 'My'. Below the search bar, a red arrow points from the text 'Searched Rooms by "My"' to a room card. The room card displays the following information:

- Room Name:** My Cool Room!
- Genre:** Pop
- No of members:** 1
- Link to join :** [Click here](#)

A red circle highlights the 'Click here' link.

- e. After selecting the ‘Click here’ link on a room card, you should be redirected into your room! It should appear in a new window as such:

The screenshot shows the interior of the 'My Cool Room!' room. The room details are displayed at the top:

**My Cool Room!**  
Room Genre: Pop

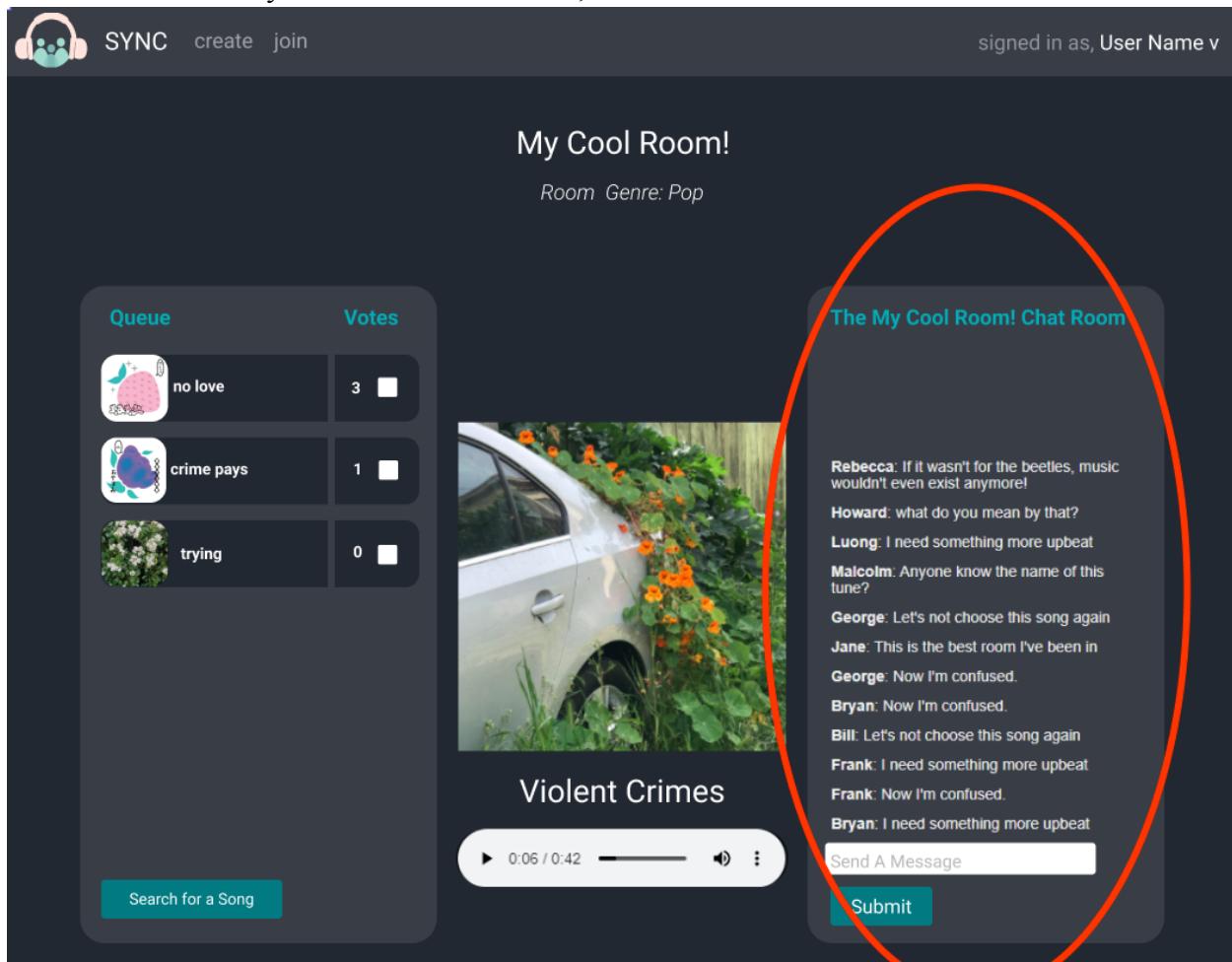
The interface includes the following sections:

- Queue:** A list of songs in the queue:
 

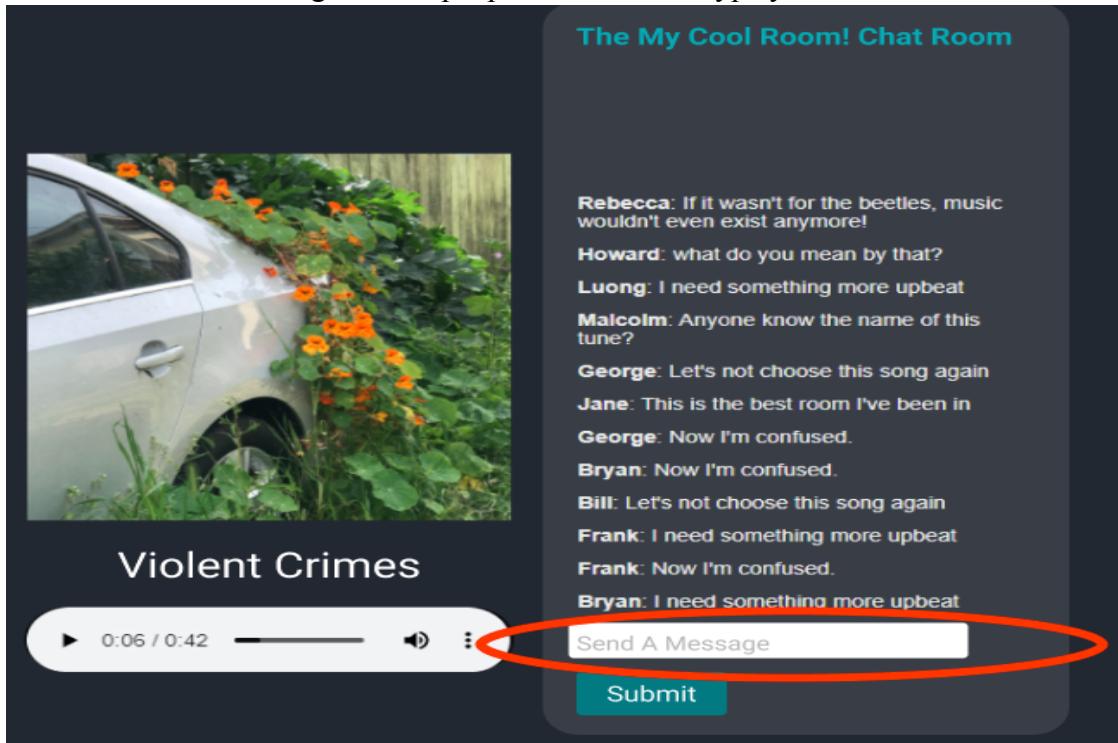
| Song       | Votes |
|------------|-------|
| no love    | 3     |
| crime pays | 1     |
| trying     | 0     |
- Player Controls:** Includes a play button, progress bar (0:06 / 0:42), volume control, and other playback options.
- Violent Crimes:** The currently playing song.
- The My Cool Room! Chat Room:** A sidebar displaying a conversation between several users:
  - Rebecca: If it wasn't for the beetles, music wouldn't even exist anymore!
  - Howard: what do you mean by that?
  - Luong: I need something more upbeat
  - Malcolm: Anyone know the name of this tune?
  - George: Let's not choose this song again
  - Jane: This is the best room I've been in
  - George: Now I'm confused.
  - Bryan: Now I'm confused.
  - Bill: Let's not choose this song again
  - Frank: I need something more upbeat
  - Frank: Now I'm confused.
  - Bryan: I need something more upbeat
- Message Input:** Fields for 'Send A Message' and 'Submit'.

## 5. Text in chat within a room

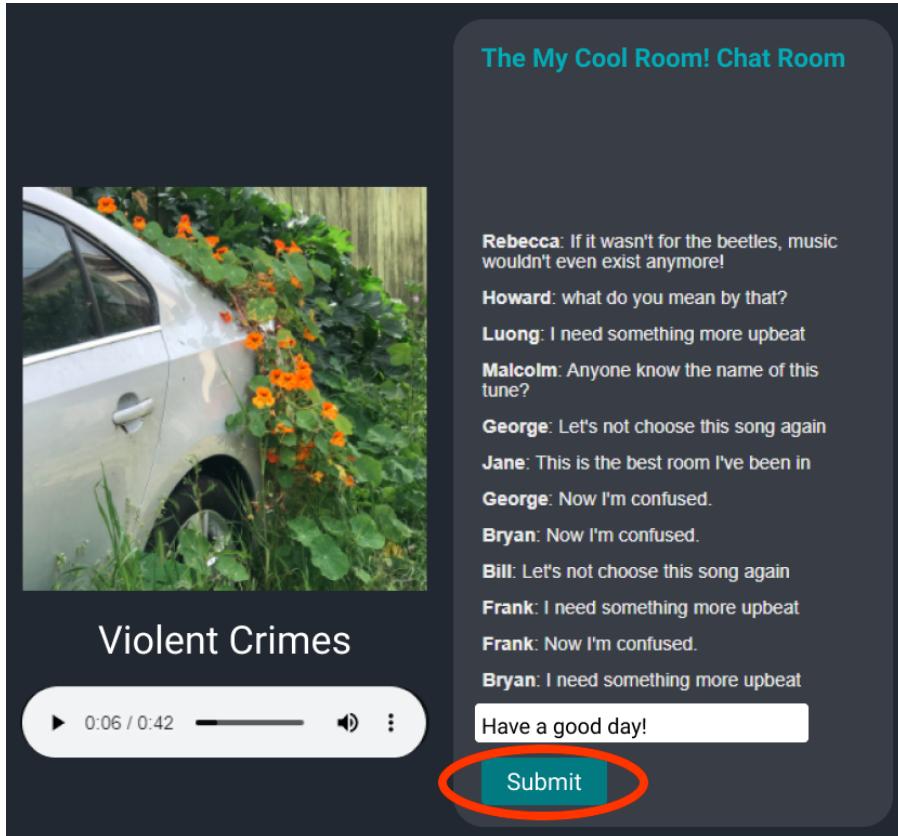
- a. Navigate to your room or a room by following step 4.
- b. Once you have entered a room, locate the chat room



- c. At the bottom of the chat window there is a text box where you are able to text something to other people in the room. Type your comment in the white text box.



- d. Once you have typed your message, hit the submit button below the text be sent



- e. After submitting the message, the message will displayed in the chat above for all to see

The My Cool Room! Chat Room

Rebecca: If it wasn't for the beetles, music wouldn't even exist anymore!

Howard: what do you mean by that?

Luong: I need something more upbeat

Malcolm: Anyone know the name of this tune?

George: Let's not choose this song again

Jane: This is the best room I've been in

George: Now I'm confused.

Bryan: Now I'm confused.

Bill: Let's not choose this song again

Frank: I need something more upbeat

Frank: Now I'm confused.

