



## CHALLENGE FRONT END SSR

### INTRODUCCIÓN

Nos complace conocerte y queremos saber más sobre tus habilidades técnicas a través de un desafío de desarrollo en diversos escenarios. El objetivo de cada escenario es que puedas demostrar cómo los resolverías, utilizando las herramientas que consideres necesarias.

Si algún tópico o escenario está fuera de tu conocimiento, no te preocupes. En ELDAR valoramos el trabajo en equipo, el acompañamiento y el mentoring de las personas, y es por eso que estamos interesados en conocerte.

Creemos en tu potencial y sabemos que darás lo mejor de ti.

¡Te deseamos mucho éxito!

## Creación de una Aplicación Web/Móvil con Login y Roles

Desarrollar una aplicación web o móvil con autenticación avanzada, manejo de roles, y una arquitectura optimizada para proyectos escalables. El rol "user" solo tendrá acceso de lectura a los datos obtenidos de una API, mientras que el rol "admin" podrá crear y editar datos. **Dependiendo de la entrevista, se te pedirá desarrollar una aplicación web o móvil con las siguientes tecnologías:**

- **Aplicaciones web: React o Angular.**
- **Aplicaciones móviles: React Native o Ionic.**

### 1

## Autenticación y Roles

- Implementa un sistema de autenticación básico para validar credenciales.
- Define dos roles de usuario: "admin" y "user".
- Valida formularios asegurándote de que los campos no estén vacíos y que las contraseñas tengan una longitud mínima.
- Utiliza un contexto o estado global para manejar roles y permisos.
- Según el rol, muestra u oculta elementos de la interfaz:
- El rol "user" podrá solo visualizar datos obtenidos de una API.
- El rol "admin" podrá crear y editar datos.
- Protege las rutas que requieren autenticación y roles específicos.

### 2

## Diseño de interfaz de usuario

- Diseña una interfaz atractiva y accesible.
- Crea una página de inicio de sesión donde los usuarios ingresen sus credenciales.
- Diseña una página de visualización de datos para el rol "user".
- Para el rol "admin", incluye opciones para crear y editar datos.

### 3

## Estilos y Maquetación

- Implementa el diseño utilizando CSS o alguna librería de estilos como Tailwind, Ionic, PrimeNg o Material UI. (A libre elección en caso de no ser especificado en la entrevista)
- Asegúrate de que la aplicación sea responsiva y funcione bien en diferentes tamaños de pantalla.
- Organiza los estilos en una estructura de carpetas reutilizable y escalable.
- Evalúa la calidad de tus estilos para asegurarte de que sigues buenas prácticas de desarrollo.

4

## Comunicación con la API

- Realiza las solicitudes HTTP necesarias para obtener y manipular datos utilizando herramientas como Axios (o HttpClient en Angular).
- La API sugerida es: <https://jsonplaceholder.typicode.com/guide/>

5

## Funcionalidades adicionales optativas

- Si lo deseas, puedes agregar funcionalidades adicionales como paginación, filtrado o búsqueda para mejorar la experiencia del usuario.
- Añade notificaciones para operaciones exitosas o fallidas.
- Implementa autenticación con tokens JWT simulados para hacer el desafío más realista.

6

## Requisitos técnicos

- Utiliza React, React Native, Angular o Ionic, según lo pautado en la entrevista.
- Estructura el proyecto con una arquitectura de carpetas bien organizada.
- Implementa TypeScript.
- Entrega tu trabajo en un repositorio (no es necesario hacer deploy).
- Incluye un README detallado que explique la estructura del proyecto y cómo instalar, ejecutar y probar la aplicación.

7

## Requisitos específicos para Aplicación Móvil

- Una vez que el usuario se haya logueado, deberá tener la opción de tomar una foto con la cámara del celular.
- La foto deberá ser guardada y mostrada como un avatar en la navegación de la aplicación o donde lo consideres.
- Para aplicaciones móviles, es necesario entregar el archivo APK para poder probar la aplicación.

Quedamos a tu disposición para cualquier consulta

¡Gracias por tu trabajo!