Laravel Job Interview Task Assessment

(After getting the task submission email, you will get 48 hours to submit the project.)

Project Overview

Develop a course creation web page using HTML, CSS, Vanilla JavaScript, and jQuery for the frontend, and PHP Laravel for the backend. The page should enable users to create courses with multiple modules, each containing multiple content items.
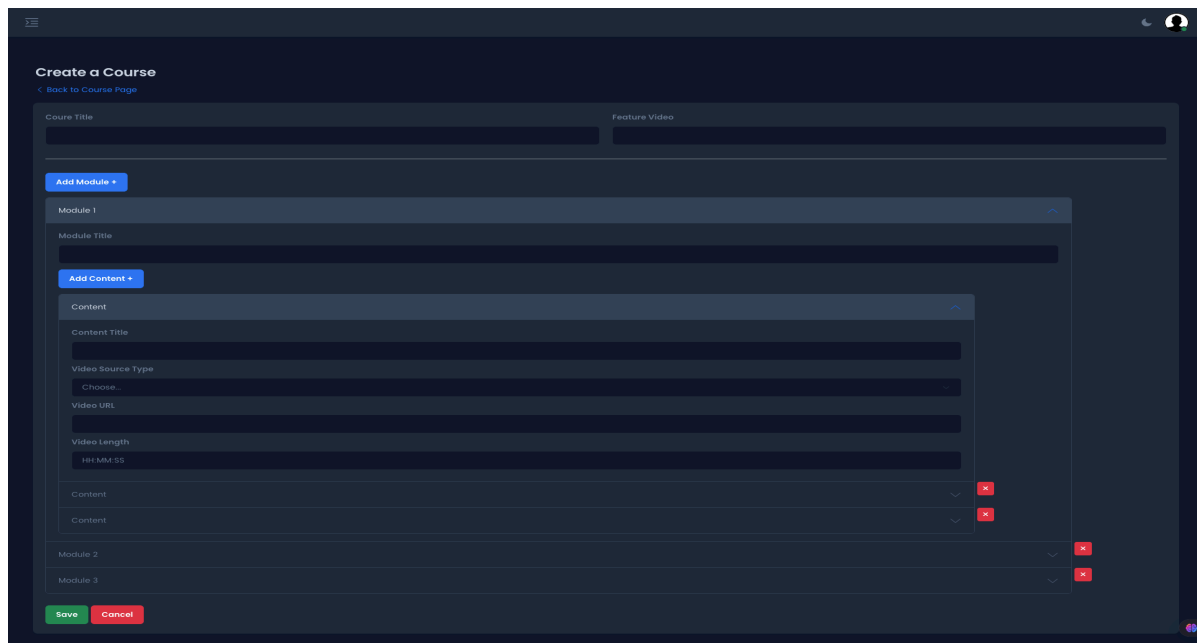
Design Requirements

- **Follow the provided video instructions:**

  ⇲ Loom

[**Tips:** Correctly implement frontend and backend functionalities, including validation and data storage. While the course provides a specific scenario, you can explore alternative scenarios, ensuring these functionalities are present.

Project Structure

Have a look at this scenario:

Project Objective

Building a Webpage includes:

- Course Creation
- Multiple Modules (Inside the course)
- Multiple Contents (Inside the course module)
- Save to the database

Tasks Breakdown

1. Course Creation:
   a. **Create Course Page**: Develop a simple, user-friendly web page for course creation.
   b. **Course Fields**: Include essential input fields such as course title, description, category, and other related details.
2. Module Section:
   a. **Module Creation**: Provide a section to create multiple modules within a course.
   b. **Module Fields**: Each module should have a title and any other relevant information or fields.
   c. **Unlimited Modules**: Allow the user to add as many modules as needed for the course.
3. Content Creation:
   a. **Content per Module**: Inside each module, create a content section where users can add multiple types of content.
   b. **Content Fields**: Each piece of content should have multiple related input fields such as text, images, videos, links, etc.
   c. **Nested View**: Display content within the module as a nested structure, providing a clear visual hierarchy.
   d. **Unlimited Content**: Users should be able to create an unlimited number of content entries within each module.
4. Data Handling and Storage:
   a. **Design Pattern**: Use an appropriate design pattern for the controller, ensuring that the code is maintainable and scalable.
   b. **Frontend and Backend Validation**: Ensure proper data validation is implemented both on the frontend (e.g., field validation, required fields) and backend (e.g., data integrity, preventing invalid entries).
   c. **Database Storage**: Store all the course, module, and content data in the database, with proper relationships between the entities.
   d. **Error Handling**: Implement error handling for exceptions (e.g., failed validation, database errors) and provide user-friendly feedback.

Submission Instructions

- Upload your complete Laravel project to a **public GitHub repository**.
- Share the **GitHub repo link** with us for review.
- Ensure the README includes:
  - Project setup instructions (optional)
  - Screenshots (optional but appreciated)
- Write a project submission email and send it to the selection email.

Once you've submitted your project, please allow us a few hours to thoroughly review your work. We appreciate your time and effort, and we'll get back to you with feedback or next steps as soon as possible.

Thank you and best of luck! 🚀