Bryan: I need something more upbeat

User: Have a good day!

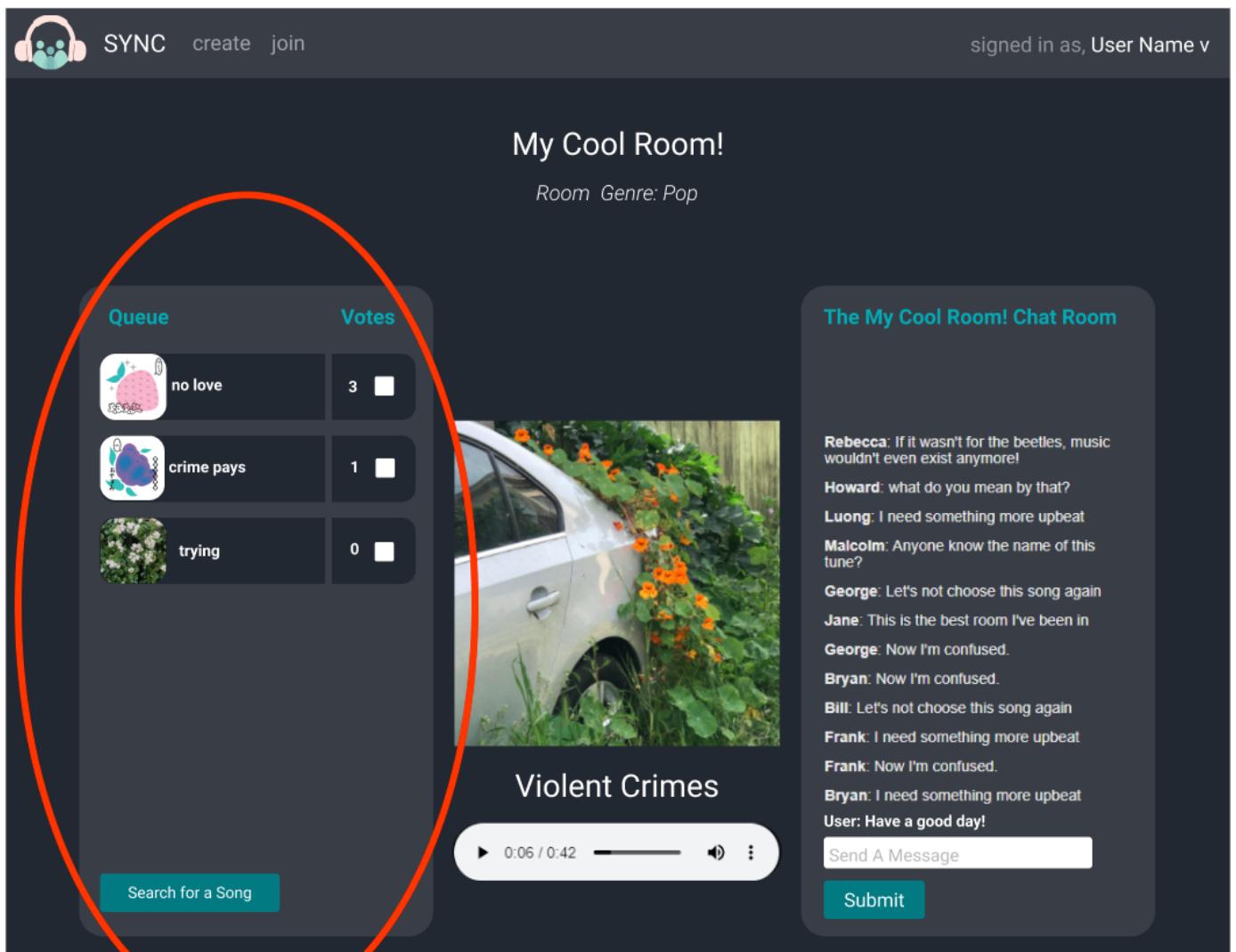
Send A Message

Submit

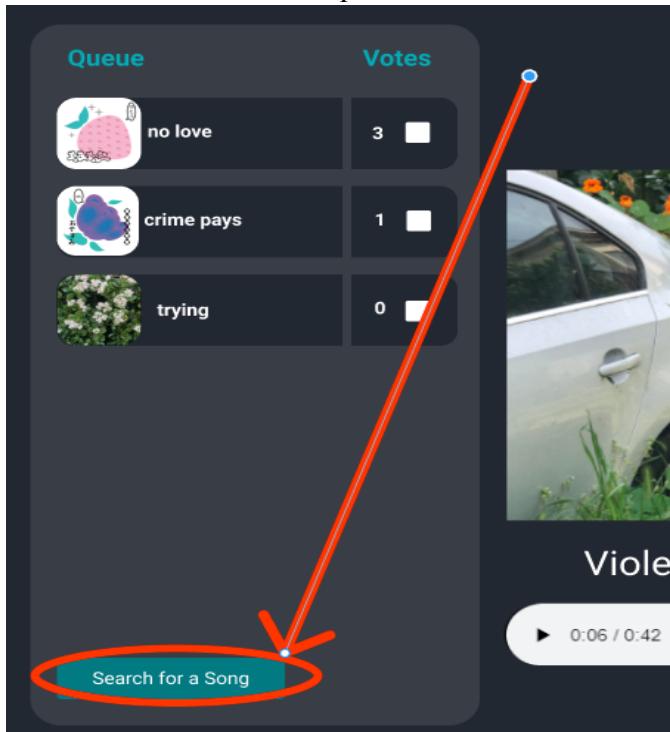
## 6. Search and add song into queue

- Navigate to your room or a room by following step 4.
- Once you have entered a room, locate the queue list located on the left of the screen

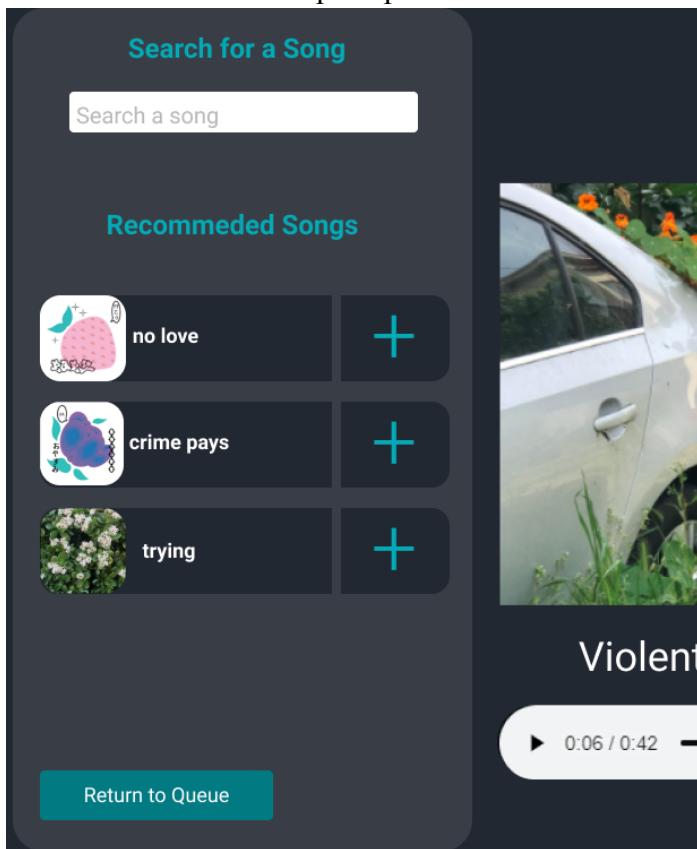
Here is listed all the songs in the queue, if any.



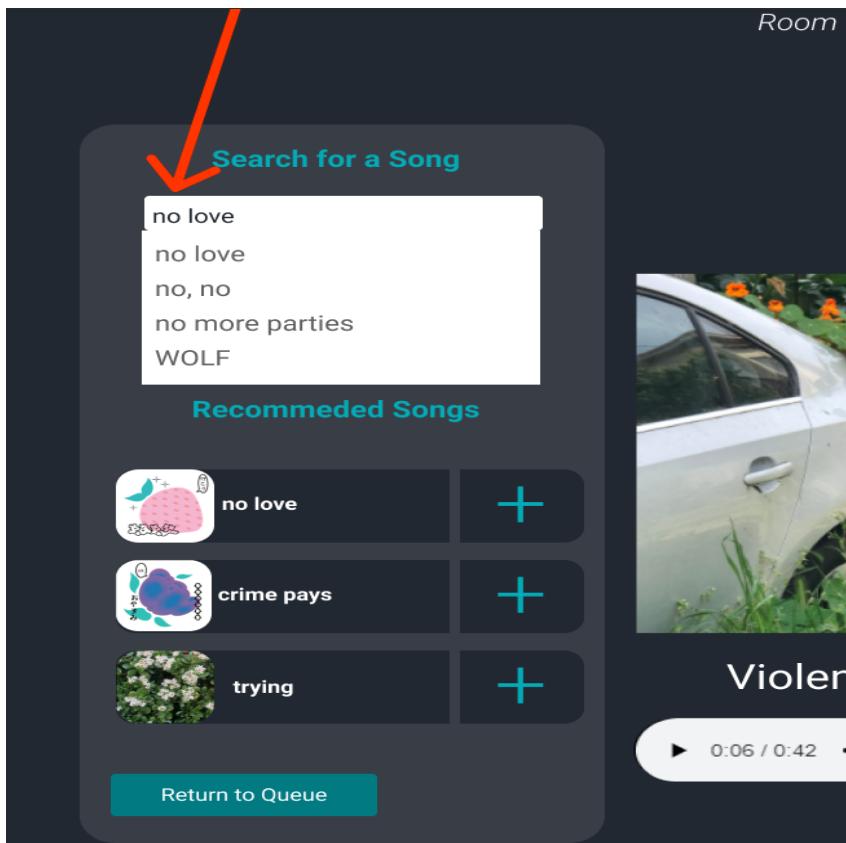
- c. To add songs to queue select the ‘search for a song’ button located at the bottom left on the queue section



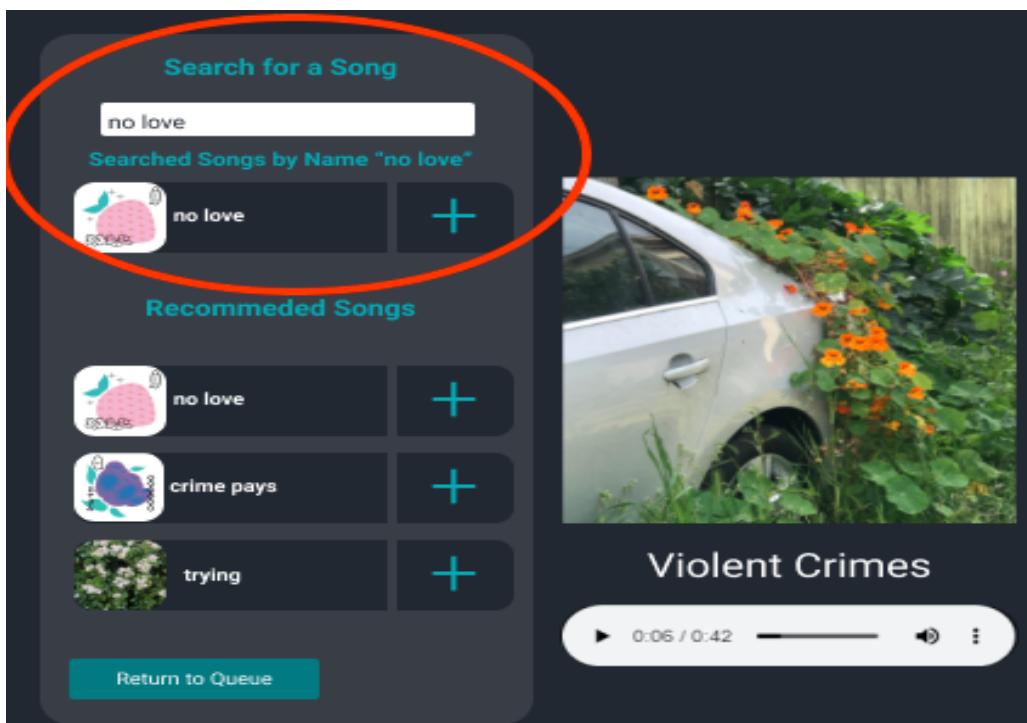
- d. This will prompt a search bar above the queue to open.



