CS 30700

MelodyMunk

Team 24: Sprint 3 Planning Document

Table of Contents

Sprint Overview **4**
Risks and Challenges 4

Current Sprint Details **5**
User Story #1 (Facebook) 5
Acceptance Criteria: 5
User Story #2 (Explicit vs Non Explicit Songs) 7
Acceptance Criteria: 7
User Story #3 (Restrict artists and genres) 8
Acceptance Criteria: 8
User Story #4 (Basic Youtube) 9
Acceptance Criteria: 9
User Story #5 (Youtube search/add) 11
Acceptance Criteria: 11
User Story #6 (QR Code and Chats) 13
Acceptance Criteria: 13
User Story #7 (Social Media Sharing) 15
Acceptance Criteria: 15
User Story #8 (Room Settings) 16
Acceptance Criteria: 16
User Story #9 (Song Suggestions) 18
Acceptance Criteria: 18
User Story #10 (Joining Room) 19
Acceptance Criteria: 19
User Story #11 (Mobile Interface) 20
Acceptance Criteria: 20
User Story #12 (Playlist) 21
Acceptance Criteria: 21
User Story #13 (Spotify Search) 23
Acceptance Criteria: 23
User Story #14 (Token Management) 24

Acceptance Criteria: 24
User Story #15 (Chat Announcements) 25
Acceptance Criteria: 25
User Story #16 (DJ tools) 27
Acceptance Criteria: 27
Remaining Backlog 29
Functional Requirements 29
Non-functional Requirements 31

Sprint Overview

Team Leader: Avnish Bablani

Meeting Schedule: Tuesday/Thursday 3pm, Sunday 4pm

Risks and Challenges

The biggest challenge in this sprint is to making sure that all of our user stories are integrated into one giant project. Currently our project seems like a series of components and combining these components into one cohesive project will be a challenge. Unlike Sprint 1 and 2, during this sprint our project needs to end as one cohesive whole.

Current Sprint Details

User Story #1 (Facebook)

As a general user, I would like to be able to log in with a Facebook account, and have preferences accounts)for rooms saved.

#	Task Description	Estimated Time	Owner
1.	Research Facebook Auth API	2 Hours	Matt
2.	Create Login with Facebook function in Auth Controller	3 Hours	Matt
3.	Create UI for signing in with Facebook	1 Hours	Matt
4.	Join User Entity with Facebook tokens	2 Hours	Matt

Acceptance Criteria:

- Given that the AuthController function is created successfully, when logging in with facebook, it will authenticate the user.
- Given that the UI for logging in with facebook is created, when a user opens login page, they will be able to login with facebook.

- Given that the User Entity and Facebook tokens are joined correctly, when a user logs in using Facebook, they will be logged into their MelodyMunk account.

User Story #2 (Explicit vs Non Explicit Songs)

As a party host, I would like the option to restrict songs to non-explicit versions,

#	Task Description	Estimated Time	Owner
1.	Create Controller that lets you choose whether to allow	3 hrs	Avnish
2.	Add function to check if song is explicit or not	3 hrs	Avnish
3.	Add UI for checking whether a song is explicit or not	3hrs	Avnish
4.	Add unit tests for explicit songs	1hrs	Avnish

Acceptance Criteria:

- Given that a party host chooses not to allow explicit songs, when a user selects songs they should not be able to choose explicit songs.
- Given that a party host chooses to allow all types of songs, when a user selects songs they should be able to play both types songs.
- Given that a certain setting was selected by the admin, when a user is in the room that setting affects them.

User Story #3 (Restrict artists and genres)

As a party host, I would like the option to restrict songs to certain genres or artists.

#	Task Description	Estimated Time	Owner
1.	Research Spotify API and song filtering	3 Hours	Matt
2.	Create Song Restriction Entity and DB Table	2 Hours	Matt
3.	Create Song Restriction function in RoomController	3 Hours	Matt
4.	Create UI for adding a new song restriction	2 Hours	Matt

Acceptance Criteria:

- Given that song restriction entity and DB table are created, when I add a row in the DB for a song restriction, I will be able to pull restrictions with a query.
- Given that Song Restriction function is added to the RoomController, then when I submit a song restriction, it will be added to the room song restrictions.
- Given that the UI for song restrictions is implemented, when a user is in a room, they will be able to submit a song restriction.

User Story #4 (Basic Youtube)

As a general user, I would like an embedded Youtube player with the option to be hidden, that plays music from a playlist.

#	Task Description	Estimated Time	Owner
1.	Research Youtube API for getting embedded player	2 hr	Brooks
2.	Create test page for a player using a static playlist	2 hr	Brooks
3.	Allow player playlist to be specified	3 hr	Brooks
4.	Allow player to be hidden and provide custom UI for controlling	3 hr	Brooks

Acceptance Criteria:

- Given that a user is on the page, an embedded youtube player will play a video.
- Given that a user is on the page, there is an option to specify a URL to a playlist and have the player start that playlist.
- Given that a user is on the page, there will be a button that hides the Youtube player from view while still playing.

