Technical Assessment

Project Description

You are tasked with creating a web application that displays information about the latest cryptocurrency prices. You will use a publicly open REST API to fetch the data and display it on the frontend.

Instructions:

1. Create a small web application that integrates with a publicly open REST API using Laravel, Vue JS, Nuxt JS (or similar framework), and Tailwind.

2. The web application should display data retrieved from the REST API and allow users to interact with it through the frontend.

3. You have a maximum of four hours to complete the assessment.

4. Submit your code in a GitHub repository along with a covering letter explaining your thought process and design choices.

Requirements

1. You should use Laravel to build the backend API.
2. You should use Vue JS and Nuxt JS or similar framework for the frontend.
3. You should use Tailwind for the styling of the application.
4. The application should display the top 10 cryptocurrencies by market cap on the homepage.
5. The user should be able to click on a cryptocurrency to see more detailed information, such as price, volume, and market cap.
6. The application should have a search functionality that allows the user to search for a specific cryptocurrency by name or symbol.
7. The application should be responsive and work on both desktop and mobile devices.

Expectations:

1. Your code should be well-structured, efficient, and readable.
2. Your frontend design should be responsive, user-friendly, and visually appealing.
3. Your code should incorporate best practices for security and error handling.
4. Your REST API integration should be functional and error-free.
5. Your code should be thoroughly commented and documented.

API:
You should use the CoinGecko API to fetch the latest cryptocurrency data.

You should use the following API endpoints:

- /coins/markets: This endpoint returns information about the top cryptocurrencies by market cap.
- /coins/{id}: This endpoint returns detailed information about a specific cryptocurrency.

You can find more information about these endpoints in the CoinGecko API documentation.

Deliverables:

You should submit a GitHub repository containing your code and instructions on how to get the application up and running. The repository should include:

1. Backend API code built with Laravel
2. Frontend code built with Vue JS and Nuxt JS or similar framework
3. Instructions on how to set up the application locally
4. A README file with a brief description of the application, how to use it, and any other relevant information.

Assessment Criteria:

We will assess your application based on the following criteria:

1. **Functionality**: Does the application meet the requirements and work as expected?

2. **Code quality**: Is the code well-organised, easy to read, and follow best practices? We will evaluate the quality of your code based on its structure, readability, and efficiency.

3. **UI/UX**: Is the application visually appealing, easy to use, and responsive? We will assess the quality of your frontend design based on its responsiveness, usability, and visual appeal.

4. **Use of technologies**: Have you effectively used the specified technologies, and have you integrated the CoinGecko API correctly?

5. **Creativity**: Have you demonstrated creativity and initiative in the design and implementation of the application?

6. **REST API Integration**: We will evaluate the functionality and error-handling of your REST API integration.

7. **Best Practices**: We will assess the extent to which your code incorporates best practices for security, error handling, and documentation.

Technical Requirements:

1. Use Laravel to build the backend of the web application, including REST API integration.

2. Use Vue JS to build the frontend of the web application, including user interaction and data display.

3. Use Nuxt JS (or similar framework) to build the frontend of the web application, including server-side rendering and SEO.

4. Use Tailwind to style the application.

Summary:

You have between two to four hours to complete the assessment. Please let us know if you need any clarification on the requirements or if you have any questions about the assessment.

Good luck, and we look forward to seeing your work!