Full-Stack Developer Technical Assignment

Objective

Build a Full-Stack Blog Post Manager that allows users to **add, view, update, and delete** blog posts, ensuring a smooth and interactive user experience while following best practices in **both frontend and backend development**.

Backend Requirements (API Dev.)

- Develop a **RESTful API** using **C# .NET, Java, Rust or any language**.
- Implement **CRUD operations** for managing blog posts.
- Ensure proper **error handling, logging, and input validation**.
- Adhere to **SOLID principle** and best practices in development.
- Use a **relational or NoSQL database** for data storage.
- Write **unit tests** to ensure API reliability.
- **Bonus**: Implement **containerization** using Docker.

Frontend Requirements (UI Dev.)

- Develop a **React-based user interface** to interact with the backend.
- Implement forms for **adding, updating, and deleting posts**.
- Display blog posts in a **list/grid format**, ensuring a clean and responsive UI.
- Use **React state management** to handle form inputs and post rendering.
- **Bonus**: Implement **React Router** for navigation between views (homepage, add post, view post).

Evaluation criteria

1. **Code structure & best practices**
   Maintainability, readability, and adherence to best practices.
2. **Functionality**
   All CRUD operations should work as expected.
3. **User Experience & Security**
   Proper validation, error messages, and security measures.
4. **Error Handling & Security**
   Proper validation, error messages, and security measures.
5. **Bonus Points**
   Implement additional features (e.g. authentication, microservices, Docker)

Submission Guidelines

- **Backend & Frontend code**
  Push your project to **GitHub** and share the repository link via email.
- **Documentation**
  Include a **README file** explaining how to run the project, design choices, and additional features.

Please stick to the above requirements!

career@fintrellis.com
+230 454 1564
www.fintrellis.com
fintrellis