

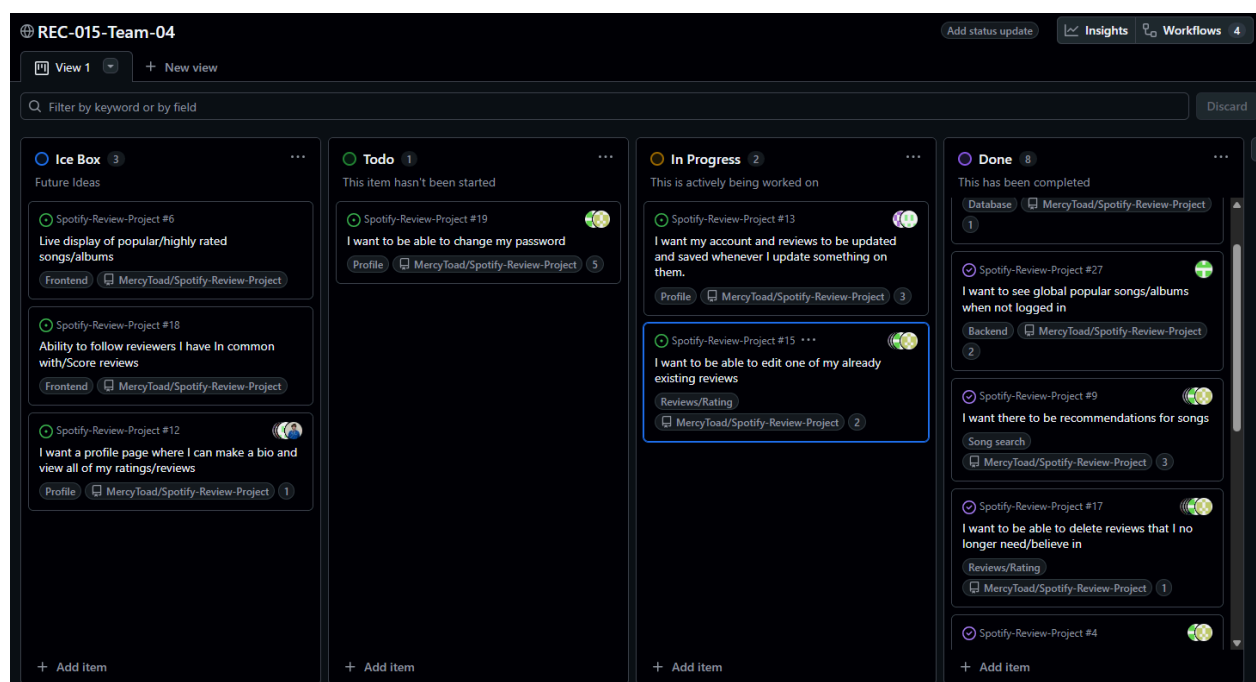
## Project Report:

- **Title:** SongRate
- **Who:** Adie Hofle, Juan Marin, David Todd, Nisha Iyengar, Aayush Sharma, Zack Biehl
- **Project Description:**

SongRate is a web application for discovering, rating, and reviewing songs using real Spotify data. It lets any visitor see a public landing page with popular tracks and a genre-based discovery section. Users can register and log in to access a personal area, where they can search songs by name, open a track's detail (fetched from the Spotify API) and save reviews and ratings. Reviews appear immediately in the UI (client-side render) and are also persisted to the database for aggregates like average ratings and "recent reviews" widgets. Navigation includes a header with live search, a profile menu with login/logout, and an authentication modal. The app combines local data (users, songs, reviews) with Spotify metadata (titles, artists, album art) so it can display rich cover art and details without storing all external content. It's ready to be deployed with Docker Compose (Postgres service + web service) and includes sample data for testing. Overall it provides a lightweight social experience: find music, read community opinions, and leave your own ratings to inform recommendations built from aggregated scores.

- **Project Tracker - GitHub project board:**

<https://github.com/users/MercyToad/projects/3>



- **Video:** 5 minute or less video demonstrating your project. Your audience is a potential customer or person interested in using your product.

<https://youtu.be/ywzYtGES9yc>

- **VCS:** <https://github.com/MercyToad/Spotify-Review-Project>
- **Contributions:**

Adie Hofle: I created the backend for the website, mainly comprised of fetching data from the spotify API to render pages and allow users to review any song. The two endpoints I used frequently were fetching songs by their id, and the search endpoint to parse user input into an array of songs to choose from. I also created the endpoint to create a review, which in turn adds songs to

Juan Marin: I worked on the frontend of the project. To run some tests, I might have touched on a bit of the backend, but nothing beyond that. I worked on the public homepage and the user registration and login. I also suggested the idea of having a public homepage and a private one, which my teammate Nisha mainly worked on.