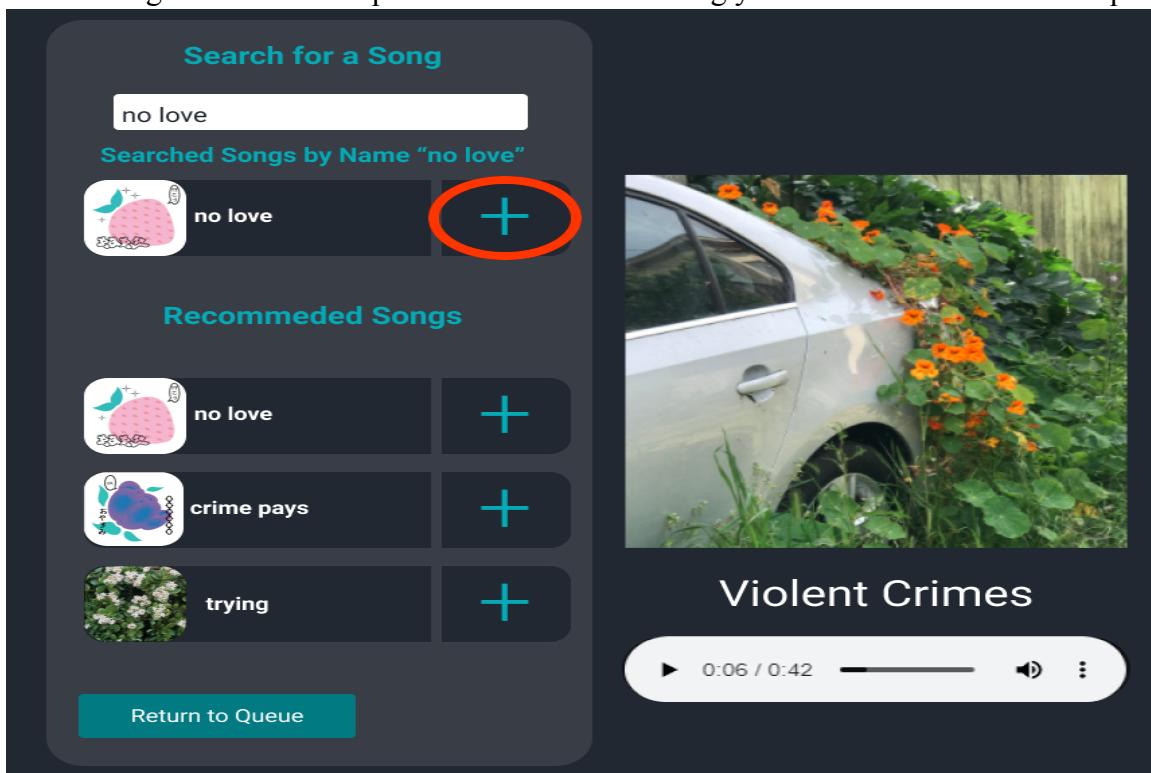
- e. Type in the search bar a name of a song you would like to play. An example would be 'no love'



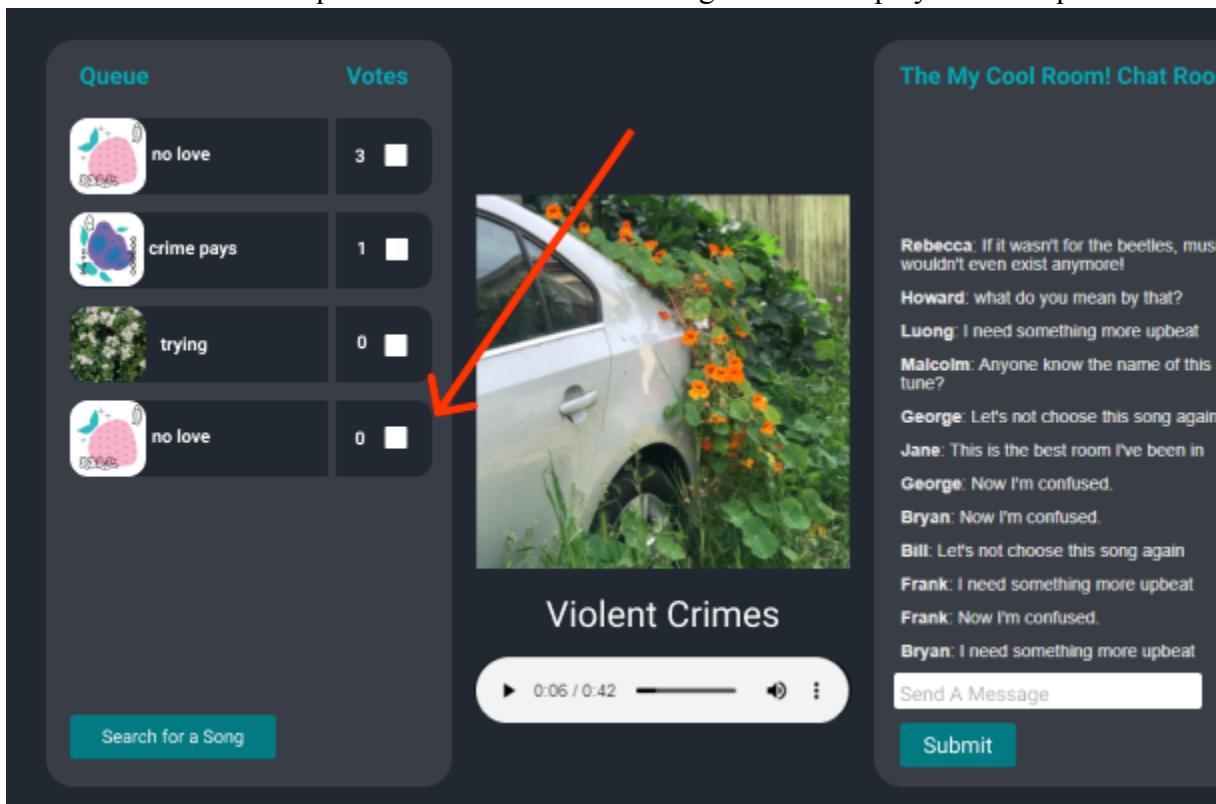
- f. After typing in everything hit enter and the results would show up directly under the search bar



g. Select the ‘+’ plus button next to the song you would like to add to the queue



h. Once the plus button is selected the song added is displayed in the queue



## 7. Add a vote to a song in queue

- a. Navigate to your or a room by following step 4.
- b. Once you have entered a room, locate the queue list located on the left of the screen

Here is listed all the songs in the queue if any.

The screenshot shows a mobile application interface for a music room named "My Cool Room!". The room is set to genre "Pop". On the left, a red oval highlights the "Queue" section, which lists three songs:

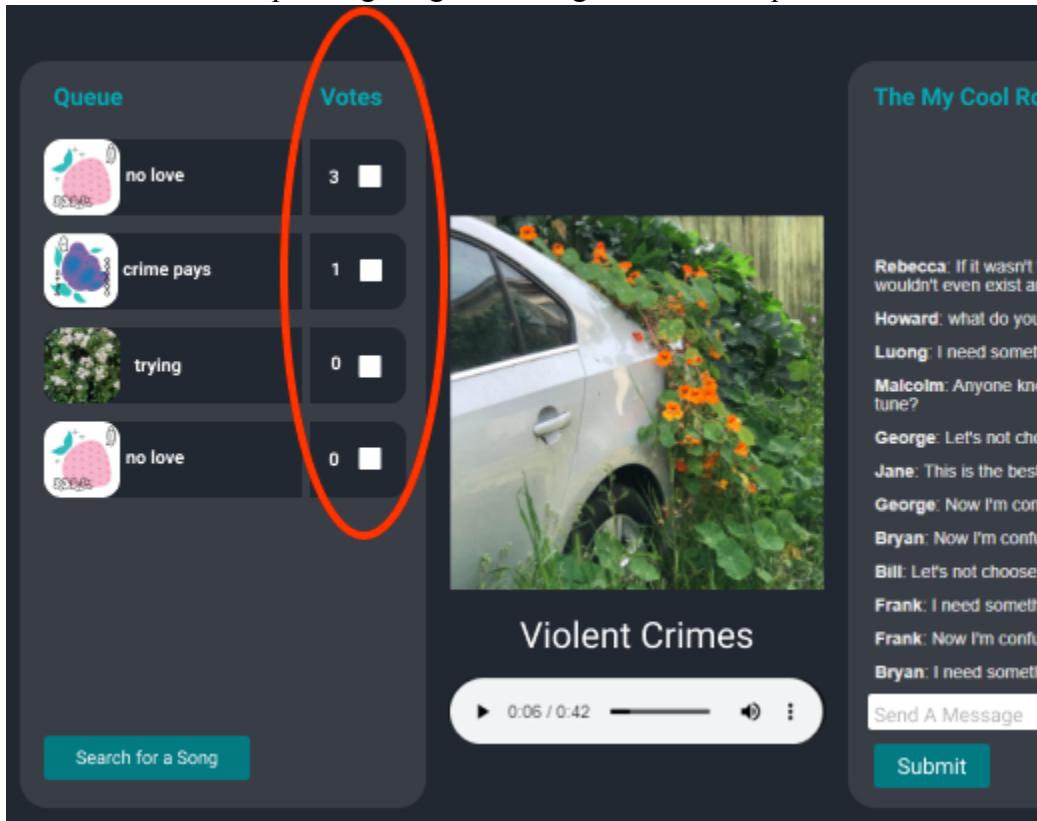
| Queue      | Votes                      |
|------------|----------------------------|
| no love    | 3 <input type="checkbox"/> |
| crime pays | 1 <input type="checkbox"/> |
| trying     | 0 <input type="checkbox"/> |

Below the queue is a search bar labeled "Search for a Song". In the center, there's a thumbnail image of a silver car surrounded by orange flowers, with the text "Violent Crimes" overlaid. At the bottom, a playback control bar shows "0:06 / 0:42" and various control icons. To the right, a separate box titled "The My Cool Room! Chat Room" displays a conversation log:

