**Simulacro de prueba de desempeño:**

**Team Lead:** Stiven Loaiza

**Historia de Usuario:**

Empresa: Librería Digital Siglo Van Rossum

**Contexto:**

La empresa se dedica a la venta y gestión de libros electrónicos. Necesitan una API para gestionar su catálogo de libros, autores y ventas.

**Tarea Asignada:**

Como desarrollador junior de una plataforma de gestión de una librería digital, quiero implementar las funcionalidades para crear, actualizar, eliminar y consultar libros, autores y ventas, para poder administrar eficientemente el catálogo y las transacciones. La API debe permitir registrar, consultar y actualizar un autor, crear un libro, actualizarlo, eliminarlo, agregar autores a ese libro y consultar el libro con sus autores. Además, debe permitir la creación y gestión de ventas de libros, incluyendo el registro de transacciones con el cliente, libro y fecha de venta. Debo poder consultar las ventas por libro con un filtro de fechas (>=), ordenamiento por fecha y paginación.

**Tiempo:** 8 horas

**Criterios de Aceptación:**

* **Persistencia de Datos:** Utilizar TypeORM con PostgreSQL o MySQL para la persistencia de datos. Modelar y crear las tablas requeridas para soportar los libros, autores y ventas. Plus si se implementan scripts de poblado de datos (seeds, helpers o migrations). Para los casos de eliminación utilizar softdelete.
* **Manejo de Rutas (Routing):** Implementar rutas según RESTful API para las operaciones CRUD de libros, autores y ventas. Asegurar que las rutas sean intuitivas y sigan las convenciones REST.
* **Data y DTO:** Manejar correctamente los params, query params, body y headers. Utilizar DTOs para la validación de datos en las rutas de creación y actualización.
* **Códigos de Respuesta:** Devolver los códigos de respuesta adecuados para cada operación (201, 200, 400, 401, 404, 500, etc.).
* **Documentación:** Incluir un archivo README que explique cómo configurar y ejecutar el proyecto. Utilizar Swagger para documentar la API generada.
* **Gitflow:** Utilizar Gitflow para la gestión de ramas y versiones del proyecto. Asegurarse de que los commits sean descriptivos, en ingles y las ramas sigan la convención de Gitflow.
* **Paginación, Filtros y Ordenamiento:** Implementar paginación en las consultas de libros, autores y ventas. Incluir al menos un filtro básico en la consulta de ventas por libro (por ejemplo filtro de búsqueda por autor). Plus si se aplican filtros en otras consultas o filtros más avanzados. Incluir al menos un ordenamiento en la consulta de ventas por fecha. Plus si se aplica ordenamiento en otras consultas.
* **Validación de Peticiones:** Implementar una validación que si es entre las 6pm y las 6am entonces rechazar la petición de venta de libro con un no autorizado debido a que los pedido no se pueden procesar en ese horario.
* **Uso de Herramientas del Framework:** Implementar guards para asegurar las rutas que quizás requieran protección. Implementar validaciones y/o pipes para asegurar la integridad de los datos y demás herramientas que ofrece NestJS.

**Entregables:**

Código fuente del proyecto en un repositorio GitHub. Adjuntar url del repositorio y el código en ZIP y enviar al email stiven.loaiza@riwi.io. (solo se revisan ramas con commits que máximo tengan la fecha y hora de finalización de la prueba).

**Notas Adicionales:**

Se valorará la claridad y organización del código, así como la correcta implementación de las herramientas y prácticas solicitadas.