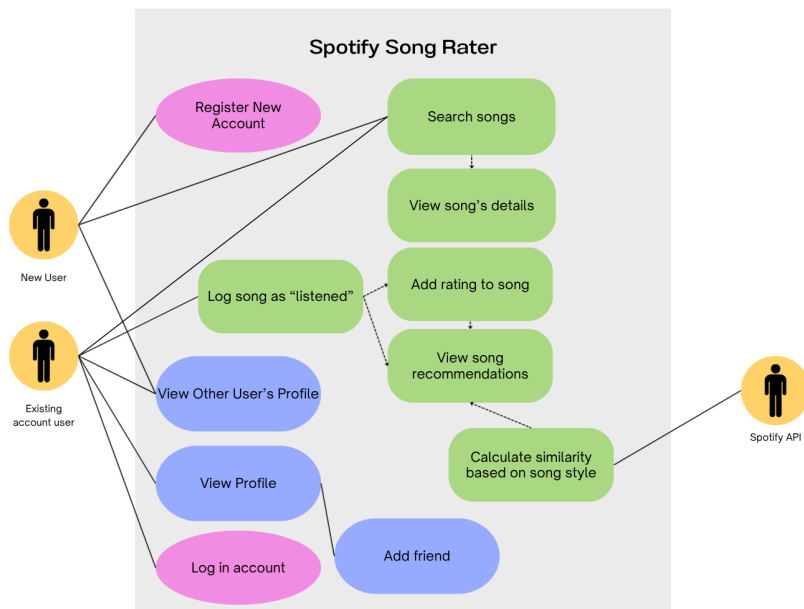
David Todd: Mostly worked on the database and the Spotify API. For the database, we had the basic user information (username, password, etc.) along with the songs and artists from Spotify, and the reviews that the users will be submitting. Also did some of the non-coding specific tasks, such as making sure each person can obtain the Spotify API for testing their code, since the key only lasts an hour, writing meeting logs, and hosting the website with render.

Nisha Iyengar: I worked on the frontend of the project, designing the layout and style of the private pages of the app, such as the home page, discover page, my reviews page, and profile page. I created the style.css file that was the main styling for the pages and made the handlebars templates for items on those pages, such as the song cards and similar. There were difficulties integrating some of my work with the backend, but my teammate Adie helped significantly with the integration.

Aayush Sharma: I worked on the backend of the project, I helped with coding the functionalities for the login system before we made a few changes to its implementation in the program. I also attempted to work on the review system of adding, editing, and deleting which proved to be more challenging than I expected. Other than that I worked on a few smaller snippets of code for minor issues.

