

Less than 100: Finding Obscure Music in Spotify

Jamie Lewis, Capstone Project, WATS 4030

Description

This is a website that will allow the user to search for songs on Spotify that have been listened to less than 100 times to find new music. The user will be able to input keywords, such as artists, genres, etc to find songs similar. There will also be a feed of recent searches of songs for other users to see while searching. It will include a play count along with other information about the song. There will be three pages, a home page, the search Spotify API feature page, with the recent feed. And lastly an about page, to give more information about the build and project.

Goals

1. To create a website that pulls data from the Spotify API the produces a song that have been played less than 100 times.
2. Provide information about the project and the website through the about page.
3. The user has a great experience that provides a fun way to find new music.
4. The user will experience a clean, bright website that is easy to understand.

Project Build

The build will be built in four phases. Firstly the planning and research phase which will make for a solid built from the beginning. Secondly, I'll gather the frameworks, content, and images for the built. The three stage of the build will be getting the API, framework and structure of the website. The last stage will be applying stylization, debugging and accessibility.

Technologies

Here is the list of potential technologies that will be used in this project:

- Vue
- Firebase
- Spotify API
- Font Awesome
- Google Fonts
- Bootstrap
- Axios

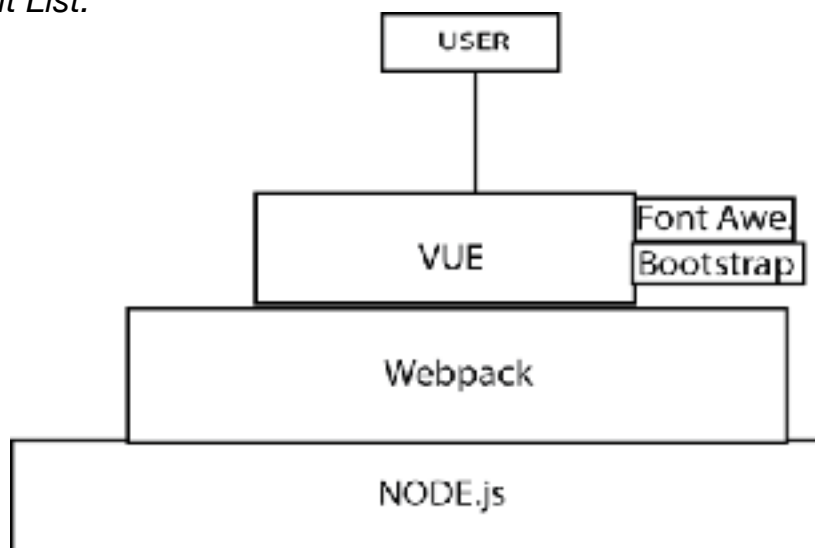
Features

- Intriguing front page with explanation of the website
- User will use search with keywords to find songs with less than 100 playbacks
- Scrolling and updating recently played list
- About page with information on the project and process
- Link to GitHub repo.

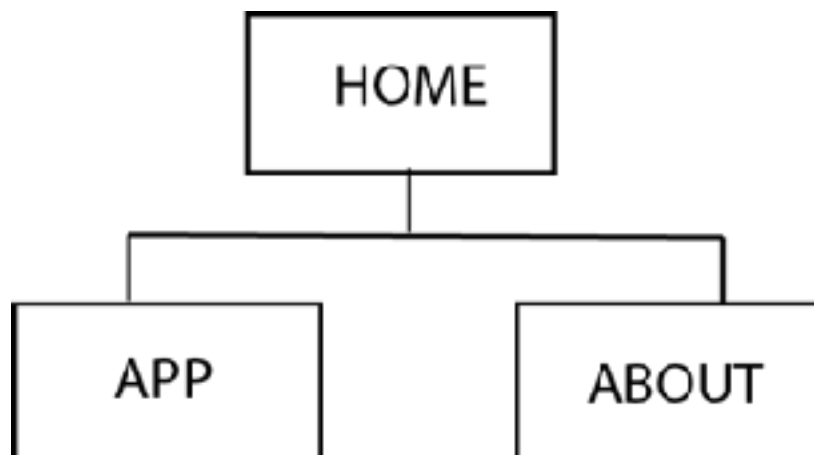
JavaScript

- Keyword Search Bar
- API call
- API recently played list

Component List:



Information Architecture:



Accessibility:

1. Create a site that is clear and has content for the ease of as many users as possible.
2. Add information such as alternative text and make sure the html is clear for screen readers.
3. Check and revisit Krug's book to make sure the site overall is user friendly.

