

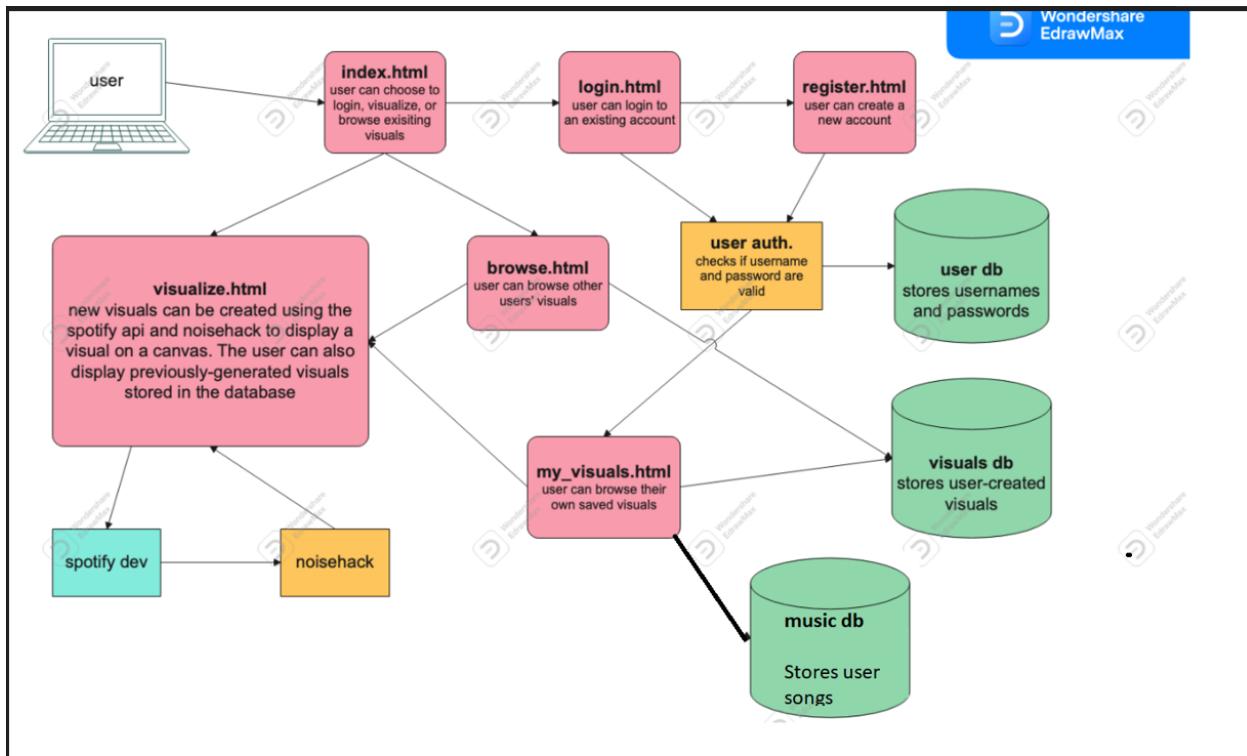
Project Milestone 4

Names: Tyler Huynh, Brady Stark, Andy Strong

Revised List of Features:

- **Unique Music Visualization (1):**
 - Using the data from the Spotify API, we will have the ability to create unique music visualizations for the user.
- **Audio Music Visualization (2):**
 - The main feature of our website will be the audio music visualizer
 - The music visualizer uses javascript in order to run such as defining what and how high the wavelengths are
- **Password Hashing (3):**
 - This will be implemented to protect user's personal information allowing for a safe and effective way for users to use our product.
- **Account Creation (4):**
 - This will be the page where users can create their accounts and connect their Spotify account using the Spotify API
- **Login Page (5):**
 - Login page with the ability to connect your Spotify account to it in order for us to access the data from your account.
 - They can also access their account to view their old music visualizations
- **Browsing Page (6):**
 - This will be a type of forum where users can view other created music visualizations

