**Back-end Team Lead: Test Assignment**

Please submit the following tests: It’s fine as much as you can (even for issues that you couldn’t do).

**Section I :** Please summarise your responses to the following questions in a brief statement.

1. What are the main steps you consider in terms of problem-solving?
2. How do you work in an environment that has fast-paced processes?
3. What steps do you take for team building?

**Section II :**

**Coding Assessment: Advanced Order Management System**

**Problem Statement:**

You are tasked with creating a highly specialized order management system for a large e-commerce platform. This system will handle a high volume of orders, requiring advanced database design and optimization, secure authentication, real-time order tracking, and complex business logic.

**Features to Implement:**

**User Authentication and Authorization:**

* Implement role-based authentication with multiple user roles (e.g., customer, admin, support).
* Define specific permissions for each role (e.g., customers can place orders, admins can manage products, support can handle customer inquiries).

**Advanced Database Design:**

* Design a complex MongoDB schema to efficiently store and retrieve orders, products, customers, and related data.
* Optimise the database for high scalability and performance, considering factors like indexing and sharding.

**Real-Time Order Tracking:**

* Implement a real-time order tracking system using Socket.IO.
* Customers and support staff should be able to view the live status of orders (e.g., processing, shipped, delivered) in real-time.

**Inventory Management:**

* Implement a system to manage product inventory, including tracking available quantities and updating in real-time as orders are placed and fulfilled.

**Security Measures:**

* Implement advanced security measures, such as encryption of sensitive data, protection against common security threats (e.g., SQL injection, XSS), and secure API endpoints.

**Performance Optimization:**

* Optimise the server and database for high concurrency and low response times, considering techniques like caching, load balancing, and connection pooling.

**Evaluation Criteria:**

* Advanced database design and optimization techniques.
* Effective use of MongoDB features (e.g., indexing, sharding, transactions if applicable).
* Proper use of Express.js and Node.js for efficient routing and handling requests.
* Implementation of Socket.IO for real-time order tracking.
* Strong security measures and protection against common vulnerabilities.
* Performance optimization and scalability considerations.
* Clean and well-documented code.

**Submission Guidelines:**

* Create a GitHub repository containing your MEAN stack project.
* Include a detailed README.md file with instructions on how to set up and run the application.
* Provide documentation on database design and optimization strategies.

***Please feel free to take references and inspiration from other sites. \*\*\****

**\*\* Please describe your tech-stack for this system with a clear description. \*\***

\*\*\*\* Note: Don't clone project code from related projects on **Github** and other **resources**. Implement code development if possible. We will appreciate your effort. Thanks.

**\* There is no project as far as possible. \***

\*\* Please send me back the Github link until 30 April 2024.

Deadline – 30. April.2024 (Tuesday – 5:00 PM Myanmar Time)