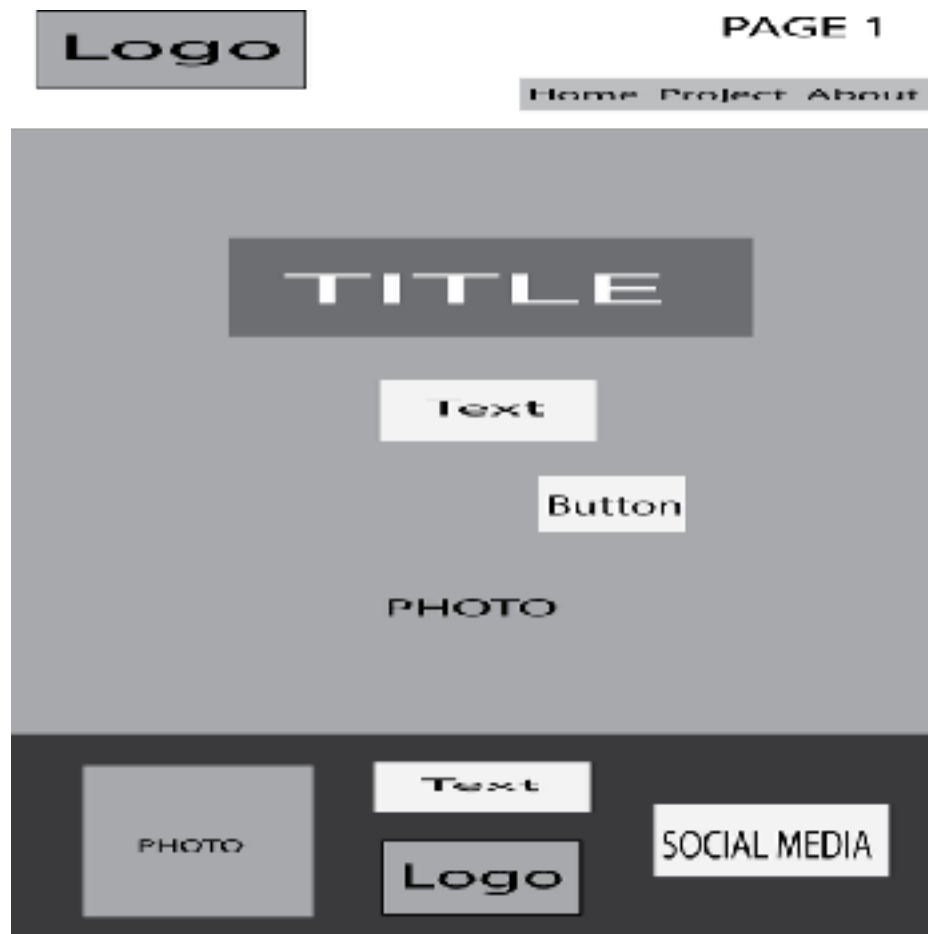
Analytics:

The site will use Google analytics to track traffic and the basic information for the website. There will also be goals set up such as tracking the number of songs played. There will also be goals made around bounce rates and return users to help improve usability in the future.

Server:

Originally I was going to use Github to deploy the project but after some more information, I'll use firebase a free server hosting site to deploy and set up the site online.

Wireframe:





TITLE

Text

Text



Text



Design Comp:

Design Inspiration

Color Palette



Typography
Trebuchet
Myriad Pro



Visual Description:

This site will have a bright colorful design with a clean look that focuses on the playback. It will be a happy design that is playful, yet professional. I will approach the site with pops of color for contrast that will enhance the design but not distract from the overall site. The design will use vectors to create a playful and layered look.

Here are a few source sites:

<http://www.happierlivingtoshiba.com>

<https://loans.no>

<https://www.truedigital.co.uk>

<http://cyclemon.com>

To Do List:

1. Plan Documents: Complete by Week One
2. Gather Content, token, logos, photos: Complete by Week One
3. Set up Repo: Complete by Week Two
4. Complete Dev Environment: Complete by Week Two
5. Set Up Framework, hosting, Html: Complete by Week Two
6. Create API & Call: Complete by Week Three
7. Set up JS features: Complete by Week Three
8. Work on CSS & Visual: Complete by Week Four
9. Set up tracking goals: Complete by Week Five
10. Fix and add any other accessibility traits: Complete by Week Five
11. Test and Debugged: Complete By Week Six
12. Caching and Deploy Final: Complete By Week Six

Perceived Risks:

Here is a list of perceived risks when working on this project:

- The Api call and getting the API to work has been a challenge in the past and something that could be difficult
- The responsive design with Vue. Still researching the best responsive design
- The recently played area of the website is new, making it challenging
- Working with Firebase since I haven't use it to host.