🛠️ Backend Developer Assignment – RESTful API Task

This assignment is designed to evaluate your backend development skills including API design, data modeling, authentication, error handling, and code structure. You may choose your preferred tech stack – either **Node.js (Express.js)** or **Java (Spring Boot)** – based on your expertise.

🚀 Assignment Overview

You are required to build a simple RESTful API for a **Bookstore Application**. The API will manage a collection of books and users, and allow for basic CRUD operations, authentication, and filtering.

📌 Core Functional Requirements

1. User Authentication (JWT-based):
   - User Signup (email, password)
   - User Login (returns JWT token)
   - Protected Routes for Books (only logged-in users can access)

2. Books API:
   - Create a new book (title, author, category, price, rating, published date)
   - Get all books
   - Get book by ID
   - Update book by ID
   - Delete book by ID

3. Filtering & Search:
   - Filter books by author, category, and rating
   - Search books by title (partial matches)

4. Error Handling:
   - Handle invalid requests with proper HTTP status codes
   - Validate input data

🎯 Bonus Features (Optional but Appreciated)

- Pagination for GET /books
- Sorting by price or rating
- Swagger/OpenAPI Documentation

- Unit or integration tests
- Dockerize your application

🧰 Tech Stack Guidelines

Option 1: Node.js (NestJS Framework Preferred)
- Language: TypeScript (Preferred)
- Framework: NestJS (https://nestjs.com/)
- Database: MongoDB or PostgreSQL (your choice)
- ORM (optional): TypeORM / Prisma
- Apply modular architecture using services, controllers, DTOs

📋 Submission Guidelines

- Ensure your code is modular and follows clean architecture principles
- Use proper naming conventions and project structure
- Use Git for version control
- Include a README file with:
  - Setup Instructions
  - API Endpoints and sample requests
  - Any assumptions or enhancements

📦 How to Submit

You can submit your assignment in either of the following ways:
- 🔗 Share a public GitHub repository link
- 📦 Send a ZIP file of your project directory via google form

🚫 Important Note on Originality

❌ Do NOT use ChatGPT or any AI tools to generate your code.
We will be reviewing the code thoroughly for originality and understanding. Copied or AI-generated code will lead to immediate disqualification.

🧠 Evaluation Criteria

Criteria	Weightage
Code Structure & Cleanliness	★★★★☆
API Design & REST Principles	★★★★☆
Authentication & Authorization	★★★★☆

Database Modeling & Queries	★★★☆☆
Bonus Features	★★★☆☆
Error Handling	★★★☆☆
Originality (No AI Use)	💯 Must

🚀 Good Luck!

We’re excited to see your approach and code style. Treat this as a real-world mini project and make it production-grade. Clean, well-documented, and working code always stands out. Good luck!