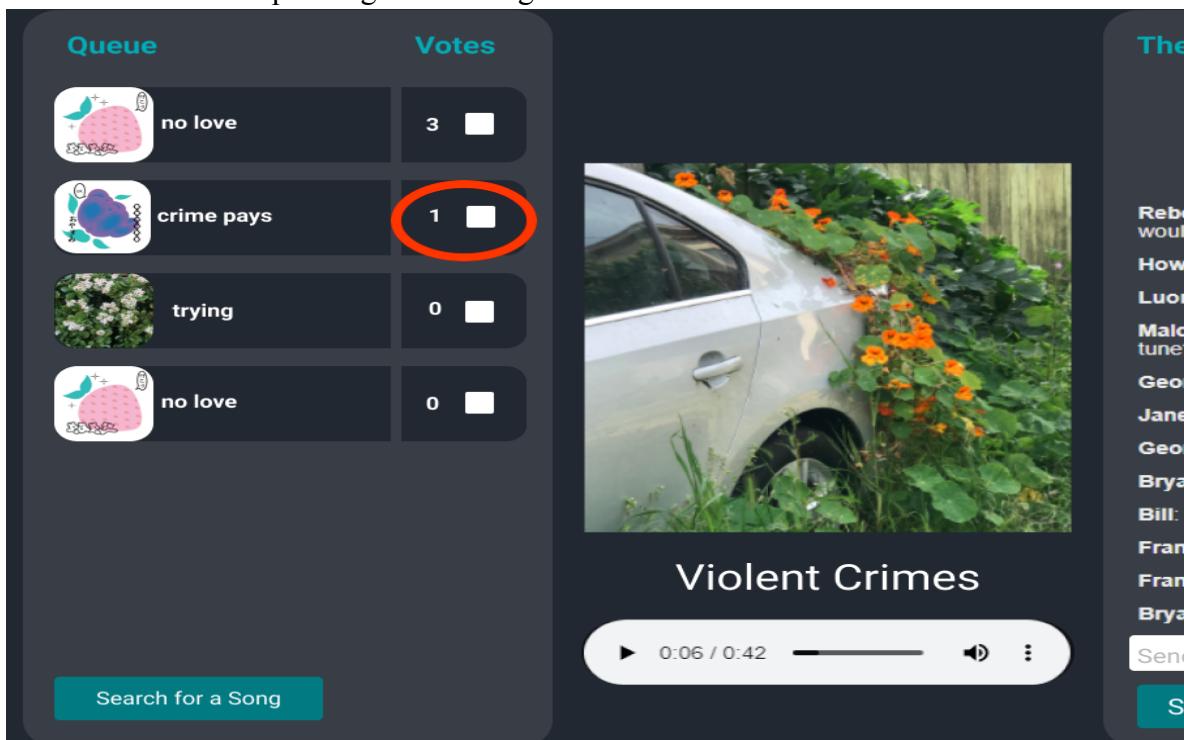
- Rebecca: If it wasn't for the beetles, music wouldn't even exist anymore!
- Howard: what do you mean by that?
- Luong: I need something more upbeat
- Malcolm: Anyone know the name of this tune?
- George: Let's not choose this song again
- Jane: This is the best room I've been in
- George: Now I'm confused.
- Bryan: Now I'm confused.
- Bill: Let's not choose this song again
- Frank: I need something more upbeat
- Frank: Now I'm confused.
- Bryan: I need something more upbeat
- User: Have a good day!

At the bottom of the chat box are "Send A Message" and "Submit" buttons.

- c. To add vote go to any of the songs in the queue: Navigate to the vote column to corresponding songs on the right side of the queue list



- d. Choose a song you'd like to add a vote for, and select the vote check box corresponding to that song



- e. After selecting the checkbox, the box will appear checked and the number will change up one vote. You have voted!

| Queue      | Votes                                 |
|------------|---------------------------------------|
| no love    | 3 <input type="checkbox"/>            |
| crime pays | 2 <input checked="" type="checkbox"/> |
| trying     | 0 <input type="checkbox"/>            |
| no love    | 0 <input type="checkbox"/>            |

Search for a Song

The My Cool

Rebecca: If it was  
wouldn't even exist

Howard: what do

Luong: I need som

Malcolm: Anyone  
tune?

George: Let's not

Jane: This is the b

George: Now I'm

Bryan: Now I'm co

Bill: Let's not choo

Frank: I need som

Frank: Now I'm co

Bryan: I need som

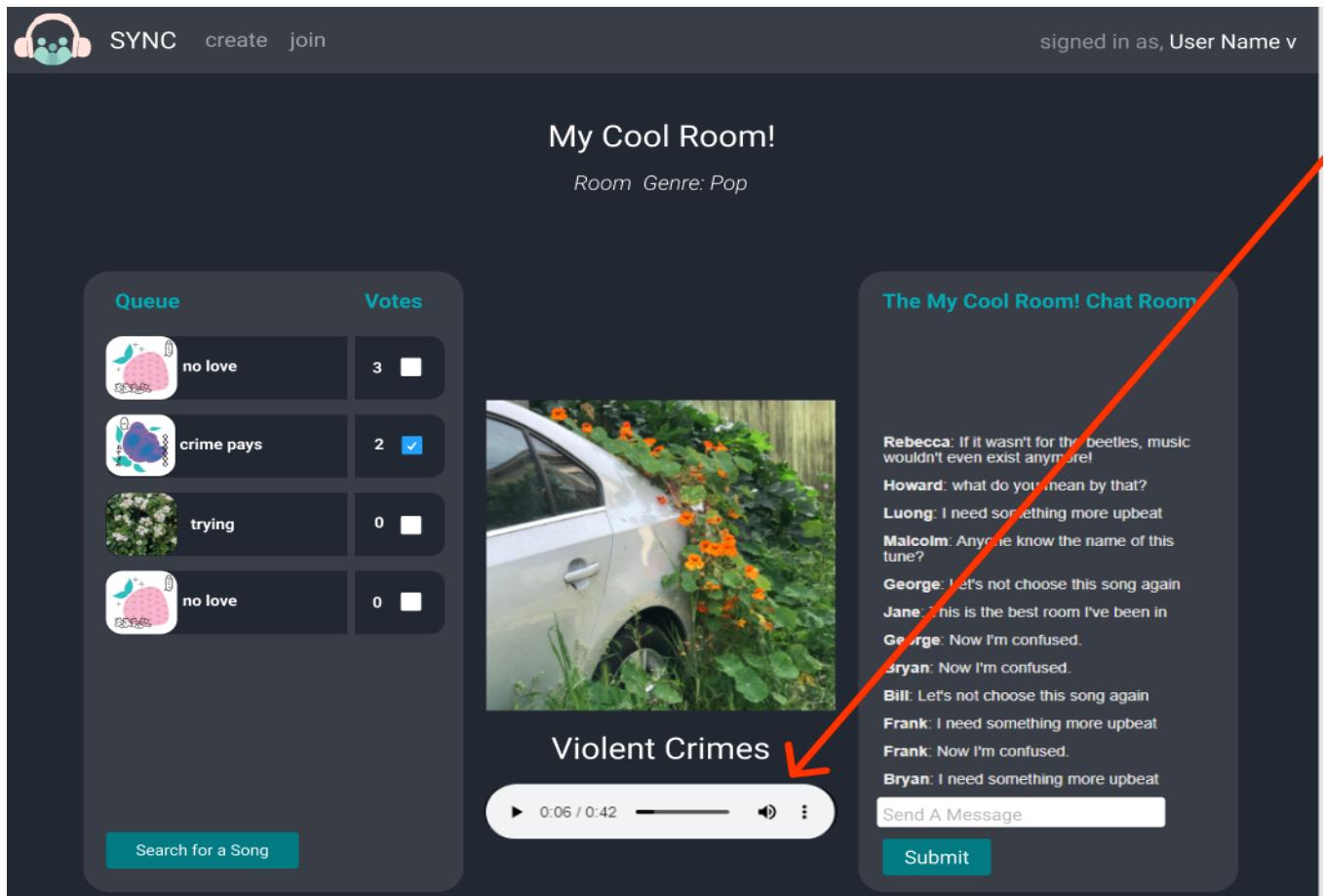
Send A Message

Submit

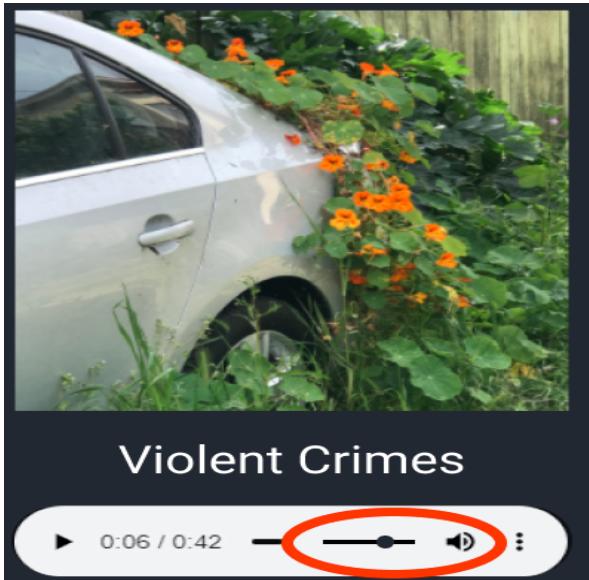
## 8. Change Volume in room

When you enter a room the volume is quite low, here is where you can change the volume to your liking!

- a. Navigate to your or a room by following step 4.
- b. Once you have entered a room, locate the player below the album cover of the currently playing song



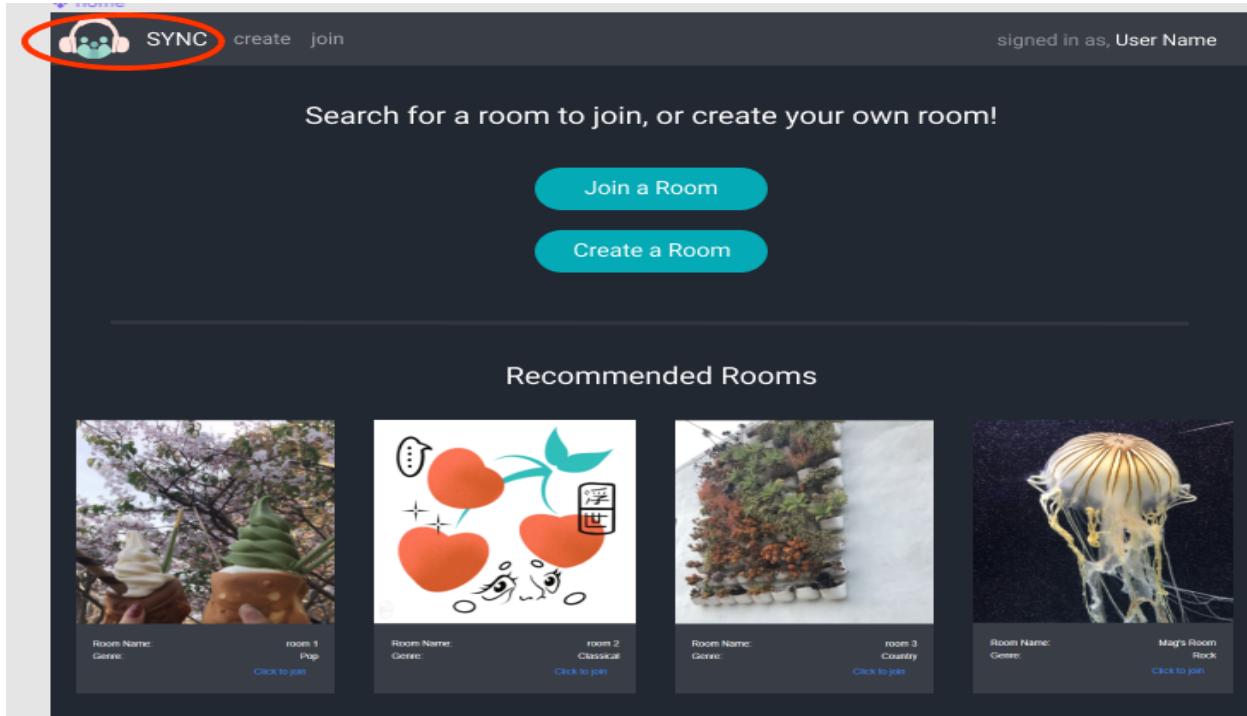
- c. Locate the volume icon and select the slider button to change the volume (if you are on chrome, it will appear if you hover over the volume icon )



