Layla Assignment Description

Thanks for taking on the Layla coding assignment. Here we'd like to assess your ability to explore API documentation and build a sample application.

Below are some high-level guidelines but we have left some things intentionally vague. We'd like to explore your approach to this. Be creative and have fun.

Time expectations: You can spend as much time as you'd like, but really we expect that you don't spend more than 3 hours on this exercise. We're not expecting high-quality production-ready code but are focusing more on understanding what decisions you took and why.

Tech Requirements:
- Please use React

Preparation:
- Sign up free account at discogs.com
- Create application and get the API key and secret
- Review API docs here

App requirements:

We're creating a song discovery app designed to help us explore music based on various filters.

- When the app loads, it will initially display all Canada albums from 2024.
- I can filter by year, genre, and more to refine the results.
- By selecting an artist, I can view their albums along with the cover art.
- For each album, I have the option to bookmark it in my favorites.

To Submit your App:
- Please push your code to a Github repository and share the URL for that repo with us
- Please make sure your repository contains any instructions or requirements to run your app. Please make this as quick and easy as possible

Bonus Points:
- If you deploy your app somewhere so we can see it live
- Any other creative way you come up with to explore music

- You chose web-based technologies

There might be some unclear requirements or conditions for this assignment, Please feel free to make your own decisions or workarounds, as we are interested in seeing how you handle these situations.

General Evaluation Criteria:

Junior: Fundamental skills, functional implementation, and basic problem-solving.
Intermediate: Practical application, code organization, and some architectural considerations
Senior: Advanced problem-solving, scalability, software engineering best practices and design choices

Evaluation considerations include:
- Code / App architecture
    - API design
    - Any Unit test examples / coverage?
    - Any documentation / flow diagrams?
    - Data structure and design
    - Application is deployed and functional without errors
- Frontend
    - Robust UI/UX
    - Code organization
    - Use of React patterns, state management etc
- Non-technical
    - Making good assumptions about the problem and how to build a user friendly solution
    - Communication / Presentation skills