Objective

Our objective is to assess your proficiency in both back-end and front-end web development.

We aim to observe your approach in building a basic full-stack web application, as outlined below, while considering factors such as performance, code structure, error management, and clarity of the architecture.

Duration

2 weeks.

Deadline: 28/08/2025

Task Definition

Build a small web application (backend + frontend) that implements the main features listed below.

You are free to choose additional libraries or structure, but you must respect the core requirements and functionality described.

Main Features

Backend

- Expose an API to manage a list of books.
- Provide endpoints to:
    - Retrieve all books (**with search** and optional filtering).
    - Retrieve a single book by ID (should have a title, an author, genre and a synopsis).
    - Add a book to a global “library” (favorites).
    - Remove a book from the “library”.
    - Retrieve all books in the “library”.
- The “library” is **global**: there are no users or authentication.
- Data should be stored in a database (should be dockerized).
- Written in **TypeScript** and use a clean architecture (DDD + controller–service–repository pattern).
- Include at least one automated test.

Frontend

- Display the list of books from the API
- Include a navigation method with two sections:
    - **Explore** – displays the general list of books.
    - **My Library** – displays the books the user has added.
- Allow adding/removing books from the “library”.
- Show the details of a selected book (author, title, genre, synopsis).
- Display the list of books in the “library”.
- Written in **React** with TypeScript and componentized design.

- Handle loading states, empty states, and basic error management.

Data Source

You should seed the database with mock data

Each book entry should include: id, title, author, genre, synopsis, and optionally year. If the team comes up with additional relevant information to display, they are free to include it.

Submission

Once the task is complete, we will coordinate a call so you can present the resulting app.

You should be ready to explain:

- Your design and architectural choices.
- The main challenges faced during the exercise.
- How you would support potential future functionality.

Doubts

Any doubts you have can be written into the forum of discord.