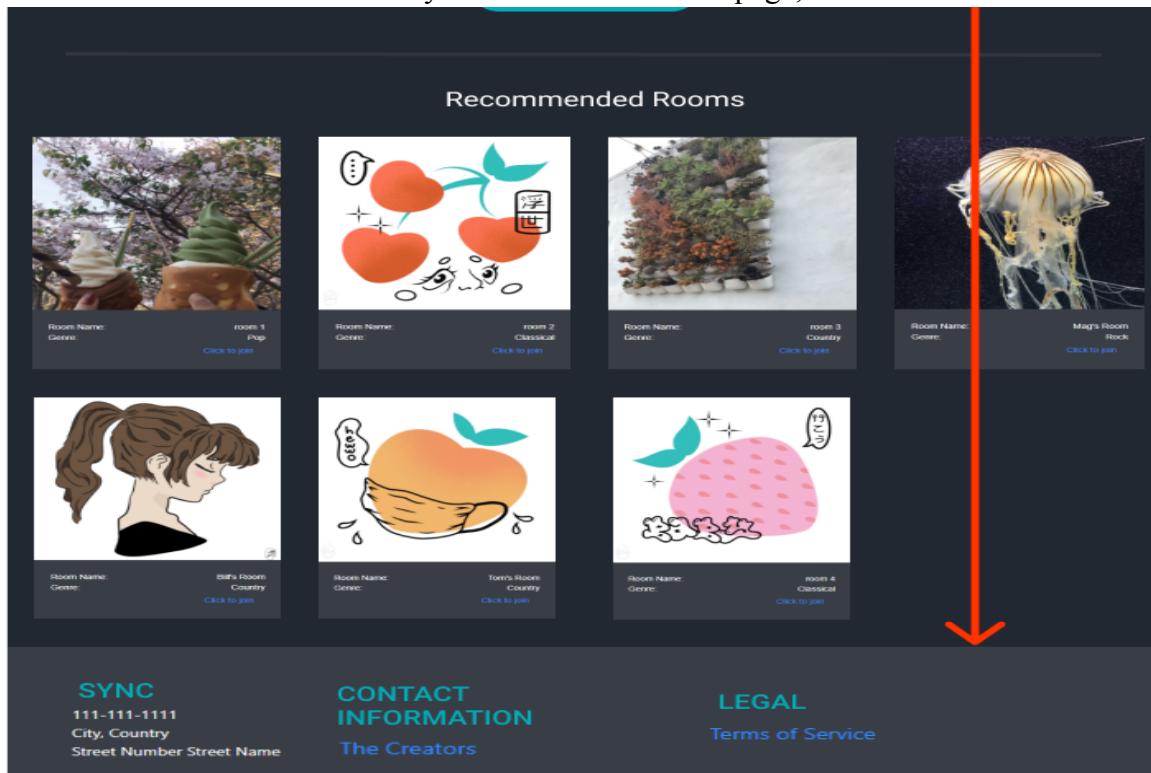
- d. Slide left or right to lower the volume or bring it up.

## 9. View ‘The Creators’ and ‘Terms of Service’

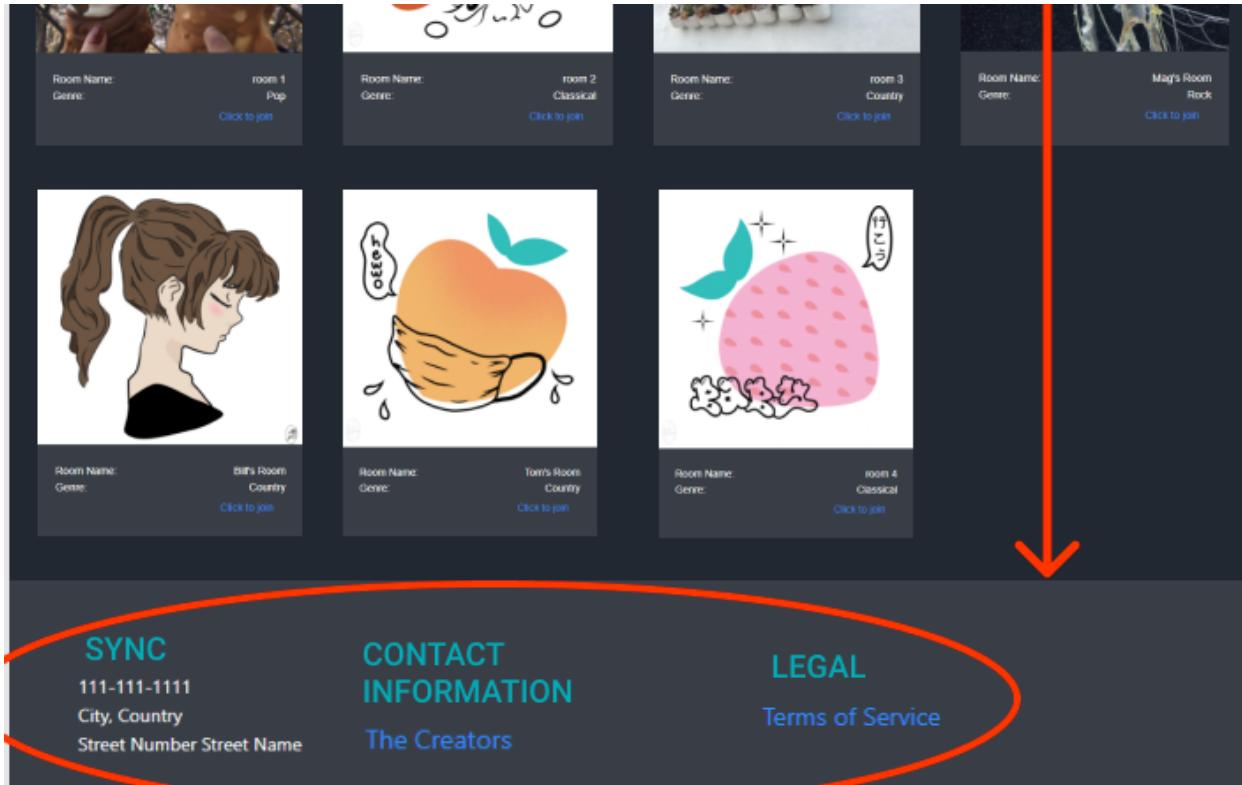
- Navigate to <http://3.17.66.0:3000/Home> or if already in SYNC, selecting the SYNC logo



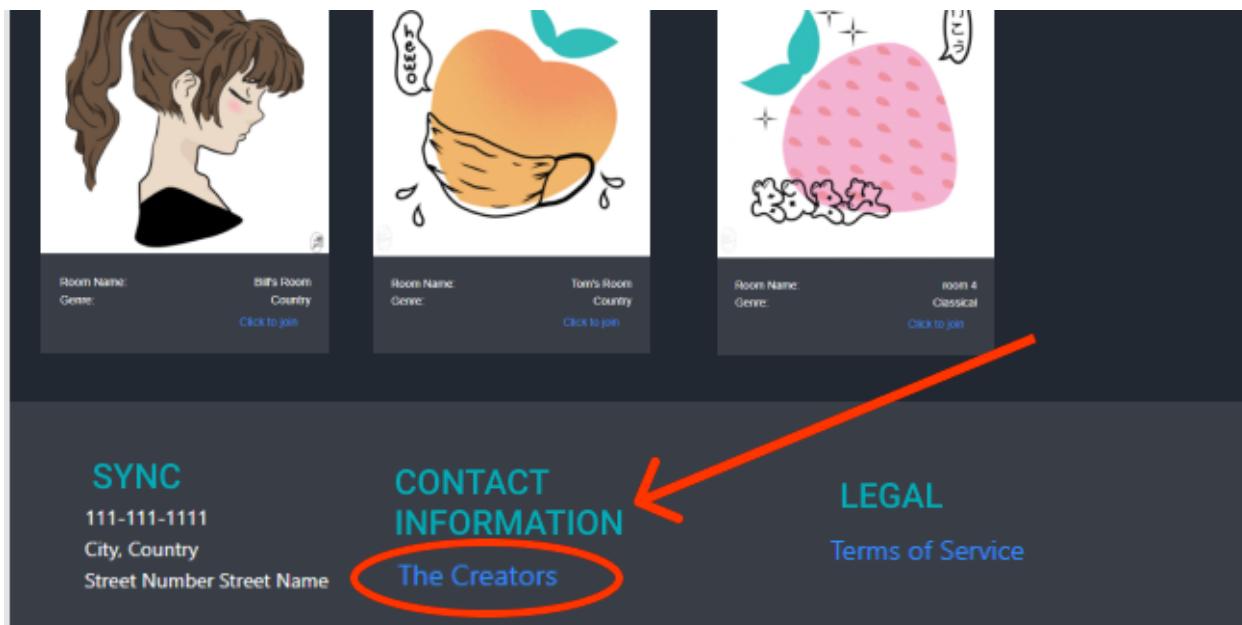
- Scroll down until you hit the footer of the page, it should look like this:



This is the footer :



c. Select 'The Creators' link to navigate to the creators page.



