## Coding Assignment (Monorepo Microservice for E-Commerce)

**Scenario:**

You are working on an e-commerce platform using a monorepo structure. The repository contains two main apps:

- **Store Dashboard**: For managing products, orders, and customer interactions (Next.js).
- **Store**: The customer-facing service (Next.js).

You need to create a simple **microservice** that:

1. **Handles product inventory updates** across both the store and the dashboard.
2. The service should interact with a **Postgres** database using **DB Queries or any ORM of your choice (Drizzle preferred)**.
3. The service should expose a REST API to:
    - **Update inventory** when an order is placed.
    - **Fetch inventory** details for the store front.

**Task:**

1. Set up a basic microservice using **TypeScript**, and **Drizzle ORM** that:
    - Receives a POST request to update inventory when a product is purchased.
    - Receives a GET request to return the current inventory of a product.
2. Assume the database structure is as follows:
    - `products` table:
        - `id` (uuid, primary key)
        - `name` (varchar)
        - `inventory_count` (integer)
3. Upload this task in a git repository and share the same on

**Key Constraints:**

- Implement efficient error handling, particularly for cases where inventory updates would fail (e.g., inventory count going negative).
- Implement unit tests to test your functions. The code should be easy to interpret.
- A function should be less than 60 lines of code. Please avoid any duplication of code.
- Please don't add any kind of authentication, just a simple counter on store for the current inventory and an input on the dashboard side.
- Show how the microservice would be integrated into a **Turbo** monorepo with both dashboard and store running simultaneously.
- You are not expected to create a user interface neither you'd be judged on it, however, it would be good for presentation purposes.