

Desafío técnico Frontend

Realizar un sitio web con React (se valora usar Next.js 13 como Framework) donde el 'Home' de la aplicación es un calendario con vista mensual de todo el mes de febrero de 2023. Para armar la grilla del calendario no se puede usar ninguna librería como lo es react-calendar, sino que se debe construir usando CSS. Se puede usar CSS Grid. Para obtener los días que contiene febrero de 2023 se puede usar librerías, como moment u otras. Es un plus si el calendario contiene botones para navegar por los meses anteriores.

La aplicación debe consumir la API de la NASA. Leer la documentación para obtener los endpoints que se necesiten usar.

Cada día del año la NASA sube una fotografía. Por lo tanto, por cada casillero del calendario se debe mostrar la miniatura de la fotografía que corresponde a ese día.

Cuando el usuario hace clic sobre un casillero del calendario, la aplicación lo dirige a otra página o se abre un modal (queda a elección del programador) donde se muestra la fotografía en alta resolución y el detalle de la misma.

Toda la aplicación debe seguir los lineamientos estéticos de la NASA.

Documentación de la API de la NASA

<https://api.nasa.gov/index.html>

Lineamientos estéticos de la NASA

https://www.nasa.gov/sites/default/files/atoms/files/nasa_graphics_manual_nhb_1430-2_jan_1976.pdf

Resumen de la consigna

El usuario debe poder:

- Ver la vista mensual de un calendario (no usar librerías tipo react-calendar, etc. Se puede usar CSS Grid).
- Cada día del calendario está representado por un cuadrado que tiene una imagen de fondo.
- La imagen de fondo de cada día del calendario es la miniatura de la imagen del día de la NASA.
- Al cliquear sobre una miniatura se abre una página o modal (a elección del programador) con la fotografía en alta resolución y el detalle de la misma.

Especificidades técnicas

- Realizarlo con React. Se valora usar el framework Next.js 13.
- El código debe estar completamente escrito en TypeScript y tipado.
- Diseño responsive.
- Utilizar la guía de marca de la NASA.

Plus

- **Lighthouse** u otra herramienta para pasar las siguientes métricas:
 - Performance 90%
 - