

# Desafío técnico FrontEnd React - Flow

## Consigna

El siguiente desafío plantea el desarrollo de una aplicación de **consulta de clima** en la que el usuario pueda visualizar el pronóstico actual y el de los próximos 5 días de la ciudad en la que se encuentra, y a su vez un selector que permita cambiar entre 5 ciudades preseleccionadas

## Requisitos

- El desafío debe ser resuelto en React.js.
- Se recomienda utilizar el servicio de API de [Open Weather Map](#), pero se puede utilizar cualquier otro de preferencia.
- Además del desarrollo específico de las funcionalidades, se requiere identificar y generar los casos de test que se consideren necesarios (Opcional).
- La entrega del código debe ser en un repositorio público de GitHub.
- Se evaluará la progresión del desarrollo en el historial de commits del repositorio.
- No se evaluarán skills de diseño pero sí la experiencia del usuario.
- El tiempo demorado en entregar el challenge **NO** será una variable a evaluar.

## Criterios mínimos para considerar válida la entrega

- Debe existir un SELECTOR con cinco ciudades preestablecidas arbitrariamente por el candidato.
- El desafío debe funcionar en ambiente local.
- Si el desafío cuenta con test, no deben existir test fallidos.
- La resolución base para la revisión es 1280\*768, el funcionamiento en otras resoluciones se consideran puntos extra.
- Es mandatoria la existencia de un archivo "Read me" con las instrucciones para levantar el proyecto e información relevante de las decisiones tomadas a la hora del diseño y desarrollo del desafío.
- Una vez entregado, el challenge será desestimado en caso de seguir agregando código en la rama *main/master* del repositorio. Si se busca hacer mejoras o corregir deuda técnica, deben estar en una branch separada y documentada en el archivo *readme*.

## Puntos a evaluar

- Arquitectura de código
- Buenas prácticas (React, Javascript y estilos)
- Fundamentos de programación funcional
- Test cases que se asemejan al comportamiento del usuario
- Separación de responsabilidades
- Cualquier definición o hipótesis a considerar DEBE estar aclarada en el Readme

## Deseables

- Typescript
- Testing (Preferentemente Jest o Cypress)
- Geolocalización
- Accesibilidad
- Etiquetas semanticas
- Deploy en plataforma web (vercel, netlify, GH Pages)
- Configuración de Linter