Brandon Vickrey
Ryan Schneider
Michael Lee
Joe Swanson
Mitch Hanberg
Austin Rauschuber

Sprint Retrospective (Group 9 / Blotto Beats)

Tasks Implemented

Implemented and Working

- **Melodic Lines** - The client is successfully able to generate songs that not only contain standard common chord progressions, but also unique melodies that can play over top of and add variety to chord progressions.
- **Genre(s)** - A second genre “Classical” can now be set with its own parameters. Generator is also modified and improved to take in more genre implementations in the future.
- **Playback** - The client can now successfully play the songs by itself. The program no longer just outputs a midi file that you have to open in an external program. The user can also pause, play, and fast forward the song as they please, using buttons in the client.
- **Generator Improvement** - Changes were made to the overall design of the generator and related classes to allow for both improvements in the current output, and ease of use in developing further output.
- **User Accounts Database** - MySql database, records username, password (hashed), expiration date, and the user token.
- **Password security** - When a user logs via the client, the server sends a secure, randomly-generated token string with an expiration date. The token string is then remembered and used to authenticate uploads and votes, so that the client never stores the users password on the local machine. Once the expiration date passes, a new token must be retrieved from the server by logging in again. Furthermore, the database stores a salted hash of the password rather than the password itself.
- **Advanced Settings** - A new form is can be accessed through the drop down settings menu that includes a plethora of other user settings. These settings will persist through application sessions. Settings include changing the ip address, changing the max songs in the backlog, you can view the current number of songs in the backlog, and have access to the account creation/login menu.
- **Song Backlog** - The client successfully keeps track of previous songs played and the user can freely go back and forth between songs. We also implemented a cap on the amount of songs that could be stored which can be changed in the advanced settings menu.
- **Client Account Creation** - This menu is where a user will log in to the program and can

create a new account. The user will stay logged in between application sessions. This menu will show up on first start of the program on a new computer and can be accessed through the advanced settings menu

- **Client Song List** - The user can pull up a list of the most upvoted songs from the database and click on them to play through the client.
- **Server List of Top Songs** - Returns a list object that has a certain number of songs in it and they're in by top votes.

Implemented and Not Working

- **User Accounts Server -** The server communicates with the database to authenticate users. It supports both registration and authentication upon uploading or voting on a song. At the moment, user authentication upon uploading or voting will sometimes randomly fail. However, both registration and the token system function properly.

Not Implemented

- **Server List of User's Songs** - The Song database should have a column that stores the id of the user that uploaded the song. Then the program should return a list of the songs in the database that have the specified user's id in that column.

How to improve

- Remember to read through the planning document so we don't forget tasks we need to implement.
- Don't wait until the last minute to patch all of the parts together.
- Better communicate with each other during long breaks.