

Assignment: Nest.js - Intermediate

Project Overview:

You are tasked with building a **Customer Management System** (CMS) using Nest.js, following the Domain-Driven Design (DDD) pattern. The application will include basic CRUD operations for customers and orders, interaction with a third-party payment service, and the implementation of CQRS (Command Query Responsibility Segregation). The project should be organized using an `nx-monorepo` structure.

Objectives:

- DDD Pattern Implementation:**
 - Identify and define the core domains: `Customer` and `Order`.
 - Implement these domains in their respective modules.
 - Ensure proper separation between different layers: Domain, Application, and Infrastructure.
- CRUD Endpoints:**
 - Sign-In:** Implement sign in endpoint to feature sign-in process returning a JWT Token.
 - Customers:** Implement CRUD endpoints for customer management. (Authentication required)
 - Orders:** Implement CRUD endpoints for managing orders associated with customers. (Authentication Required)
- Interaction with 3rd-Party Payment Service:**
 - Integrate with a mock third-party payment service (e.g., Stripe) to handle payments for orders.
 - Implement the payment process using the service's API (mocked for the assignment).
 - Ensure the system can process payments and update the order status based on the payment response.
- Database Implementation:**
 - Datasource should be compatible with Postgres database.
 - Connection information should be implemented in `.env` file.
 - If database migration implemented, it would be a positive point.
- CQRS Implementation:**
 - Commands/Queries should be implemented separately for each module that reads or write to database.
- Monorepo Structure:**
 - Use `nx` to create a monorepo structure.
 - Ensure the project is modular, with each domain and its corresponding modules separated into libraries/apps as appropriate.
 - Implement shared modules for common utilities or services.
- Validation & Error Handling:**
 - Implement validation on input data for the CRUD operations.
 - Ensure proper error handling and return meaningful error messages
- Versioning**
 - ensure everything is being tracked by a Github versioning tool and submit proper commit messages.

Deliverables:

- A fully functional Nest.js application following the DDD pattern within an `nx` monorepo.
- Application should be uploaded to a public git repository.
- Postman file should be also included to project so that we can simply send traffic to implemented endpoints.

The assignment should be delivered back within 7 days.