- Given that the Youtube player is hidden, there will be a set of playback control buttons that control the player.

User Story #5 (Youtube search/add)

As a general user, I would like to be able to search for and add songs to the Youtube player playlist.

#	Task Description	Estimated Time	Owner
1.	Create a search box that can show youtube videos	4 hours	Brooks
2.	Enable search box videos to be added to playlist	2 hours	Brooks
3.	Set up all clients in a room to open a websocket to receive playlist updates	2 hours	Tristan
4.	Dispatch playlist updates to all clients	3 hours	Tristan

Acceptance Criteria:

- Given that a user types a query in the Youtube search box, it will display a list of search results from Youtube.
- Given that a user wants to add one of the searched videos to the playlist, there will be a button that adds it to the playlist.
- Given that a user is on the room page, it will attempt to connect to the websocket server for playlist updates.

- Given that a room’s youtube playlist is modified or updated, users connected to the room will get the updated playlist and see the UI reflect it.
- Given that a user is in a room, they can see a list of the songs that are in the queue.

User Story #6 (QR Code and Chats)

As a general user, I would like to have a chat and QR display for a room.

#	Task Description	Estimated Time	Owner
1.	Take last sprint's QR code display and insert into the room view	4 hr	Brooks
2.	Take last sprint's chat work and insert into room view	4 hr	Brooks
3.	Refine websocket behavior with reconnecting and error handling	4 hr	Brooks
4.	Make separate chat for each room	3 hr	Brooks

Acceptance Criteria:

- Given that the chat is open, if the chat websocket disconnects, it will attempt to reconnect.
- Given that the chat is opening, it will attempt to connect to the chat websocket server, and if an error occurs display the error and the option to attempt to connect again.
- Given that a user is on a room page, they will find a button to display a QR code for the room for quick sharing.

- Given that a user wants to print a QR code, they can go to a printing page from the QR code display in the room page.

User Story #7 (Social Media Sharing)

As a general user, I would like to notify people that I have started a party.

#	Task Description	Estimated Time	Owner
1.	Give option to promote party on Twitter and Facebook	3 hours	Shrikar
2.	UI for allowing users to add their Facebook, Twitter, Snapchat and Instagram handles on their user profile page.	3 hours	Shrikar
3.	UI for users to add info regarding their favorite genres of music and artist recommendations.	3 hours	Shrikar
4.	UI for users to add their favorite music videos (i.e. YouTube)	4 hours	Shrikar

Acceptance Criteria:

- Given that a user would like to start a party, enable the ability for users to promote their party via social media platforms twitter and facebook.
- Given a user’s about me page, allow users to add their personal facebook/twitter/snapchat/instagram handles, so that other users can contact them if need be.

- Given a user’s about me page, allow users to add information regarding their favorite genres of music and favorite artists.
- Given a user’s about me page, allow user to add their favorite music videos from sites such as YouTube etc.

User Story #8 (Room Settings)

As a party host or admin, I want to have a central place for changing room settings.

#	Task Description	Estimated Time	Owner
1.	Amalgamate all settings into one easily accessible area on the UI	3hr	Elizabeth
2.	Move all settings into one file in our class structure	2hr	Tristan
3.	When attempting to add a user into a room, verify the user is allowed based on the current restriction mode	1hr	Tristan
4.	Connect backend and front end admin promotion	2hr	Tristan
5.	Unit testing to ensure restrictions hold	1hr	Elizabeth

Acceptance Criteria:

- Given a host has a variety of room settings, if they wish to change those settings, then there is one spot in the UI wherein all those changes can be made.
- Given that a developer is working on user privileges, if they need to add another privilege that can be toggled, then they know precisely where to add it to, lowering their confusion and confusion for others down the road.
- Given that a user attempts to join the room, if they do not have the privilege to do so, then they will be given an error.
- Given that a host/admin attempts to promote a user, if they have the privilege for it, then the front end sends the necessary information to the database allowing the specified user to now have the privileges of an admin.
- Given that a user does not have host or admin privileges, if they attempt an action outside of what is defined by the host/admin, then that is checked, disregarded, and potentially the user is given an error.

User Story #9 (Song Suggestions)

As a general user, I would like suggestions for songs that I may be interested in adding.

