

Prueba técnica

Lenguaje: C#, JavaScript y/o Bootstrap y/o JQuery

Base de datos: SQL SERVER o Cualquier otra.

Back-End: .NET framework **.NET 8**

Front-End: MVC **react** o .NETCORE MVC

Proyecto 1:

Se requiere un API REST en las opciones antes mencionadas en sección Back-End

- Registrar los datos de un libro: titulo, año, genero, número de páginas, autor (Rut)
- Registrar los datos de los autores: Rut, Nombre completo, fecha de nacimiento, ciudad de procedencia, correo electrónico.

Reglas de negocio:

- Se debe garantizar la integridad de la información
- Se debe controlar el número de libros permitidos por autor (máximo 10)
- Si al intentar registrar un libro se supera el máximo permitido, debe generarse una excepción y responder con el mensaje “No es posible registrar el libro, se alcanzó el máximo permitido”
- Si al intentar registrar un libro y no existe el autor, responder el mensaje “El autor no está registrado”

Aspectos técnicos:

- Tener en cuenta una estructura de paquetes:
 - Entidades
 - DTO
 - Interfaces
 - Servicios (implementación)
 - Controladores
- Inyección de dependencias, uso de modelos, código legible son aspectos que serán tomados en cuenta.
- Todas las buenas prácticas de desarrollo serán tomadas en cuenta. (SOLID)

Proyecto 2:

Se requiere crear un proyecto aparte utilizando tecnologías señaladas en sección Front-End con el fin de consumir la API del proyecto 1 para buscar y registrar los libros y autores en el sistema

- El sistema debe permitir buscar por palabras claves, rut autor, autor, titulo libro, año libro.
- El sistema debe permitir crear un autor en el caso de no existir (identificador rut autor)
- El sistema debe mostrar una tabla resultado con las coincidencias encontradas, de forma paginada
- El sistema debe tener los siguientes campos requeridos y validados en el front-end
 - Autor
 - Rut
 - Nombre Completo
 - Fecha nacimiento
 - Correo electrónico
 - Libro
 - Titulo
 - Año
 - Genero
 - Número de páginas
 - Autor

Aspectos técnicos:

- Usar IDE Visual Studio 2022 o Visual Code.
- Usar librerías Bootstrap, JQUERY, JAVASCRIPT, etc. (opcional)
- Usar HTML en la confección de las vistas.
- Una buena organización y estructura del proyecto serán tomados en cuenta.
- Una buena maquetación de templates serán tomados en cuenta.

Entregables:

Como entregable final, se espera un repositorio público (Google drive, github, gitlab, etc.) que contenga:

- Código fuente de la(s) solución(es).
- Imágenes de evidencia con ejecución de pruebas funcionales.
- Archivos de script de base de datos y estructura de datos utilizados para la prueba.

Nota: el enlace final debe ser enviado por correo electrónico a gmedina@grupoeuro.cl