**Evaluación tecnica**

**Objetivo:**

Desarrollar un microservicio en Java para el manejo de usuarios. El microservicio debe ser capaz de crear, leer, actualizar y eliminar registros de una entidad de "Usuario". Esta prueba se puede completar en menos de dos horas.

**Instrucciones:**

El microservicio debe tener una interfaz REST para el CRUD de usuarios.

La aplicación debe utilizar una base de datos embebida como H2 o similar.

Debe utilizar Spring Boot para el desarrollo del microservicio.

No se permite el uso de bibliotecas o frameworks adicionales.

La entidad de "Usuario" debe tener los siguientes campos: id, nombre, apellido, correo electrónico y fecha de nacimiento.

No es necesario desplegar la aplicación en OpenShift, sólo se requiere que se implemente localmente.

Es importante que el código esté bien estructurado, documentado y siga las convenciones de codificación de Java.

Se debe agregar paginación y ordenamiento a la interfaz REST para el listado de usuarios.

Además, debe incluir cobertura de pruebas unitarias y de integración en el proyecto utilizando JUnit y Mockito.

Finalmente, se deben proporcionar instrucciones claras sobre cómo compilar y ejecutar la aplicación, y cómo acceder a la interfaz REST

**Preguntas:**

¿Qué estrategia utilizaría para manejar errores en la capa de controladores REST?

Se debería crear un ControllerHandler y un ControllerAdvice para el cachado de excepciones, con el fin de presentar una respuesta completa al usuario.

¿Cómo garantizaría la seguridad de la aplicación y sus datos?

Cifrado de datos, con el fin de garantizar que solo usuarios autorizados puedan acceder a la información.

¿Qué es un endpoint de salud y cómo lo implementaría en este microservicio?

Es un endpoint de estatus, el cual permite saber si el microservicio se encuentra levantado.

**Entregables:**

El código fuente de la aplicación.

Instrucciones claras sobre cómo compilar y ejecutar la aplicación.

Respuestas a las preguntas planteadas.

Requerimientos para ejecutar el microservicio:

tener JDK 16 instalado

tener maven instalado

conexión a internet

Abrir el proyecto en springtools, generar el jar (clic derecho en el proyecto y Maven install).

Clic derecho en el proyecto y dar clic en Run As Spring Boot App.

Checar colección en postman (raíz del proyecto).