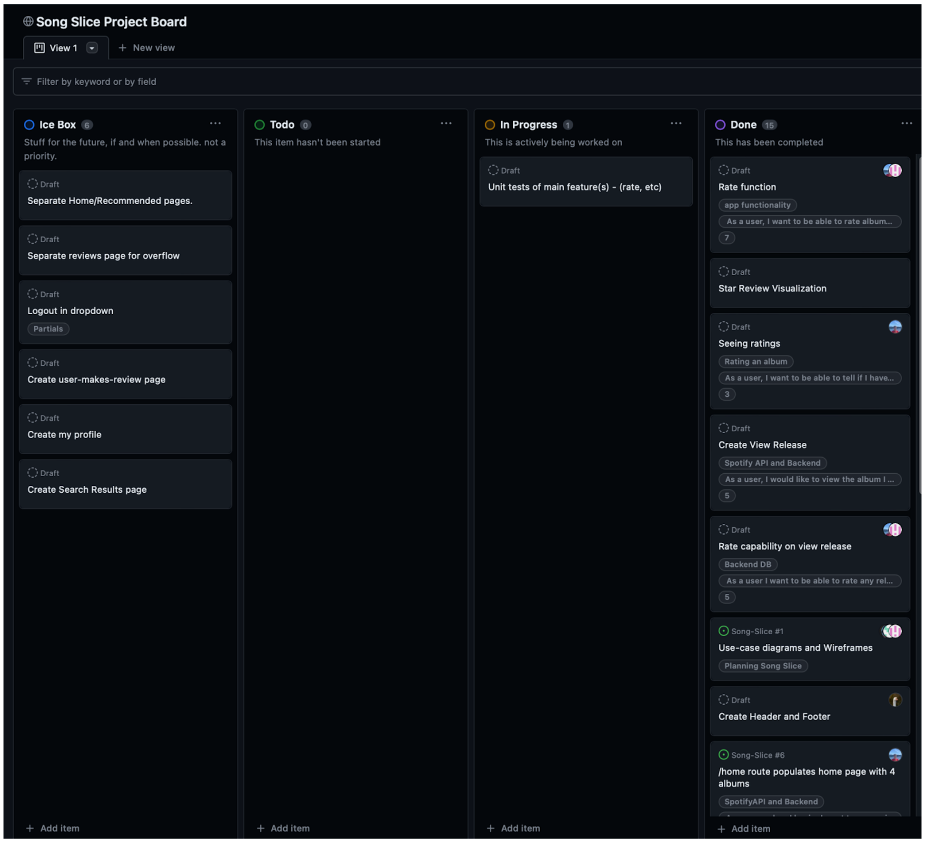
Title: Song Slice

Who: Matthew Long, Lilia Garcia, Chris Delisle, Trey Verhelle

Project Description: Song Slice's purpose is to allow music lovers to find new music, as well as review a set of albums and discuss their opinions. Song Slice provides a communication platform for people to convey opinions. This was executed by allowing users to create personal accounts to access the website. Doing this allows users to have personal written comments and reviews tied to their account. The rating and review system allowed the user to rate an album out of 5, impacting an overall average rating of every user who reviewed the album. This lets the user give their own opinion, as well as see everyones. In addition, users can write reviews below the album, and see other users' reviews. This gives users a way of communicating with other users, sharing opinions, and reading reviews. As for the albums, users have 50 albums to choose from when logged in. They are allowed to review and rate any of the 50 albums. These albums were chosen from what is currently popular, as well as our (the developers) favorite albums. This allows for people to review albums they most likely know, the currently popular ones, but also discover new music as they may not know all of the albums added. This encourages users to find new music, which is part of Song Slice's vision. Users can go listen to an album for the first time, then share about their experience, or they can review an album they already know and love, as they are passionate about it.

Project Tracker: https://github.com/users/MidLogRun/projects/2/views/1



Song Slice Project Board

Ice Box 6
Stuff for the future, if and when possible. not a priority.

