

## Pre-Interview Task: Backend Developer

Company: Immutly

Objective:

Demonstrate your ability to design, implement, and document a small backend service. We're interested in your problem-solving skills, code quality, and your ability to create functional, well-structured, and easily understandable solutions.

Task:

You are to develop a RESTful API service for a unique digital product management system, reflecting a simplified version of our B2B SaaS product. The service should manage basic CRUD (Create, Read, Update, Delete) operations for digital products in the system. Each digital product should have at least the following properties:

- Product ID (unique identifier)
- Product Name
- Product Description
- Price
- Availability Status

Technology Stack:

- Node.JS for the main server-side logic
- Java for a secondary service that processes product price updates
- MongoDB for the database
- Any other technologies or libraries you find necessary

Requirements:

### 1. Node.JS Endpoints:

- `GET /products` - Retrieve a list of all products.
- `GET /products/{product\_id}` - Retrieve details of a specific product.
- `POST /products` - Add a new product.
- `PUT /products/{product\_id}` - Update details of a specific product.
- `DELETE /products/{product\_id}` - Remove a specific product.

### 2. Java Service:

- Create a small Java service that receives price update requests, processes them, and then updates the relevant product information in the database.

### 3. Validation:

- Ensure all necessary fields are provided when creating or updating products.
- Ensure the product price is a positive value.

#### 4. Error Handling:

- Handle common errors gracefully (e.g., product not found, invalid data, etc.) and return appropriate error messages.

#### 5. Documentation:

- Provide a README explaining how to set up and run your service, and how to use each endpoint (include example requests and responses).

#### 6. Testing:

- Include basic tests to verify that your service works as expected.

#### 7. Bonus:

- Implement a simple authentication mechanism.
- Implement a filtering feature on the `GET /products` endpoint to search products based on name or price range.

#### Submission:

Submit your code as a repository on GitHub. Include a README file that explains how to set up and run your project, and how to use each of the endpoints. Briefly explain any design decisions you made and what you would do differently or expand on with more time.

#### Note:

- The work should be original and not sourced from online solutions or generated by AI services.
- During the review, you may be asked to explain your code and the decisions you made.