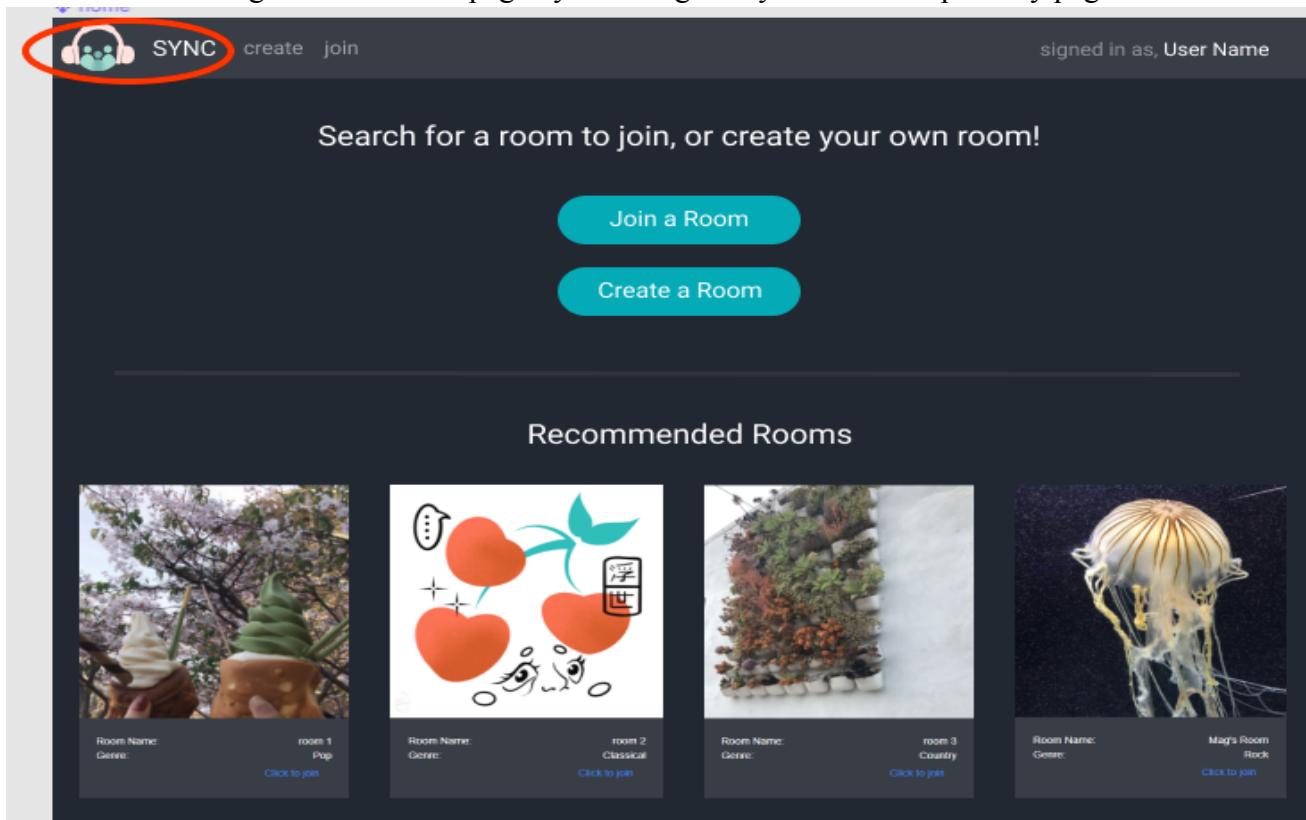
- d. Once you click on ‘The Creators’ link, you redirected to the Creators! page. It will look like this:

The screenshot shows a dark-themed web page titled "The Creators!". It displays six profiles of team members, each with a pixelated portrait, name, title, a short bio, and a "Go to [Name]'s github" button.

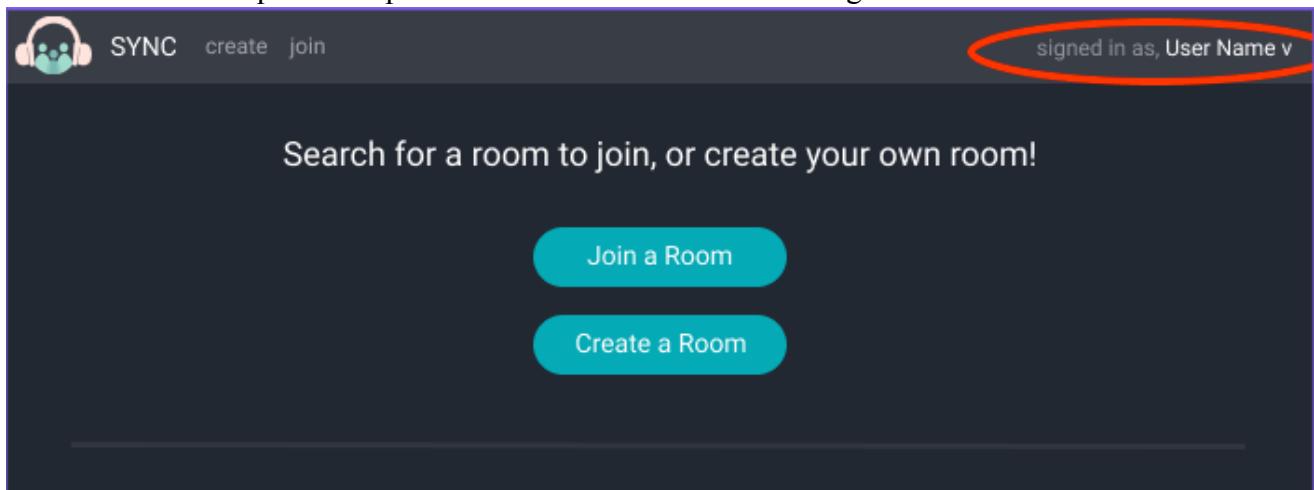
- Rebecca Zumaeta**  
Team Lead  
Hi, I am Rebecca!  
[Go to Rebecca's github](#)
- Bryan Feltner**  
Frontend Lead  
Hi, I am Bryant!  
[Go to Bryan's github](#)
- Ashwini Managuli**  
Backend Lead  
Hi, I am Ashwini!  
[Go to Ashwini's github](#)
- Malcolm Angelo De Villar**  
Frontend member  
Hi, I am Malcolm!  
[Go to Malcolm's github](#)
- Hirva Patel**  
Frontend member  
Hi, I am Hirva!  
[Go to Hirva's github](#)
- Luong Dang**  
Backend member  
Hi, I am Luong!  
[Go to Luong's github](#)
- Vishakha Tyagi**  
Backend member  
Hi, I am Vishakha!  
[Go to Vishakha's github](#)

## 10. Log out

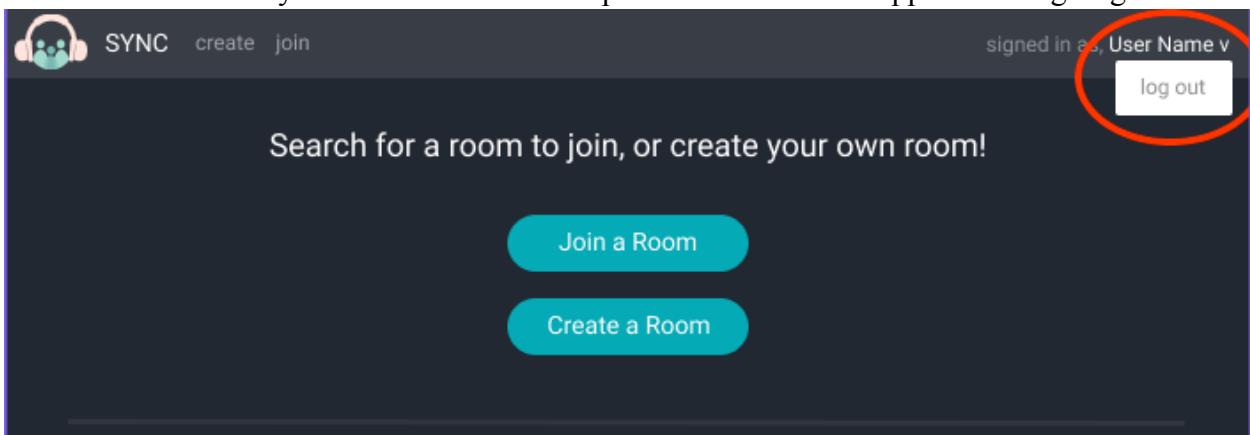
- Make sure you are within the site, anywhere within the site. You can always navigate to the home page by selecting the sync icon on top of any page



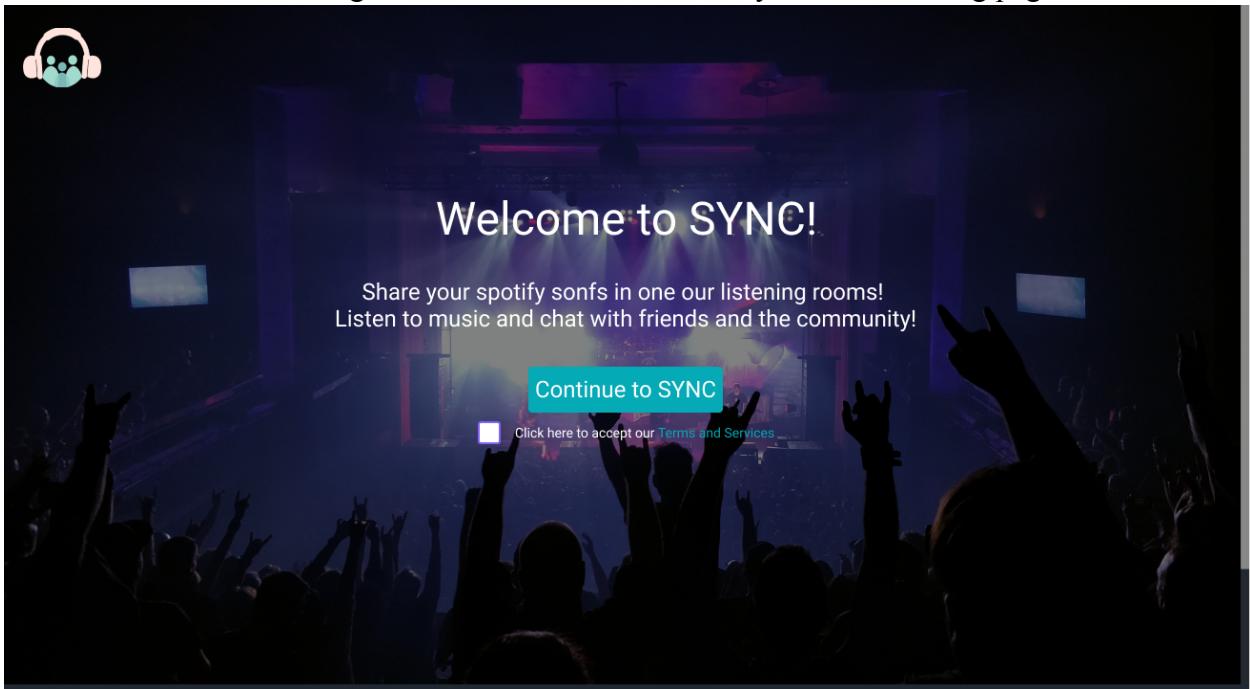
- Go up to the top left of the screen where it reads 'signed in as *User Name*'



- c. Select your username and a drop down menu should appear reading ‘log out’



- d. Select the log out button and it will redirect you to the landing page!



Link to Figma, used to make wire frames:

<https://www.figma.com/file/o1oD3F0Ud3COzAhv0U7R61/SYNC>

## **4. High level database architecture and organization V2**

The DBMS we chose to create our database is MySQL because it is easy to understand and some of us have experience with it; it is an organized and widely used DBMS.