#	Task Description	Estimated Time	Owner
1.	Create Song Suggestion Entity and DB Table	2 Hours	Matt
2.	Create Suggestion Controller for handling suggestions	3 Hours	Matt
3.	Create UI for submitting song suggestions	2 Hours	Matt
4.	Create UI for admins to see song suggestions	2 Hours	Matt

Acceptance Criteria:

- Given that Song Suggestion and DB table are created correctly, if I add a DB row for a song suggestion I should be able to create a query to pull all song suggestions.
- Given that the Suggestion Controller is implemented successfully, when a song suggestion is submitted, it will create a new row in the DB.
- Given that the UI for submitting a song suggestion is created, when a user is in a room, they will be able to submit a song suggestion.

User Story #10 (Joining Room)

As a party guest I want to be able to join a room.

#	Task Description	Estimated Time	Owner
1.	Create database relations to track a room and users inside	3 hours	Elizabeth
2.	Create helper functions to update room attributes and add/remove users	3 hours	Elizabeth
3.	Connect UI to join and create rooms to backend so they are fully functional	3 hours	Elizabeth
4.	Unit testing of joining and creating rooms	2 hours	Elizabeth

Acceptance Criteria:

- Given that a user joins a room, the database will show them belonging to the room.
- Given that a user is on the join page, the page is fully functional.
- Given that a user is on the create page, the page is fully functional.

User Story #11 (Mobile Interface)

As a user, I would like to have a mobile-friendly website interface.

#	Task Description	Estimated Time	Owner
1.	Make UI for mobile friendly room page with slide out tabs	2 hours	Elizabeth
2.	Make UI for mobile friendly chat tab in the room page	3 hours	Elizabeth
3.	Make UI for mobile friendly playlist tab in the room page	2 hours	Elizabeth
4.	Make UI for mobile friendly room info tab in the room page	2 hours	Elizabeth

Acceptance Criteria:

- Given a user is on the room page, the room will have a tab bar at the top allowing the mobile user to switch between room info, chat, and playlist.
- Given a user is on the room page, the information on the screen will be visible with minimal scrolling.

User Story #12 (Playlist)

As an admin, I would like a playlist feature that continues playing songs even when the playlist empties.

#	Task Description	Estimated Time	Owner
1.	Create playlist websocket for youtube and spotify	4 hours	Tristan
2.	Check if playlist has finished all songs in the playlist	2 hours	Tristan
3.	If playlist is empty start playing radio based off last song to be played.	3 hours	Tristan
4.	Privileged users given the ability to remove song from the playlist queue	4 hours	Tristan
5.	Connect backend ranking system with front end user recommendations	3 hours	Elizabeth
6.	Unit testing for playlist websocket and guest recommended songs	3 hours	Elizabeth

Acceptance Criteria:

- Given that a song is added with either youtube or spotify the necessary information for the song is sent through a websocket to all in the room.

- Given that the playlist empties, if a host/admin does not add another song, then either the autoplay feature on youtube or the radio feature on spotify is utilized to ensure no stoppage of music.
- Given that a song is no longer desired in the queue, if a host/admin has the privilege to do so, then they can remove that song from the queue.
- Given that a user recommends a song, if others would like to vote on it, then the UI is available to allow for voting and songs to correspondingly shift in ranking.

User Story #13 (Spotify Search)

#	Task Description	Estimated Time	Owner
1.	Make controller that handles spotify search calls	1 hrs	Avnish
2.	Create UI to make searches	4 hrs	Avnish
3.	Create UI to display and select results of the search	4 hrs	Avnish
4.	Testing for spotify search	1 hrs	Avnish

Acceptance Criteria:

- Given that a user gives a query, when the user views the page relevant results will be shown from Spotify.
- Given a user clicks on a search result, then the playlist is updated with that songs added.
- Give a user wants to search only tracks, artists, or albums, then the search results will be narrowed to only the following results.

User Story #14 (Token Management)

#	Task Description	Estimated Time	Owner
1.	Research ways to store tokens in database	2 hours	Avnish
2.	Create Token Entity	3 hours	Avnish
3.	Store Tokens in Database	3 hours	Avnish
4.	Unit Test and Testing for Token Management	2 hours	Avnish

Acceptance Criteria:

- Given that an access token is received, then it will be stored in the database.
- Given that a refresh token is received, then it will be stored in the database.
- Given that a refresh token is received, then a new access token can be generated from it.

User Story #15 (Chat Announcements)

As a host/admin, I would like to have the ability to use the chat feature to make announcements for all the room guests.

#	Task Description	Estimated Time	Owner
1.	Create UI styling for an announcement that shows on top of a chat	3 hours	Shrikar
2.	Create database entries to record announcements for a room	2 hours	Tristan
3.	Enable chat websocket to send new announcements to users in a room	3 hours	Tristan
4.	Enable room page load to preload chat with any announcements for that room	3 hours	Tristan

Acceptance Criteria:

- Given that an announcement is made it is stored within the database so that even if a user were to disconnect and reconnect the message will stay there when the page reloads

- Given that an announcement is made by a host or admin the message is loaded up and sent to all guests with it clearly denoted different than standard chat messages.
- Given that a user is joining a room for the first time, if announcements had previously been made, then those are preloaded into their chat.