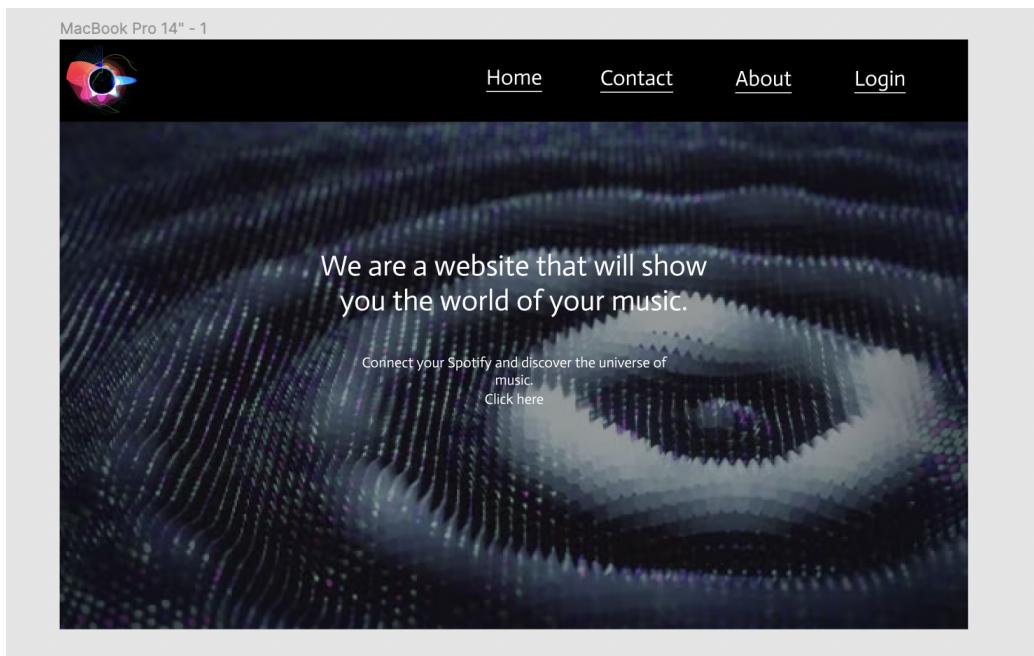
Architecture Diagram:



Front End Design:

- The front end design will consist of the login page, home page, create an account page, account page and the forum page

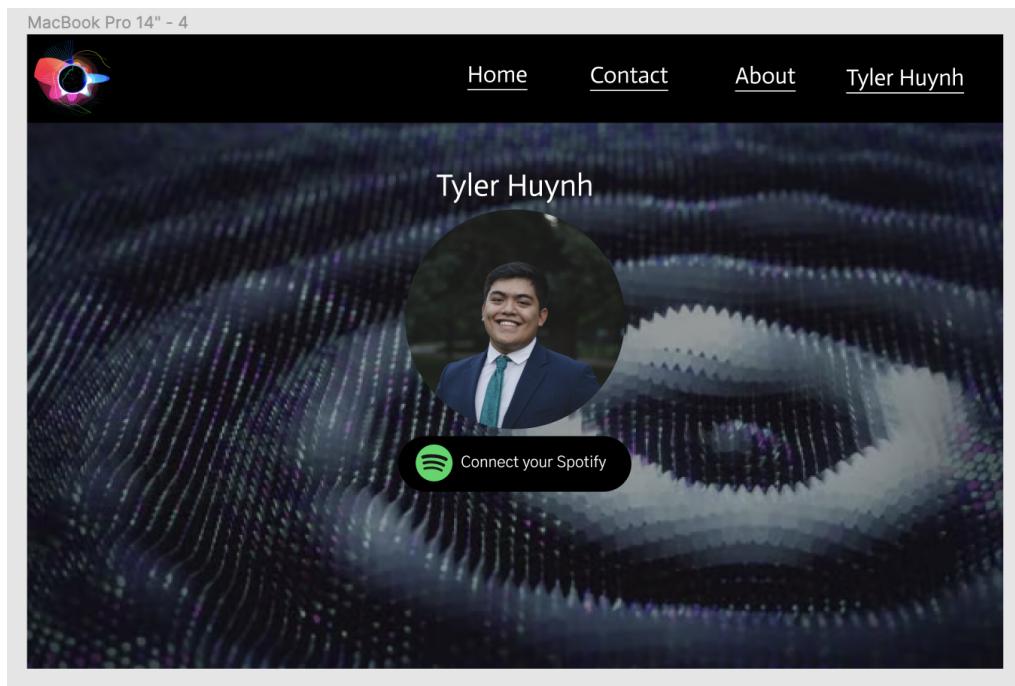
Home:



