

Task: Full-stack API-based Development Exercise (Laravel + Vue/React)

The goal of this task is to create a full-stack application using Laravel for the backend API and either Vue or React for the frontend. This exercise focuses on your coding skills, integration with external services, and clean, original code. **Understanding and correctly implementing the requirements is also a critical part of this task.** Please complete the following requirements and submit the work as instructed.

Requirements

1. Backend (Laravel API)

- **Google-based Login:** Implement a login functionality using Google authentication, specifically using Gmail IDs. The login should be API-based and secure.
- **Google Lighthouse API Integration:** Create an endpoint to test website performance using the latest version of the Google Lighthouse API. Your API should:
 - Accept a URL.
 - Accept a platform selection (either "Mobile" or "Desktop") to specify the device type for the Lighthouse test.
 - Return only the "Performance" score for the given URL and platform. Use the following endpoint for this:
`https://www.googleapis.com/pagespeedonline/v5/runPagespeed.`

2. Frontend (Vue or React)

- Build a simple frontend that allows users to:
 - Log in via Google.
 - Enter a website URL in a single text box.
 - Select either "Mobile" or "Desktop" to specify the device type for the performance test.
 - Retrieve and display the performance score from the backend API in a clean, responsive UI.
- For inspiration on the UI and functionality, refer to this example interface: [SEO Site Checkup](#). Your interface doesn't need to match this exactly, but it should be simple, intuitive, and responsive.

3. GitHub Repository

- Create a new GitHub account specifically for this task.
- Push all your code to a public repository under this account.
- Share the repository link along with a short video or screen recording that demonstrates the functionality and results of your solution.

4. Efficiency and Code Quality

- Focus on creating an efficient, fast, and well-structured codebase.
- Do not use any existing code from online repositories, as the originality of your code will be evaluated.

5. Bonus (Optional): Add basic unit tests for key API endpoints in Laravel to demonstrate testing skills.

Timeline

- **Deadline:** Please complete and submit this task within **5 days** from the date it is assigned.

Submission

Once complete, submit the GitHub repository link and the video recording link.