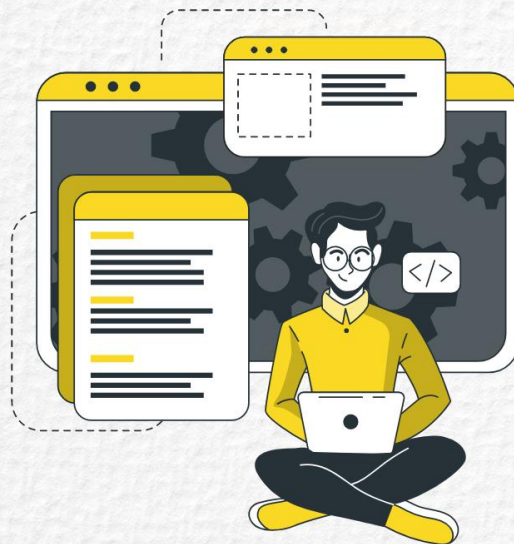


¡Tu siguiente paso está en Inlaze!

Contexto:

Inlaze es una compañía de **marketing de afiliados** que se especializa en **apuestas deportivas**. Actualmente, Inlaze se está posicionando como la **web services de apuestas deportivas más grande de Latinoamérica**.

Prueba técnica DESARROLLADORES



Si tienes alguna duda escríbela al mismo correo de **envío de la prueba**.
Se anexará el manual de marca.

Inlaze



Prueba técnica


Esta prueba técnica está diseñada en 2 partes, si usted se está postulando como desarrollador Frontend debe realizar la parte número 1, si es para desarrollador Backend debe realizar la parte número 2, si desea postularse en la posición FullStack debe realizar ambas partes.

La prueba tiene una duración 36 horas a partir del momento de su recepción.

Tecnologías a utilizar:

- Frontend: Angular
- Backend: Nest.Js
- ORM: TypeORM
- Database: Postgres
- Container: Docker

La prueba debe ser entregada en repositorios públicos de Github, incluir en el repositorio el paquete completo con assets usados, el script de base de datos, la colección de postman para probar los endpoints. Puntos importantes a evaluar:

- Clean code.
 - Best practices of coding.
 - Exception handlers.
 - Components.
 - Validations.
 - Data Structure.
 - Detailing.
- 



Prueba técnica

Parte 1: Frontend


Va a construir un portal web basado en el diseño entregado adjunto al correo, el cual hace referencia a un catálogo de cócteles, si su enfoque es solo para el Frontend utilice la siguiente API para consumir los endpoints necesarios y poder mostrar la información. Condiciones generales:

- Mostrar todo el catálogo de cócteles.
- Al ingresar a un detalle de un cóctel específico, mostrar los ingredientes, la descripción y la imagen, debe permitir seleccionar un ingrediente para actualizar la vista y mostrar los cócteles basado en ese ingrediente.
- Mostrar los cócteles según la letra inicial de su nombre.

Parte 2: Backend

Va a construir un backend encargado de administrar la siguiente información:

● Tabla Usuarios:

- id
 - full_name
 - email
 - password
 - phone
 - role
 - is_deleted
 - created_at
 - updated_at
- 





Prueba técnica

- Tabla Roles:
 - id name
 - is_deleted
 - created_at
 - updated_at Estas

2 tablas deben ser un microservicio en Nest.Js dicho microservicio va a tener la posibilidad de:

- Registrar un usuario.
- Autenticar un usuario (JWT).
- Editar un usuario.
- Eliminar un usuario (Borrado lógico).
- CRUD de los roles que se le asignan a los usuarios, en este caso es un rol por usuario.

El otro microservicio va a ser:

- Tabla Cócteles:
 - id
 - name
 - instructions
 - additional_notes
 - is_deleted O created_at
 - updated_at
 - Tabla Ingredientes:
 - id
 - name
 - is_deleted
 - created_at
 - updated_at
- 
- 



Prueba técnica

Estas tablas deben cumplir con las siguientes condiciones:

- Crear un nuevo cóctel seleccionando uno o varios ingredientes.
- Editar un cóctel existente.
- Eliminar un cóctel (Borrado lógico).
- Listar los cócteles por nombre, por ingrediente, por cantidad de ingredientes.

Cada una de estas peticiones solo se puede procesar si se tiene un token de autorización de un usuario que ha sido previamente registrado y autenticado en el sistema.

Se debe validar la data ingresada preferiblemente utilizando la librería Joi.

Ambos microservicios deben tener un Dockerfile que permita compilarlos y ejecutarlos en una imagen de Docker lo más liviano posible.

Entregables según aplique su caso:

- Repositorio con el Frontend.
- Repositorio con el Backend.
- Script de la base de datos.

Si tiene alguna duda, contacte a la persona de recursos humanos en el correo **hr@inlaze.com** o al número de teléfono **3222306047**.

¡Éxitos!