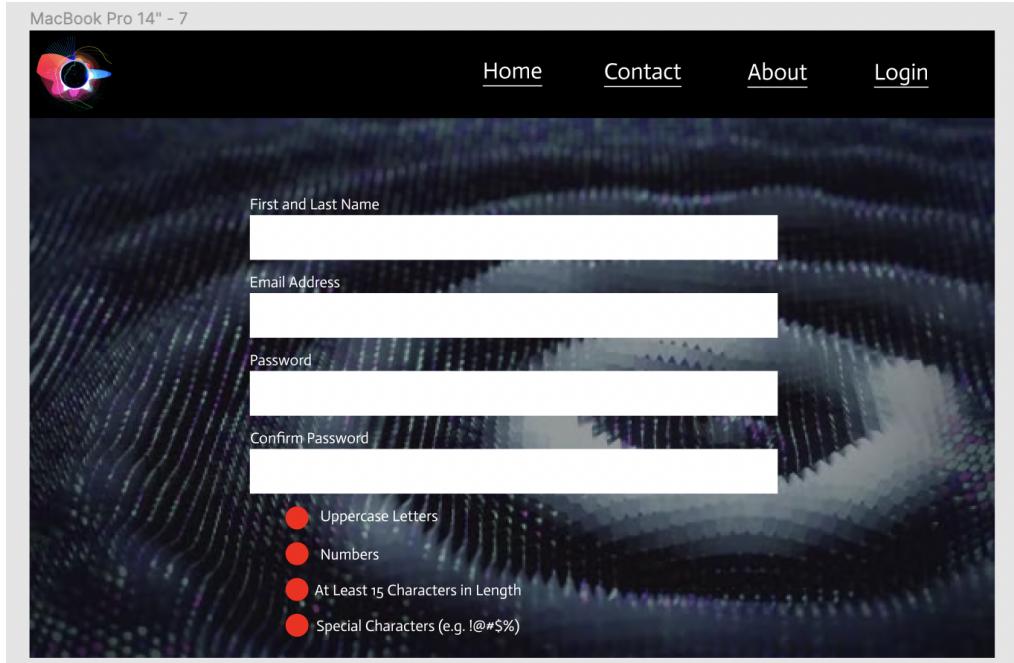
Login:



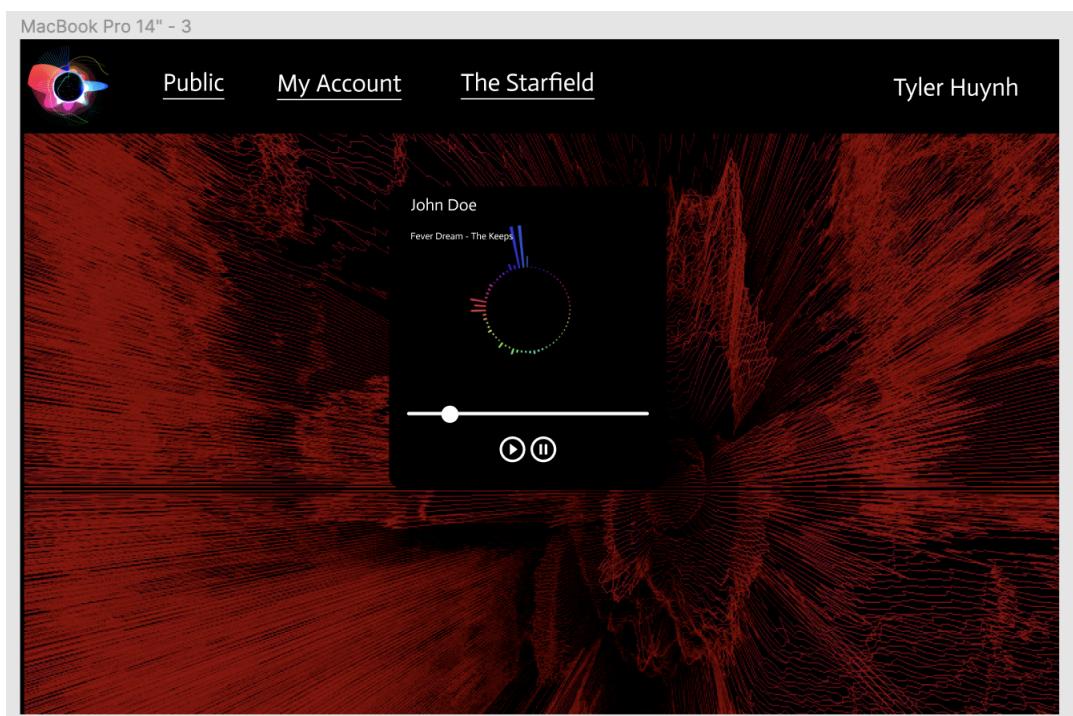
Account:



Create an Account:



Forum:

