Open Study College

Full Stack Technical Assessment Brief

The Brief

Open Study College are in need of an API that will allow their course data to be utilised in an upcoming rebuild of their e-commerce store.

PART 1

We would like you to create a simple web service using NodeJs(express) and a database of your choosing to mock an API with the ability to prototype based on the 2 following endpoints.

Courses: `/courses`
Carts: `/carts`

When consuming this API it will be necessary to apply CRUD operations.

An example of what would be expected of all available routes for a courses endpoint would be:

GET

- GET All courses
- GET a specific course based on ID.
- Limit return results eg. `limit=5`
- GET courses in ascending or descending orders (default to asc)
- GET all categories
- GET all courses in a specific category

POST

- Add a new course

PUT, PATCH

- Updating a specific course by ID

DELETE

- Delete a specific course by ID

Open Study College

Full Stack Technical Assessment Brief

PART 2

Using React or a framework of your choosing, please create a simple frontend application demonstrating how your API can be consumed. This may be simply in the form of a react component with basic styling which displays dynamic data from your API service.

Overall Build Guidelines

1. **Please ensure your code is clean, organised and well written**
2. **Please ensure your codebase is well documented via comments and the repo's README. This includes how to set up the service and examples of how to interact with the endpoints.**
3. **Please ensure you utilise a testing library with at least 2-3 examples of how you would test your API service.**
4. ***We want to understand how you work in a production environment. Where you are deviating from how you would act in a production environment because of time constraints please note this in the README file.***
5. **Please present your work in a Git version control repository hosting service such as Github, Bitbucket, GitLab, Codepen, Codesandbox etc and provide a demo of your submission working.**
6. ***We want to respect your time, so this should not take you more than 4 hours. If you run out of time, please document how you would finish this assessment in the README.***