User Story #16 (DJ tools)

As a professional DJ, I would like to showcase my skills and allow users to critique my performance via the MelodyMunk platform.

#	Task Description	Estimated Time	Owner
1.	Create UI to allow DJ's to showcase/advertise their skills.	4 hours	Shrikar
2.	Create UI to share DJ clips	4 hours	Shrikar
3.	Create UI for DJ reviews	3 hours	Shrikar
4.	Create UI for DJ contact info (link to fb / twitter page, snapchat / instagram ID etc)	3 hours	Shrikar

Acceptance Criteria:

- Given that a DJ is connected to the platform, he/she has the ability to showcase/ advertise their skills.
- Given that a DJ is connected to the platform, he/she has the ability to share DJ clips to let users experience their capabilities.
- As a user, guests should be able to leave reviews for DJ's.
- As a user, if I am interested in a DJ's services, I should be able to contact them via a variety of social media platforms (Facebook, Twitter, Instagram, Snapchat)

Remaining Backlog

Functional Requirements

- As a party host
    - ~~I would like to play songs my guests like~~
    - ~~I would like the option to restrict songs to non-explicit versions~~
    - ~~I would like the option to restrict songs to certain genres, artists, or Youtube channels (e.g. VEVO channels)~~
    - ~~I would like the option to show the video feed from the current song if there is one (youtube videos)~~
    - ~~I would like to be able to log in with a Facebook account, and have preferences accounts)for rooms saved~~
    - ~~I would like to be able to restrict room access to specific users~~
    - ~~I would like the option to enable a text chat for the room~~
    - I would like to be able to restrict chat to only admins (for announcements)
    - ~~I would like to be able to specify a list of actions guests and admins have permission to take~~
    - ~~I would like a constant flow of music~~
    - I would like to set a suggested genre of music to be played (classical, pop, kids, dance, etc.)
    - ~~I would like to be able to easily share a link to join a created room (QR code, URL)~~
    - ~~I would like the option to display a visualizer for the currently playing music~~
    - ~~I would like for songs to autoplay if the playlist is empty~~
    - ~~I would like to specify users as admins~~
    - ~~I would like to notify people that I have to started a party via Facebook, Twitter etc~~
    - ~~I would like to set a mood for the room~~
    - ~~I would like to generate a QR code that can be displayed on a screen or printed to share with guests~~

- As an admin
    - ~~I would like to be able to ban specific users from chatting~~
    - ~~I would like to be able to ban specific guests from modifying the playlist or suggesting songs~~
    - ~~I would like to remove undesired songs (Think... What's New Pussycat playing 21 times in a row, Rickrolls, etc.)~~
- As a professional DJ (all sub-bullets if time allows)
    - ~~I would like to advertize my skills~~
    - ~~I would like to share clips of my work~~
    - ~~I would like to have aggregate reviews available from past performances~~
    - I would like live reviews to come in to gauge audience reactions
- As a general user
    - ~~I would like to create and login to an account on MelodyMunk.~~
    - ~~I would like to customize my account and link my Spotify account to it.~~
    - I would like to have niche music available to play (if time allows) (Use Spotify, Soundcloud, Youtube)
    - ~~I would like to have a mobile-friendly website interface~~
    - ~~I would like to be able to search for songs~~
    - ~~I would like to be able to vote on songs~~
    - ~~I would like to upload a profile picture~~
    - ~~I would like to be able to write reviews on the party host or DJ~~
    - ~~I would like to be able to report other party guests to the admins~~
- As a party guest
    - I would like to sign a guestbook to leave a note when I'm leaving the party (if time allows)
    - ~~I would like to suggest songs~~
    - ~~I would like to join a room quickly and easily (QR code or URL)~~
    - ~~I would like my song to be played in a reasonable amount of time~~
    - ~~I would like suggestions for songs that I may be interested in adding~~
    - I would like to be alerted when my song is going to be played next (If time allows)
- As a developer
    - ~~I would like to account for host throttling to ensure users aren't blocked from~~

~~Spotify~~

Non-functional Requirements

- ~~Connection to Spotify music streaming service~~
- ~~Expand to connecting with Soundcloud or Vevo Youtube (if time allows)~~
- ~~Run smoothly without delay from the website in accordance to limitations set by 4g cell service~~
- ~~Secure user profiles and user data~~
- ~~Banning specific users from rooms can utilize IP bans or website profile bans~~
- Website must be able to handle at least 250 users at any given time
- ~~We would like to have high quality music playing (minimum 128 kbps bitrate)~~
- ~~A way to test whether or not the service is functioning properly~~