

## Trove Recommerce Engineer Coding Puzzle

As part of the engineering interview process, we like to give a short coding assignment. It is a chance to show your approach to problem solving and your engineering abilities without the pressure and inconvenience of writing code on a whiteboard. For this role, we're interested in your skills at building functional UX integrating data from APIs. Please submit this within a week of receiving. This should only take at most 2 days to complete.

You are free to use whatever server or client-side technology you are most comfortable with (bonus for Node and React!). We must be able to run your solution to evaluate it, so make sure to package all source code. Include a readme file with instructions to setup and run the program. Also describe your design decisions, things you would have worked on if you had more time and anything else you'd like us to know.

We are more interested in the content and usability of the application than in polished visuals. Try to consider the possible use cases for your app, and how do you create a relevant, engaging experience for the user in that context. Feel free to use other services or data if you find it makes the solution better or easier to implement. You are welcome to add any features you like, but emphasize a simple, functional interaction.

Finally, we ask that you not share or post these questions or your solutions (such as to public Github repos or with your friends).

### Puzzle #1: Music Track Search

Create a music track search web interface using the public Spotify API. You will need to use OAuth endpoints so this article may be useful to you:

<https://beta.developer.spotify.com/documentation/general/guides/authorization-guide/>

You can add any features you think are useful, but it should at minimum

- support two or more search filters, including artist and album
- allow sorting by popularity

Spotify web API

documentation: <https://beta.developer.spotify.com/documentation/web-api/reference/>


Spotify API test console: <https://beta.developer.spotify.com/console/>

Desktop UI

## Result List

### Artists

Sort by popularity: Ascending



Justin Bieber

Popularity: 97

It should look good on a mobile device as well (centered).