

# Entrevista Técnica

## Engineering Manager

### Parte 2

## Desarrollo de software

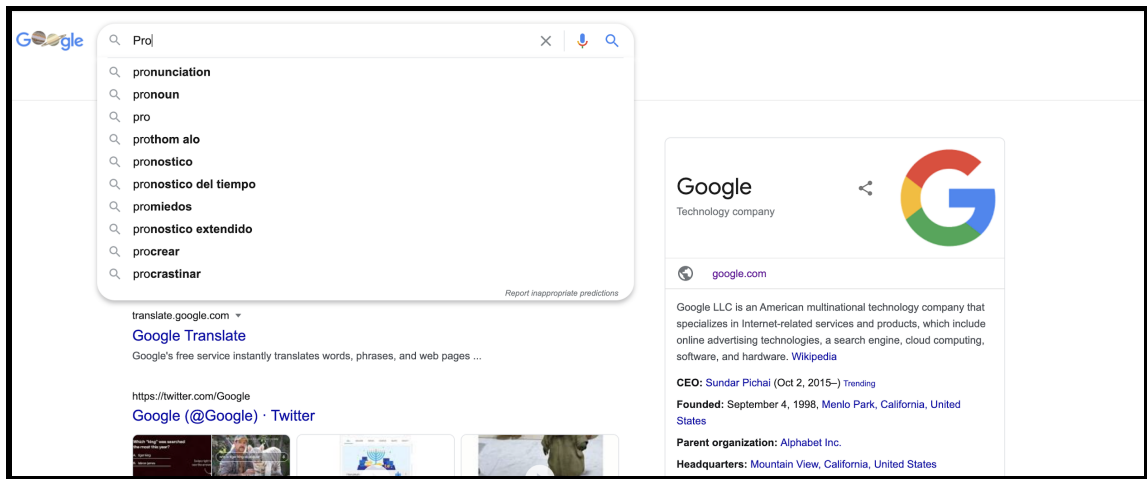
La finalidad del ejercicio es desarrollar una vista de registro de usuario haciendo foco en la calidad del código a generar. Dicha vista no tiene un diseño definido, por lo cual esperamos que se presente un diseño a criterio propio. Si bien no esperamos un diseño destacado, es deseable que la pantalla sea usable e intuitiva.

### Requerimientos y Restricciones

- No utilizar Bootstrap ni ninguna librería con clases de CSS predefinidas.
- Utilizar flexbox y Grid como principales herramientas para definir el layout.
- Las vistas deben ser responsive en mobile y tablet.
- Configurar un linter.
- Los botones deben tener algún efecto en hover.
- Utilizar al menos una animación además del efecto de hover en botones.

### Criterios de aceptación

- El formulario debe enviarse al endpoint "Sign Up" de [ésta API](#)
- Se deberán enviar al endpoint de registro los siguientes campos: nombre, apellido, país, email, teléfono y contraseña.
- El nombre y apellido deben ser 2 campos distintos. Cada uno con un máximo de 30 caracteres.
- Los países deben ser un listado obtenido de esta [API](#), el selector debe poder contar con un buscador teniendo un autocompletado (ver imagen inferior, emular el funcionamiento de Google).
  - Todos los países deben tener al costado la imagen de su bandera.
  - A la API en el campo "province" hay que enviarle la capital del país.



- El email debe tener un formato válido.
- El teléfono debe aceptar solo valores numéricos y un máximo de 10 caracteres.
- La contraseña debe ser alfanumérica con un mínimo de 6 caracteres.
- Debe existir un campo para repetir la contraseña, el cual no será enviado a la API, pero deberá validarse que sea igual a la contraseña.
- Debe existir un checkbox de “Acepto Términos y Condiciones” con un link a una página de términos y condiciones que no hace falta que sea creada.
- El usuario registrado debe ser guardado en LocalStorage cuando la API confirma que el registro fue satisfactorio.
- Todos los campos son requeridos. En caso de que alguno no cumpla con las validaciones el botón para submitear quedará deshabilitado y se le deberá dar feedback al usuario de qué validación no está cumpliendo.
- No utilizar validaciones por defecto de HTML.
- Usar un state management a elección propia (Redux/Context/Recoil/etc) para almacenar el usuario de forma global.
- Implementar testing con Jest y react testing library.
- Utilizar react hooks.
- Utilizar i18n para todos los textos del registro y agregar un botón para cambiar de idioma (español/inglés).
- Hacer deploy la aplicación a un ambiente símil productivo.

❗ **IMPORTANTE:** Se recomienda realizar consultas respecto al enunciado y a la descripción del proyecto vía mail a **gabriel.zanzotti@wolox.com.ar** para más información.