- Draft — Separate Home/Recommended pages.
- Draft — Separate reviews page for overflow
- Draft — Logout in dropdown — Partials
- Draft — Create user-makes-review page
- Draft — Create my profile
- Draft — Create Search Results page

Todo 0
This item hasn't been started

In Progress 1
This is actively being worked on

- Draft — Unit tests of main feature(s) - (rate, etc)

Done 15
This has been completed

- Draft — Rate function — app functionality — As a user, I want to be able to rate album... — 7
- Draft — Star Review Visualization
- Draft — Seeing ratings — Rating an album — As a user, I want to be able to tell if I have... — 3
- Draft — Create View Release — Spotify API and Backend — As a user, I would like to view the album I ... — 5
- Draft — Rate capability on view release — Backend DB — As a user I want to be able to rate any rel... — 5
- Song-Slice #1 — Use-case diagrams and Wireframes — Planning Song Slice
- Draft — Create Header and Footer
- Song-Slice #6 — /home route populates home page with 4 albums — SpotifyAPI and Backend

Screenshot of project board above.

Video: https://streamable.com/vmz5ba

VCS: https://github.com/MidLogRun/Song-Slice.git

Contributions:

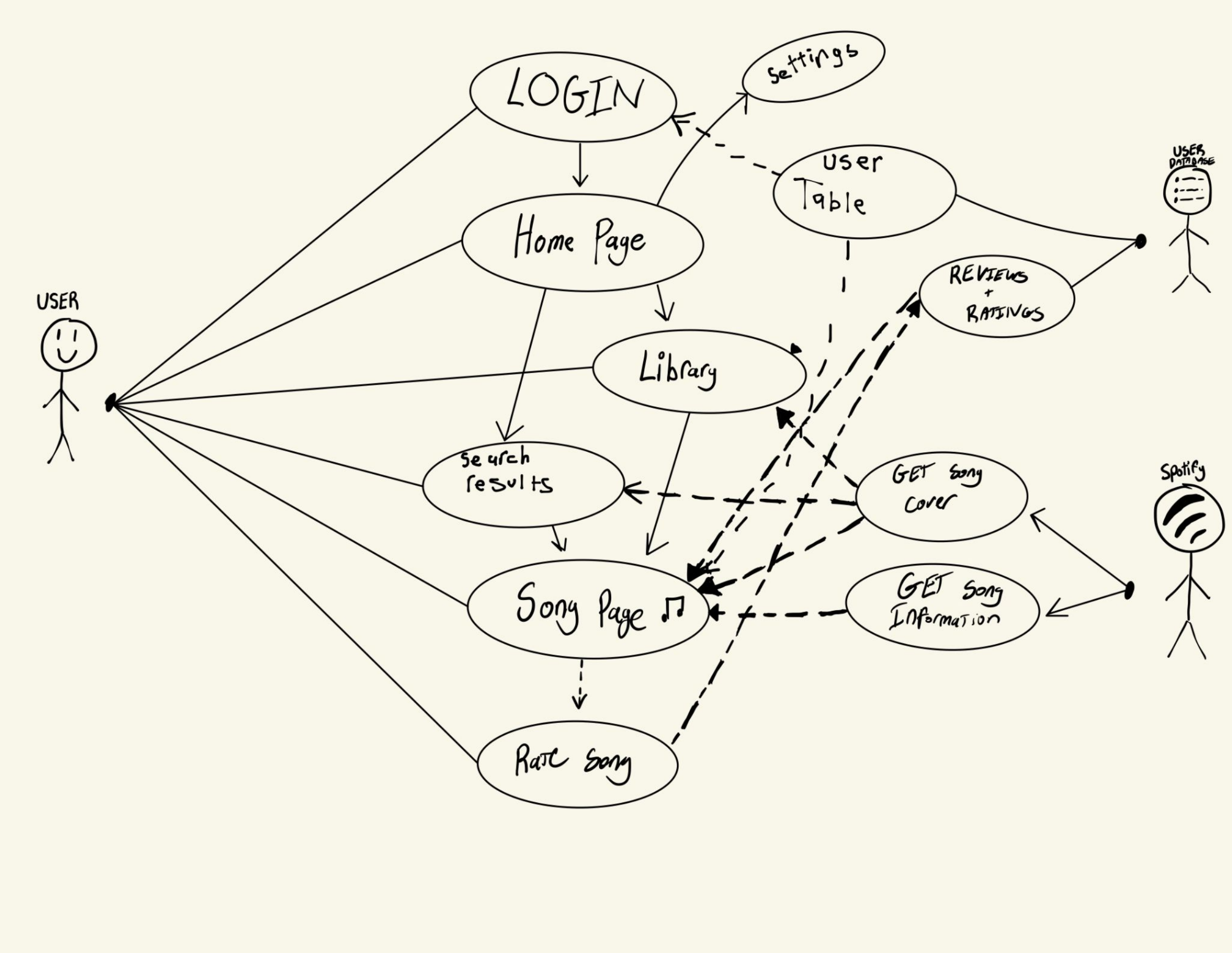
Matt: I developed much of the server-side code and front-end templating for SongSlice including all of the API routes and some of the UI elements. I implemented a Spotify API endpoint handler so that our routes could retrieve the required information about albums to render most of our webpages. I also implemented the rate and review features of the project as the group had planned so that when a user interacts with any of the rating buttons, the page is re-rendered with that user’s rating information as well as the newly calculated average for that particular album.