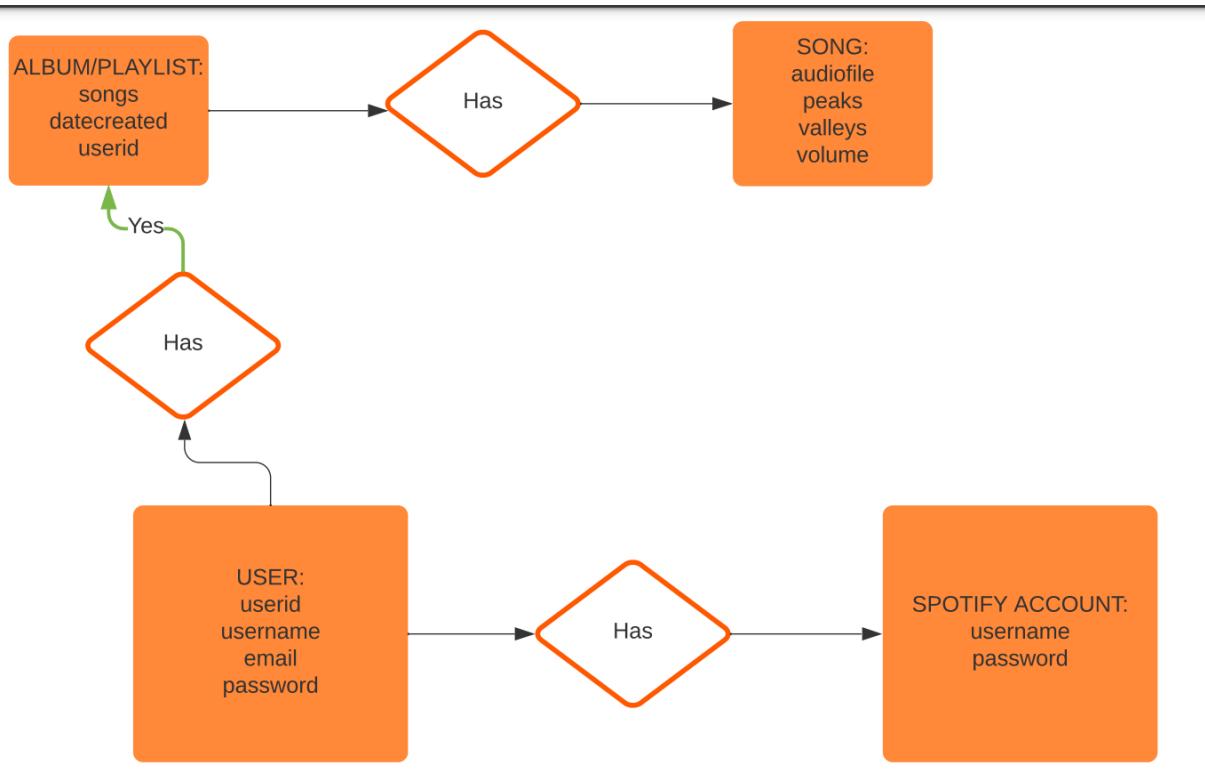


Web Service Design:

- Spotify API (RESTful API)
 - Data Retrieved:
 - Album data
 - Artist data
 - Playlists
 - Personal Account Information
 - Data Passed:
 - Security Certificates
 - Specific endpoints for the information that we need
- Google API (Maybe)
 - This is a maybe, mainly because it is more for stylistic purposes
 - Data Retrieved:
 - Fonts
 - Data Passed:
 - Specific endpoints for the stuff that we need

Database Design:

- DBMS technology used: PostgreSQL



Challenges:

- If we cannot get the Spotify API to work?
 - In the event that the Spotify API is not deployed on our website, then to induce a music visualization we would use local music files on your machine that you can upload and get a visualization from.
- Figuring out the music visualization
 - Getting the music visualization to work is the key part of our website, it has been a little hard conceptually trying to connect the Spotify API and the data from it to the visualization part. In the event that this does not work, we will use local music visualization instead and a basic waveform visualizer.
- The amount of data we will retrieve
 - The amount of data that we get from the Spotify API will be hard to parse through as there is so much data, from the amount of artists a user can have in a playlists or the amount of songs an artist could have. In the event that we can't handle this data, there is most likely a way to see if there is a better way to structure this data and retrieve only the information necessary.

Individual Contributions:

- Tyler: Did the web services design, front-end design, revised list of features, and the challenges that we have faced.
- Spencer: Conducted research into the development of the music visualizer.
- Andy: Created mock-up of database structure and the final architecture diagram with edits