### **1. User (strong)**

- a. Unregistered users has one Spotify account
- b. Registered user has one and only one username
- c. Registered user has one and only one user\_id
- d. Registered user has zero to one display name
- e. Registered user has one and only one profile picture
- f. Registered user shall be able to send zero to many invite links
- g. Registered user shall be able listen 0 to one song in real time in room
- h. Registered user shall be able to be or not be a host
- i. Registered user shall be able to be or not be a participant
- j. Registered user shall have only one profile page
- k. Registered user shall have zero to one profile picture
- l. Registered user shall be zero to one host
- m. Registered user shall be zero to one participant
- n. Registered user shall be able to create zero to many public rooms
- o. Registered user shall be able to be in zero to one public room
- p. Registered user shall be able to be in zero to one private room
- q. Registered user shall be able to be in zero to one communal room
- r. Registered user shall be able invite 0 to many users to their room
- s. Registered user shall be able choose 0 to many songs to play
- t. Registered user shall be able to choose zero to many created rooms

### **2. Rooms (weak)**

- a. Rooms has one and only one display name
- b. Rooms shall display one and only one hostname
- c. Rooms shall display one to many songs in queue
- d. Rooms shall display one and only one current song
- e. Rooms shall display one and only one genre
- f. Rooms shall display one and only one chat box
- g. Rooms shall display one to many voted songs

### **3. Host (weak)**

- a. Host has to be in one and only one room
- b. Host shall control one and only one room.
- c. Host shall create one and only one room
- d. Host shall name one and only one room

- e. Host shall play one to many songs in room

**4. Friends (weak)**

- a. Friends shall be able to be added from zero to many registered user's friend's list.
- b. Friends shall be able to be removed from zero to many registered user's friend's list.
- c. Friends shall be blocked from zero to many registered user's friend's list.
- d. Friends shall Direct Message zero to many friends

**5. Chat (weak)**

- a. Chat box shall hold chat messages from zero to many users
- b. Chat box shall exist in one to many rooms

**6. Website (strong)**

- a. Website shall display zero to many Direct Messages
- b. Website shall display zero to many added friends
- c. Website shall be able to allow user to like zero to many playlists

**1. User (strong)**

- User\_id: strong key, numeric
- spotify\_id: weak key, numeric
- profile\_pic: weak
- Display\_name: alphanumeric

**2. Rooms (weak)**

- room\_type : alphanumeric
- room\_id : strong key, numeric
- room\_name : multivalue, alphanumeric
- description : multivalue, alphanumeric
- current\_song :alphanumeric
- room\_host : alphanumeric
- password : weak key, numeric
- status : key, numeric
- max\_members : key, numeric
- Current\_number: key, numeric

**3. Host (weak)**

- Host\_id: strong key, numeric

**4. Chat (weak)**

- tab\_id : strong key, numeric
- tab\_status : key, numeric
- server : key, numeric

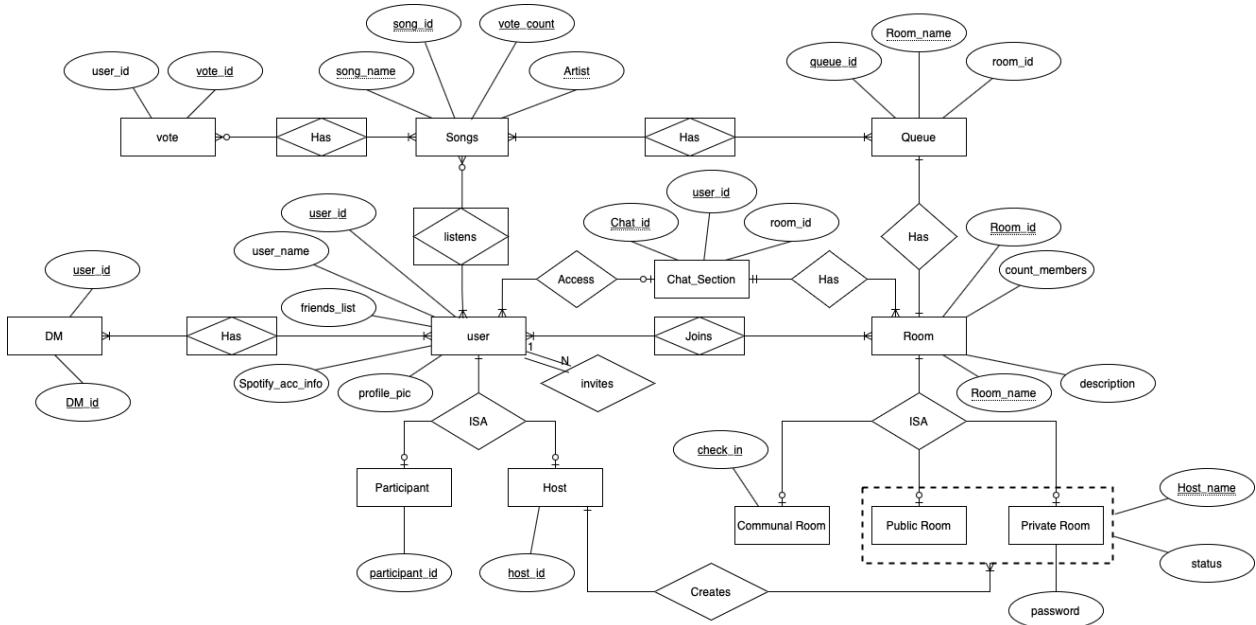
**5. Profile:** It has the information describing the registered user.

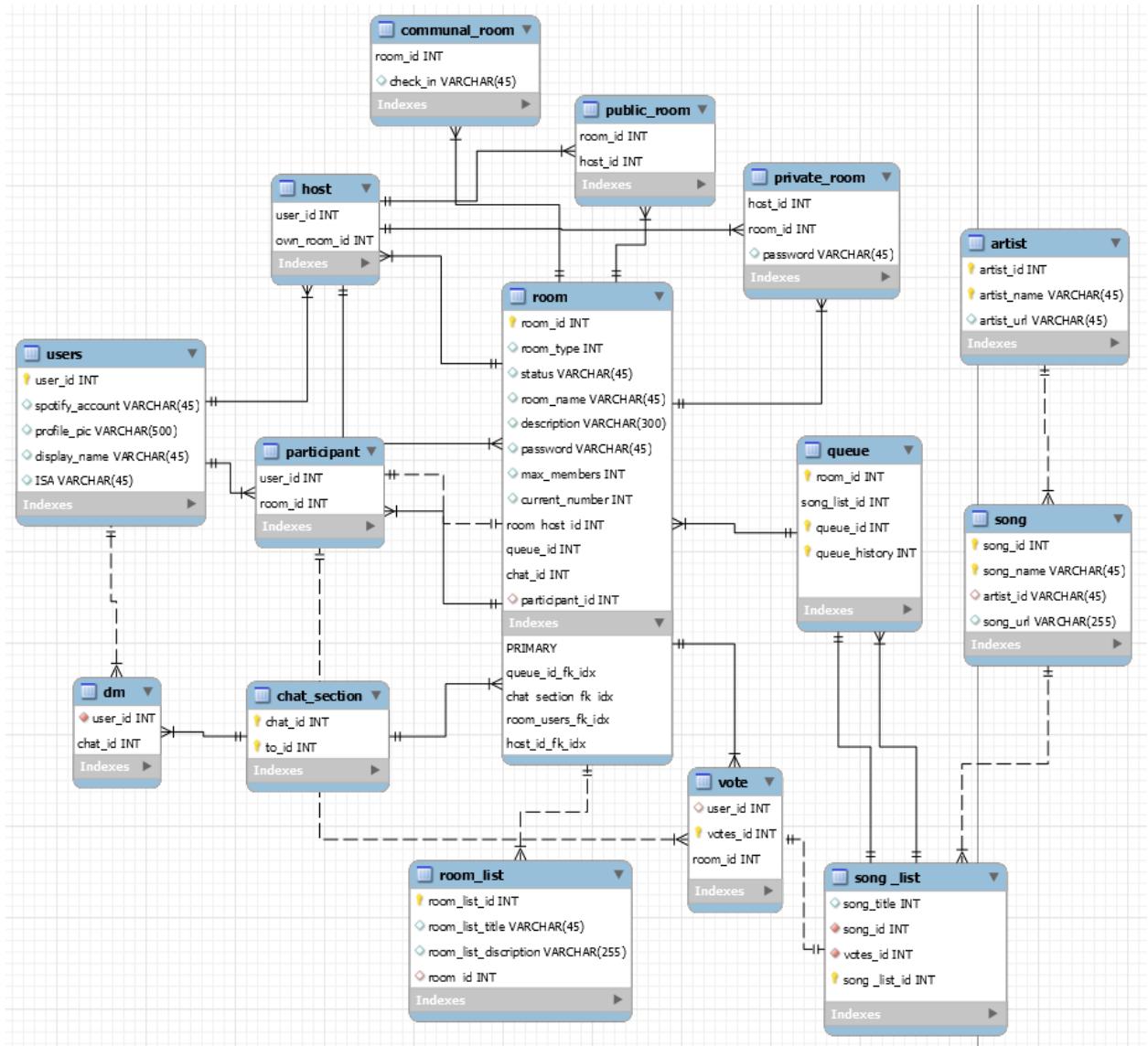
- user\_id: strong key, numeric
- activity : key, numeric
- profile\_photo:
- status :key, numeric

**6. Spotify\_API**

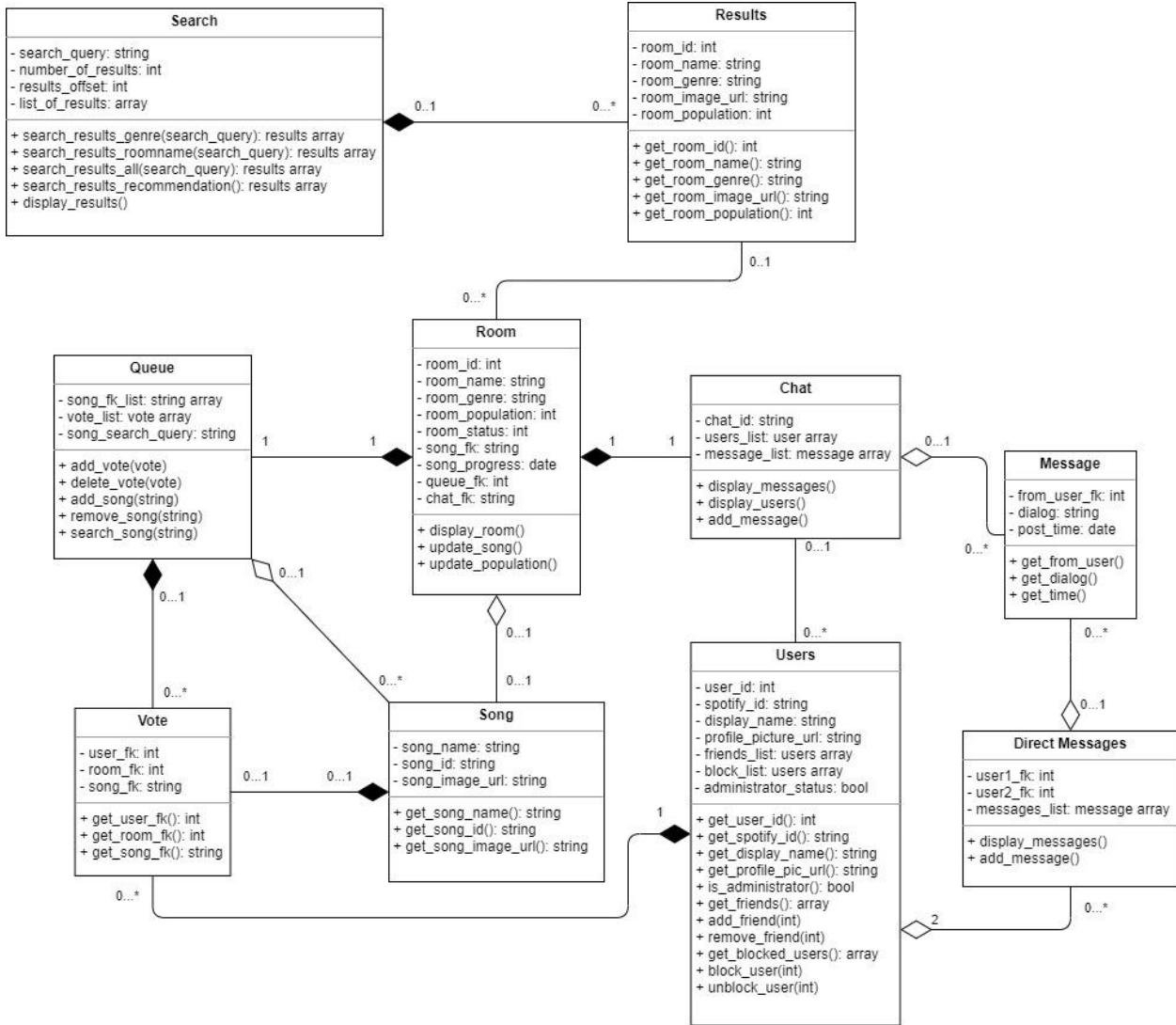
- connectivity : key, numeric
- current\_song\_title : alphanumeric
- progress : key, numeric
- song\_title : multivalue, alphanumeric
- artist : multivalue, alphanumeric
- image\_url : multivalue, alphanumeric
- genre : multivalue, alphanumeric
- album : multivalue, alphanumeric
- artist\_name : multivalue, alphanumeric
- songs : multivalue, alphanumeric
- album : multivalue, alphanumeric