Lilia: I worked on creating the database model for Song Slice. With the help and ideas from my teammates, we decided what type of information was important to store. I created and implemented the structure to ensure optimal data management. This included creating tables to store data on users, releases, reviews, and user-to-release connections. This helped make sure that we were storing the user's username/password, releases' overall ratings and ID's which gathered album data (i.e., album cover, track list, etc.) from Spotify's API, ratings and reviews created by users, and tracking what releases the user has rated/reviewed to avoid duplicates. Along the way, as some features were being changed, everyone helped decide what we no longer needed to keep track of (i.e., individual songs, etc.), and changes to the database were made accordingly.

Chris: I worked on the testing and css of Song Slice. As the group added backend features, I wrote code and debugged code for testing. I also wrote the UAT test plans for how we would push out and test our program. As front end features were added, I wrote the code for the css. This involved the styling and visuals of what Song Slice looks like. These were the two sections I was in charge of, although I, along with everyone, helped debug each other's code when people were having errors on both the front and back end.

Trey: I first worked a lot on the Figma and overall design brainstorming. Although the entire team worked very hard on this, I feel I put in a lot of effort, and having Figma definitely helped me and my teammates have a uniform design for our group, along with being able to pull some CSS out of Figma directly. Along with this, I created the original carousel for the home page, and further than that, I created most of the home page design. This involved customizing a Bootstrap carousel. Along with this, I did some other, smaller styling things like the Logo design and the menu bar.

Use Case Diagram:



Test Results:

For the UATs, we sent out the application to a few groups of people. Our observations consisted of the following. The users would first go to a register. They did not attempt to login as they do not have an account, although ensuring people without accounts can not log in is still important. Users would create an account, which was successful. Then they typically scrolled around from album to album. After rating them, some logged out, some did not. Although not everyone logged out, it is still an important feature. The reasoning for their actions (registering, logging in, reviewing albums, and logging out), was because that is the focus of the app. Due to our simple display, it was pretty intuitive that that is what the site is intended for. There were, however, deviations from the test plans. Not everyone logged out, however, as stated earlier it is still an important feature to have, as many still used it. We did not observe anyone trying to log in without an account. If we did, we would have made the register suggestion a bit more clear, to save the user from attempting to log in without an account, but because no one tried to, we felt our registration suggestion was sufficient enough. Overall, the UATs allowed us to observe the general process people go through when using our site. Due to our site's intuitive nature, we did not have to change anything after tests.

Deployment:

1. Clone the Song Slice repository
2. Change directory to components/src
3. Run the following command from src directory: docker compose up -d
4. Type in the URL address http://localhost:3000/ in Chrome