

Ejercicio React

Una plataforma de pagos que actualmente da de alta a sus usuarios en forma manual, quiere desarrollar una app web para automatizar su proceso de registro. Para brindar una mejor experiencia al usuario, se decide dividir el registro en 3 pasos de la siguiente manera:

Paso 1: Se pide el nombre completo del usuario y su número de CUIL. El nombre debe ser un campo de tipo texto y debemos validar que no ingrese una sola palabra. El número de CUIL debe tener 11 dígitos.

Paso 2: Se pide la calle, número, provincia y localidad del usuario. Las listas de provincias y localidades deben cargarse dinámicamente mediante requests a los siguientes endpoints:

- Provincias: <https://geopagos-challenge.s3.amazonaws.com/provinces.json>
- Localidades: <https://geopagos-challenge.s3.amazonaws.com/provinces/{ID}.json> (ID: id de provincia)

Paso 3: Se pide el email y contraseña que va a usar el usuario para ingresar a la plataforma. Debe ser un email válido y la contraseña debe cumplir con los requisitos de seguridad de la empresa (alfanumérica de al menos 8 caracteres y debe incluir al menos un número). La contraseña no debe exponerse mientras se escribe, salvo en el caso en que se marque la opción de mostrar contraseña (ver diseño).

En caso de error en algún campo, el usuario no podrá avanzar al siguiente paso y debe mostrarse un mensaje de error junto al mismo, el cual debe desaparecer una vez corregido el error.

Finalmente si todos los datos son correctos, deben enviarse a una API REST y luego de obtener una respuesta satisfactoria de parte de la misma, debe mostrarse un mensaje de registro exitoso. (En este paso se sugiere usar el sitio mocky.io para simular el request POST a la API).

Requisitos del ejercicio:

1. Desarrollar la app web de registro solicitada utilizando React y Sass.
2. Respetar el diseño UI enviado en el link de InVision junto al ejercicio.
3. Fundamentar el uso de toda librería de terceros ajena a React.
4. La app debe ser responsive.

Se evaluarán las buenas prácticas de desarrollo utilizadas en la implementación del ejercicio, el correcto manejo de React, y la estrategia que se utilice en la organización de los componentes y estructura del proyecto para que sea mantenible y reutilizable en futuros desarrollos.

¿Cómo sumar puntos extra?

5. Implementar una máscara para el campo CUIL del paso 1, de manera que los 11 dígitos se separen por guiones en grupos de 2, 8 y 1 dígitos (Por ejemplo: 30-12345678-0). A la API debe enviarse el número sin guiones.
6. Persistir el estado del registro en el navegador el usuario, para que pueda continuar desde donde estaba luego de refrescar la página.

Diseño: <https://invis.io/GROQD9RWCDU>