

# Promptify

Trevor Ing

**Course:** CSCI 1300 User Interfaces and User Experience

**Faculty Sponsor:** Jeff Huang

**Link:** <https://promptify.vercel.app>

**Tech Stack:** React.js, Vite, NextUI, Express

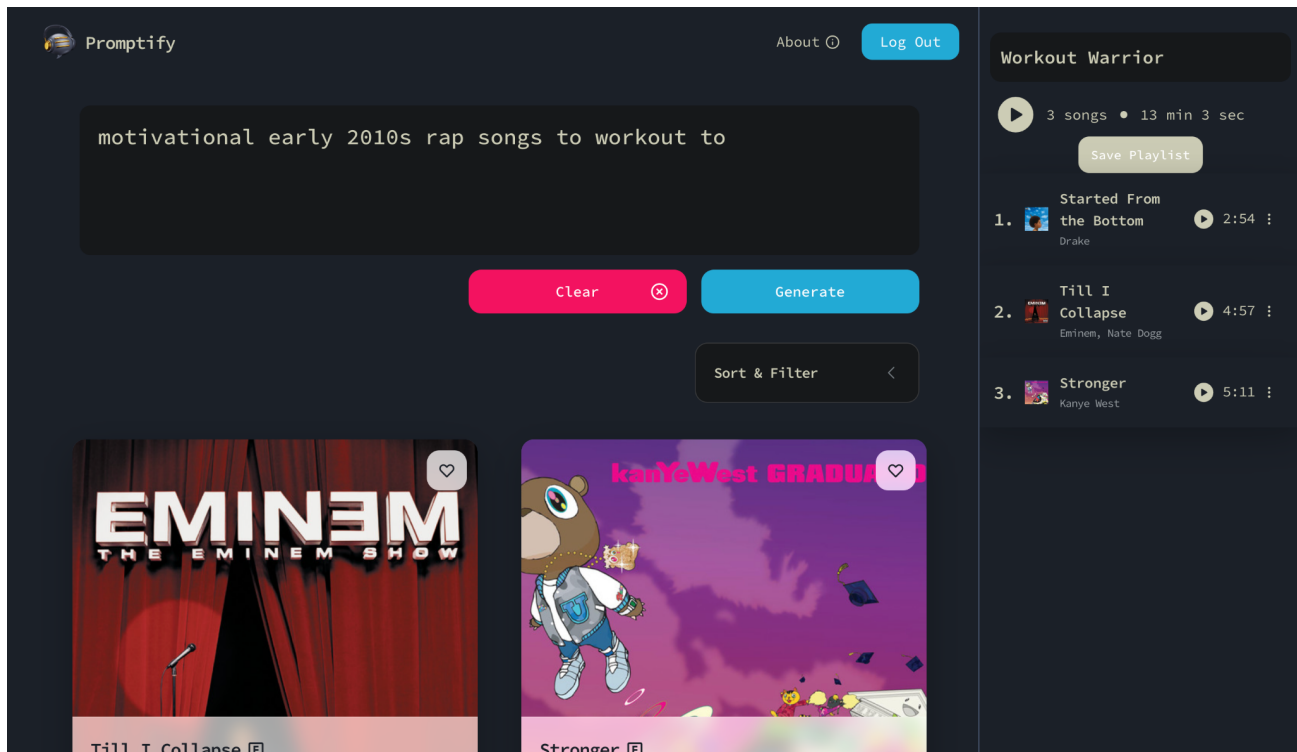
## Abstract:

This capstone project involved extending one of our assignments into “something that has substantial development and design components” and releasing it. I chose to extend my Development assignment, which I had dubbed Playlist Pal, into a revamped app called Promptify. The original required functionality of the Development assignment was to build a React web app that displayed a list of items using UI components and allowed for sorting, filtering, and aggregating these items. In the case of Playlist Pal and Promptify, these items are songs, which can be filtered and sorted by various properties and “aggregated” into a playlist. However, Playlist Pal's complexity and design was quite simple — primarily because it used a hardcoded JSON file of songs and had limited interactivity.

To incorporate “substantial development” into the extension of Playlist Pal, I came up with the idea for Promptify, which adds a nuance to the existing playlist creation functionality by allowing users to input a prompt or description of a specific vibe, scenario, musical era, etc. (or a combination of these criteria) and get back a list of matching songs. This involved feeding the prompt to OpenAI's text-davinci-003 model using their [API](#), parsing and formatting the response, and then searching for the recommended songs using [Spotify's Web API](#). A considerable amount of my additional development went into creating and fine tuning this pipeline from user-inputted prompt to Spotify songs. There are certainly still some quirky results because, despite its advanced capabilities, OpenAI's models occasionally return song names that don't exist or don't match the prompt well and the Spotify API's [search endpoint](#) doesn't always return the desired song despite a well-formatted query (e.g. returning a remix of The Weeknd's Blinding Lights rather than the song itself). In addition, users may occasionally not receive results if OpenAI's servers are overloaded with too many requests.

The remainder of my work came on the design side. Since this assignment was for UI/UX, it was expected that I adhere to good usability and accessibility principles. I focused on creating a strong visual hierarchy between the app's different sections, displaying the song results and playlist in a clear manner that adheres to users' mental models, and choosing a high-contrast color palette. A few additional features that I added to the app included playing individual songs and/or the entire playlist in the browser using [Spotify's Web Playback SDK](#) and the options to save songs and playlists to the user's Spotify library. You can see images of Promptify and Playlist Pal below!

# Promptify



# Playlist Pal

