



Frontend developer Challenge

Introduction

To evaluate your programming skills, we would like to share a fun challenge with you! The main goal will be to create a **simplified version of Spotify using the React Framework**. For this, you will have to use the Spotify API (c.f. setup instructions) to fetch data, add new ones, and update the UI accordingly.

Description of the feature

For this particular Spotify version, we want to let our users create new Playlists, and add tracks to them. Every feature listed below is **mandatory** and should be implemented as part of this challenge.

As a user (currently logged in):

- I should be able to create a new playlist and specify a name and optionally a description.
- I should be able to consult a given playlist's content (every tracks added to it).
- I should be able to select an existing playlist and display its content.
- I should be able to perform a search for any tracks and consult the search results.
- I should be able to add any search results in the currently selected playlist.
- I should be able to remove a given track from a playlist.

Nice to have

If you want to play a bit more with it, you can add one or more of the following features:

- I should be able to edit a playlist's details.
- I should be able to sort the tracks by name, artist, duration, and so on (every track's field).
- Allow the user to reorder tracks within a given playlist.
- Display additional track information (bars, beats, sections, segments, ...)
- Allow the user to switch to a Dark mode
- Want to do even more? Please feel free to surprise us! ;-)

Here is the link to the Spotify documentation where you will find any information you need to manage this challenge with success: <https://developer.spotify.com/documentation/web-api/reference>.

To help you design the app, we provide you some wireframes. We ask you to respect the place of each element. However, feel free to polish the final rendering in order to improve the final result.

Requirements

To carry out this project, we ask you to use the following: **React** (latest version), **TypeScript**, **functional components**, and **Redux (with redux-sagas)**.

To polish the UI a bit, feel free to add any library that suits your needs: ElementUI, Material, Bootstrap, ...

Setup

To use the Spotify Web API, here are the instructions you need to follow:

1. Create a free Spotify account (if you don't have one)
2. Create a new Application by visiting the page: <https://developer.spotify.com/dashboard/applications>
3. Copy your **CLIENT_ID** and **CLIENT_SECRET** within an .env file. You can use a package like **dotenv** (<https://www.npmjs.com/package/dotenv>) to load these additional environment variables into your app.
4. We ask you to authenticate your app by using the Implicit Grant Flow. That will be mandatory to act, such as creating a new playlist.
5. Don't forget to set up the redirect URI in your app settings on the Spotify Developer Dashboard.

Note 1: The code snippet will provide you the code to get a token that must be sent with every request you will make within your app. We don't ask you to implement a login page, only to get this token during your app's initialization and use it in the header of your requests.

Note 2: You don't need any server. You only need to create a React app that will consume the Spotify API.

Timing & how to get help

We will give you a **maximum of 10 days** to send us your code.

If you finish it before, that's perfect! In any case, we will organize a half-way meeting of 15minutes 6 or 7 days after you receive the challenge. The objective is to do a quick ETA of your progress and to answer all the questions you might have.

Our culture at AppTweak is defined by four values, Learn and Play are two of them. We learn from each other, thrive for developing new skills and developing ourselves. But we also know there isn't one best way of doing things, that's why we like to explore. So we value people not afraid of asking for help when they don't know, and willing to try new ideas.

Therefore, if something is not clear, if you are stuck or have a question about the direction to take, **really do not hesitate to reach out to me by email** (romain@apptweak.com).

Finally, in order to share your final work with us, push it on a public repo on Github, Gitlab, or Bitbucket.

Wireframes

Here are the wireframes on which you should base your implementation of the challenge. *Feel free to add your personal touch!*

The wireframe depicts a web application titled "AppTweak - Spotify challenge". The browser's address bar shows "https://tracks.apptweak.com". The interface includes a search bar with the placeholder "Search for a track" and a "Search" button, alongside an "Add new playlist" button. Below the search bar is a dropdown menu labeled "My new playlist" and a text input field for "Selected playlist description". The main content area features three identical track entries, each consisting of a "Cover" image placeholder, a "Track name" field, an "Artist(s)" field, an "Album name" field, and a "Release date" field. A footer bar is located at the bottom right of the page.