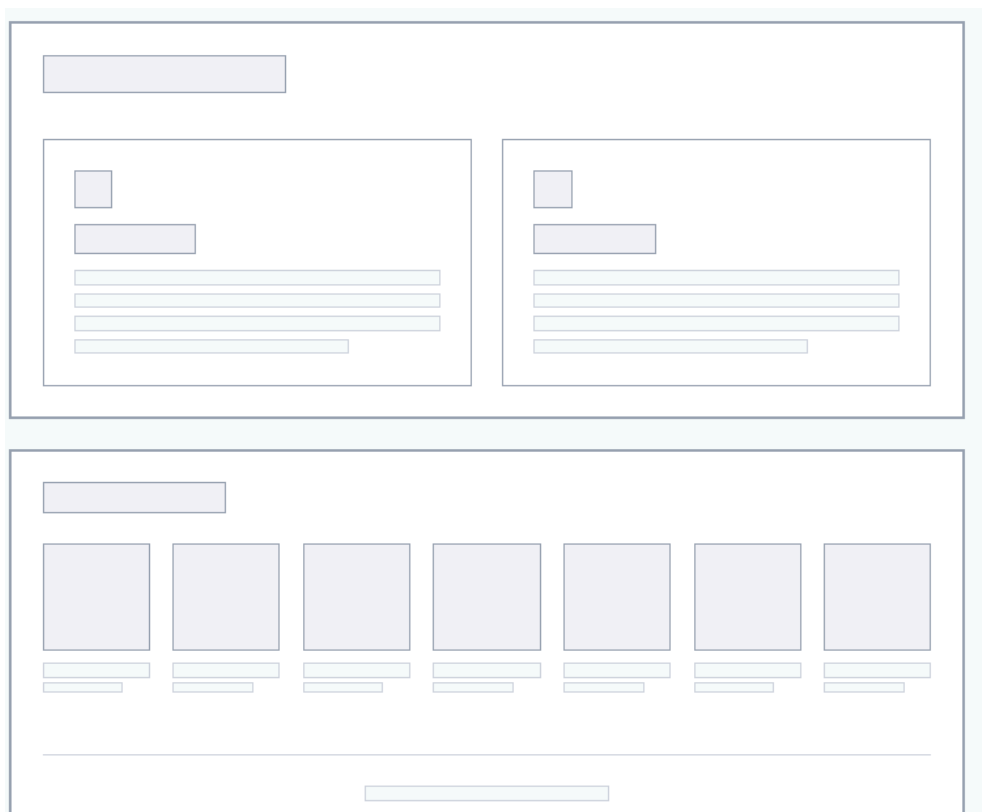
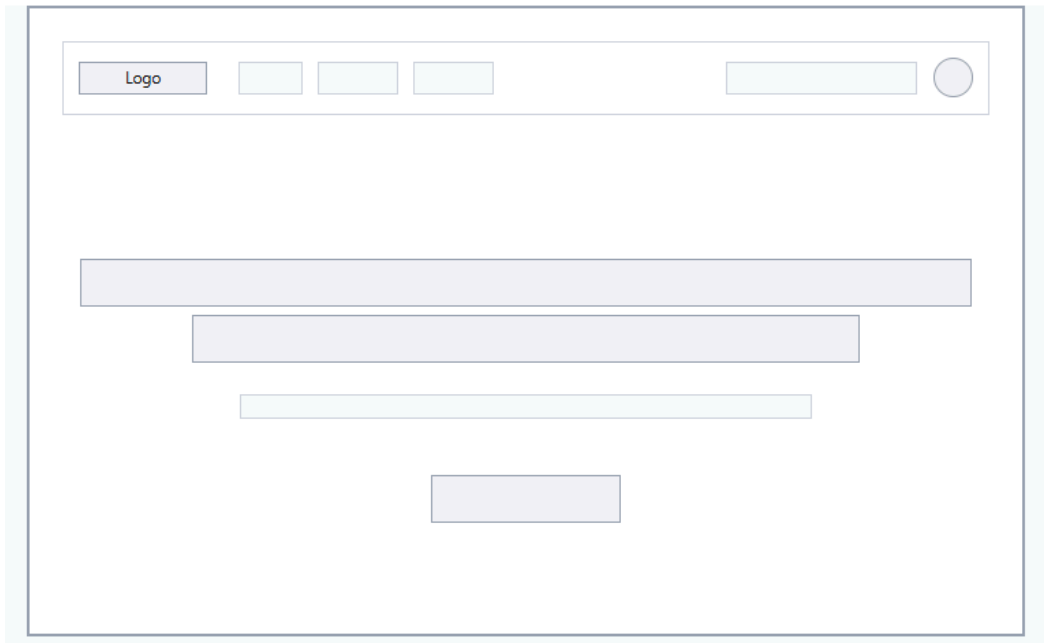
Zack Biehl: I worked on the backend by supporting the integration of the spotify api and database. I helped design tests and connect the spotify api to the database. I helped trouble shoot api connectivity issues during development and worked on getting the spotify access token to refresh automatically.