- image\_url :multivalue, alphanumeric
- genre\_name : multivalue, alphanumeric
- description : multivalue, alphanumeric
- top\_playlist : multivalue, alphanumeric

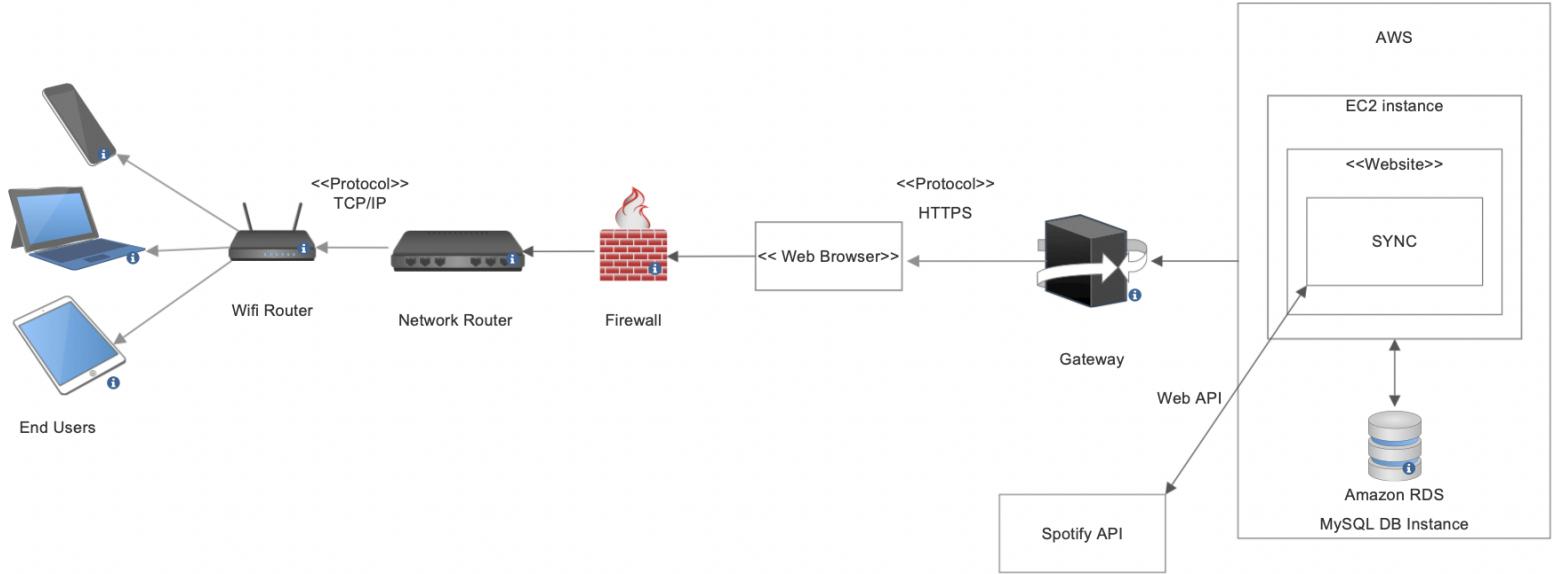




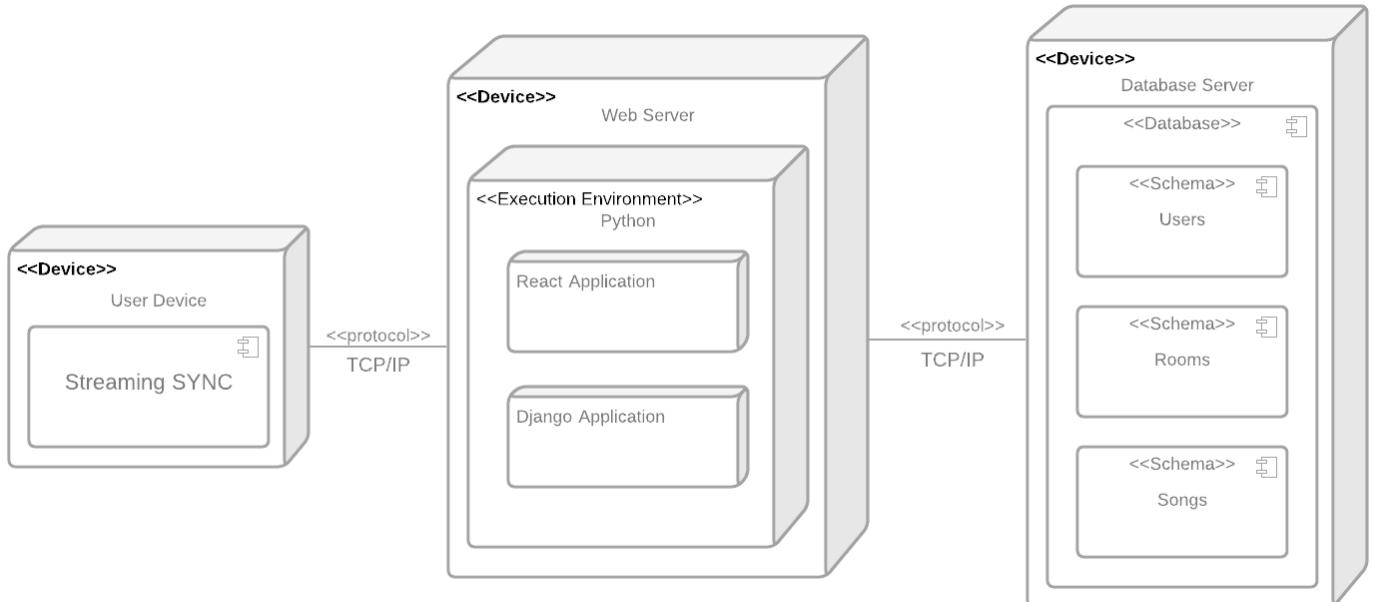
## 5. High Level Diagrams V2



## Application Network Diagram



## Deployment diagram



## 6. Detailed list of contributions

|                                                             |                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-------------------------------------------------------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Team Lead,<br>Document Master                               | Rebecca Zumaeta     | Designed tiny accents of the site. Assisted in writing various sections or documentation. Wrote all of section 3 WireFrames Assigned tasks to members on the team and became available when needing feedback/guidance. Guided member in tasks to be done for horizontal prototype. Set checkpoints for multiple tasks to be done before milestone. Set and hosted all meetings, kept in contact with every member.                                                                                                                                                                                                             |
| Front End Lead,<br>mediator                                 | Bryan Fetner        | Lead front end team in tasks to be done for the site. Built out various pages of the site and coded the UI and UX of the site along with his members. Designed pages and layouts and ‘animated’ features within them. Kept in contact with members and worked with them diligently. Attended all meetings for the entirety of the time and continuously held conversations and asked questions. Spoke out when passionate but open to other ideas when conversing with the team. Consistently updated team lead and confronted her when having questions or concerns.                                                          |
| Back End Lead,<br>Document<br>contributor,<br>Site Deployer | Ashwini<br>Managuli | Tied together back to front and deployed site every so often to reflect the development branch. Contributing to the designing of functions such as search and various other apis that will be implemented next milestone. Deployed current site. Tasked back end members to move along the process of implementing apis and back end for next Milestone. Attended all scheduled meetings and contributed into conversation when wanting to share her thoughts and experiences. Was open to feedback and implemented constructive criticism well. Reported back to the team lead a few times a week and got in contact with her |

|                                                        |                          |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|--------------------------------------------------------|--------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                                        |                          | back end team to give instruction.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| Front End Member Document contributor                  | Malcolm Angelo De Villar | <p>Contributed to coding a great portion of the front end such as landing page and all other pages within the site. Updated and worked with the front end lead in various voice meetings. Assisted in designing and building the front end of the site that is on the current deployed site. Attended all meetings and gave input when he seemed best knowledgeable about the subject. Updated team lead various times. Put out work when asked and continuously checked with the team and team lead when making decisions.</p>                                                                                   |
| Front End Member                                       | Hirva Patel              | <p>Worked heavily on converting html css and js from vertical to react for horizontal. Worked on various api's that are and not yet implemented. Contributed to the front end of the site that the front end lead tasked to do such as the search functions and room displays. Assisted in connecting front end to back with Ashwini. Attended all meetings and contributed to the conversation when she felt best fit and was passionate about the subject. Reported back to the team lead and team when she had updates on work and wanted feedback. Took constructive criticism well and worked off of it.</p> |
| Github Master and back end member Document contributor | Vishakha Tyagi           | <p>Assisted Ashwini when possible, working on api's to be implemented. Build the backend of Login Authentication and is working on its frontend. Assisted in doc when tasked such as the first 2 sections. Attended all meetings and gave input when she felt best knowledgeable upon the subject and is passionate upon ideas. Checked with other members, especially with team lead, when it comes to making any decisions within documentation.</p>                                                                                                                                                            |

|                 |            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|-----------------|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Back End Member | Luong Dang | <p>Assisted in the document when notes were given. Built functions that will be implemented when back end is implemented such as Spotify API. Contributed to api functions that have and will be implemented, such as the function of album covers to be shown for rooms and within rooms (player). Attended all scheduled meetings and contributed into conversation when wanting to share thoughts and experiences. Was open to feedback and reported back to the team lead. Built mini sights to test functions that can be implemented into ours.</p> |
|-----------------|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

