

MUSIC LIBRARY - FRONTEND PROJECT

Tech Stack

React.js, HTML/CSS, Axios, Postman

User Stories

Total Unweighted Project Points: /65

Total Weighted Project Points: /50

Main Stories

- **(5 points):** As a developer, I want to make good, consistent commits.
- **(5 points):** As a developer, I want to use the Create-React-App command to create my React project.
- **(10 points):** As a developer, I want to use AXIOS to consume the Backend API on the frontend.
- **(10 points):** As a developer, I want to display **All Song** data (song title, album, artist, genre, and release date) from the API within a table on the frontend.
- **(5 points):** As a developer, I want to create a minimum of three React components in addition to the App **component** and use them within my React application. (EX: **Header**, **SearchBar**, **MusicTable**)
- **(5 points):** As a developer, I want to have an aesthetically pleasing user interface to ensure a great user experience.
- **(10 points):** As a music enthusiast, I want to be able to **filter** the table of music by album, artist, genre, release date, or title.

- **(10 points):** As a music enthusiast, I want to be able to **add** a song to my database by using Axios to make a **POST** request to the Backend API
- **(5 points)** As a music enthusiast, I want to be able to **delete** a song from my table, initiating a **DELETE** request to my Backend API

Bonus Stories

- **(10 points)** As a music enthusiast, I want to be able to **update** the song data in my table using a modal popup window that initiates a **PUT** request to the Backend API

Checklist

- Make sure your Backend API is set up to work in a Full Stack environment by following the **Configuring Cors in ASP.NET WebAPI Rising Requirements** document.
- Run through the [Setup Setups](#) and get your project ready to begin work.
- Review the [Resources](#) outlined below - be sure to have relevant documentation and references open while you develop!
- Review all videos and documentation. Make sure to have a clear direction regarding the end result of the project as well as all supporting documentation that is available to you.
- Design the React application using a wireframing/prototyping tool like Adobe XD or Figma. Remember that this is a single-page application.
- Build and test the React Frontend using React best practices (components organized into their own folders with external CSS stylesheets, lifting state up as high as possible in component hierarchy, passing data from parents to children as props, etc)

Setup Steps

1. Start by setting up the project using CREATE REACT APP. Be sure to test-run the development server and verify that your application runs in the browser. You will need these components at a minimum:
 - **App.js**
 - **MusicTable.jsx**
 - **SearchBar.jsx**
2. Install the Axios package to your react project by running the command **npm install axios** within your React frontend application.
3. Then, set up your App component to make axios requests to the Music Library API for all songs. Verify that the data is appearing in your console upon component mounting using a console log. Your API request should be in a **useEffect** hook function with an empty dependency array.
4. Once you have confirmed you are getting the data back you can proceed to pass it down to your “**MusicTable**” child component.
5. Within the **MusicTable** component, you will create an HTML table and map over the songs passed down through props to generate the table data rows.
6. In a separate “**SearchBar**” component create a form with one input. This is where the user will type a search term that they would like to filter the table by. Once the form is submitted the value the user entered should be passed to a function on the App component that then filters the songs by the term depending on if that term matches any of the song’s properties.

Resources

Lectures

- Web Services
- Request & Axios
- Promises
- Postman
- Intro To React

Relevant Projects

- Weight Tracker
- Social Feed

Other Resources

- **Student Hub** - All React.js articles, Callbacks & Higher Order Array Methods (JavaScript)

End Result

See **Project Walkthrough - Music Library** video to see what the final result could look like!