- **Use Case Diagram:**

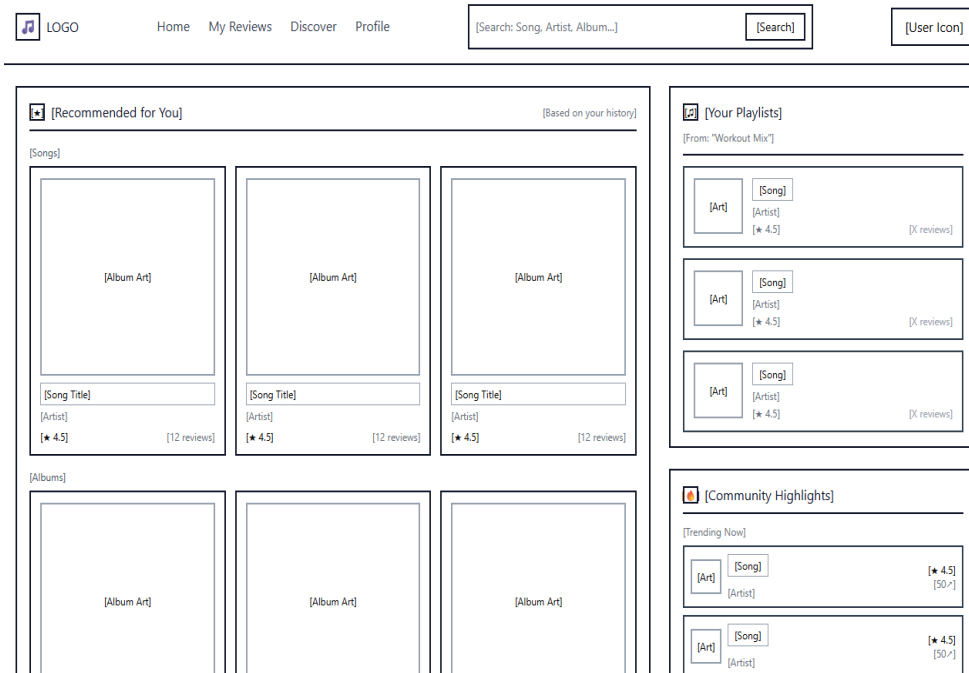


- **Wireframes:**

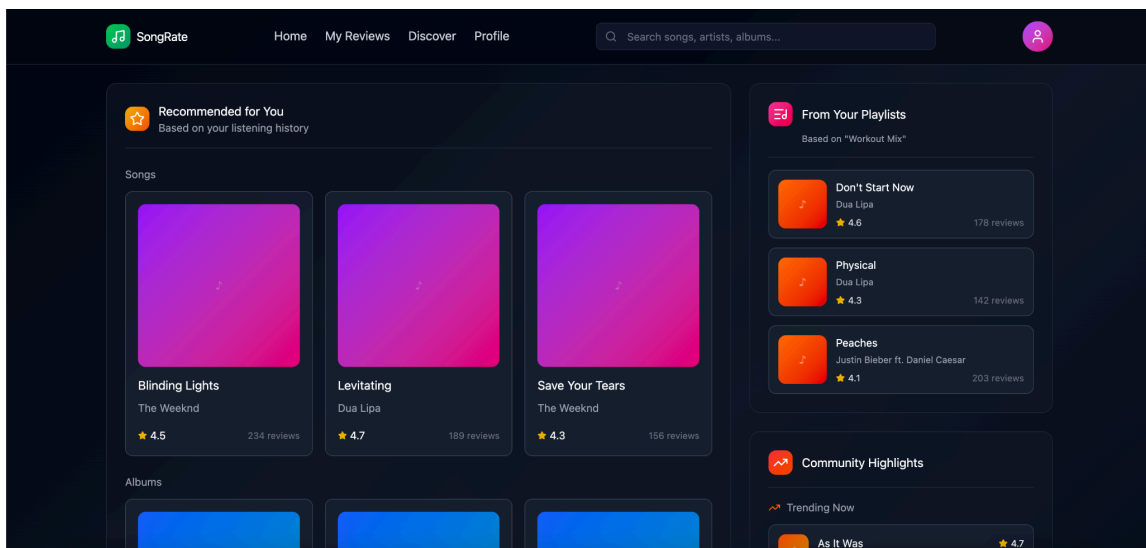
## Public Homepage



# Private homepage



## Stylized homepage wireframe (created with Figma)




## Discover Page


### Browse by Genre

PopHip-HopRockIndieElectronicR&BCountryJazzClassicalMetal

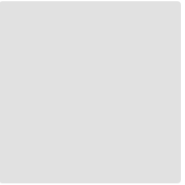
### Top pop Picks for You



The Sweet Escape  
Gwen Stefani  
★ 4.5



Placeholder



Placeholder

### Trending Hits

Top 5 Community Favorites

0	Fantasia in D Minor	★ 5.0
	Wolfgang Amadeus Mozart	
1	Misty	★ 4.9
	Ella Fitzgerald	
2	Shakedown Street	★ 4.9
	Grateful Dead	
3	Distant Lover	★ 4.8
	Marvin Gaye	
4	Friend of the Devil	★ 4.8
	Grateful Dead	

## My reviews







Placeholder



Placeholder



Placeholder



Placeholder



Placeholder



Placeholder







- **Test results:**

Login and Authentication: Everything worked as we wanted. Users are able to create an account and login in with the same username and password. It will also not allow incorrect credentials from moving past the login/register screen.

Rating System: Also mostly works how we want it to. When the user clicks on a song, they will be taken to a form to submit a review for that song, where the user can write about their thoughts on the song and give up to a 5 star rating. It will also let users make a review with only the rating filled in, but will not allow a review to be made if there is no star rating selected. However, we were not able to finish being able to have a user edit their own reviews, but would be added in future iterations.

User Profile: We were unable to finish making the profile page for this project, meaning that users were not able to edit their usernames, passwords or profile picture.

- **Deployment:** <https://spotify-review-project.onrender.com>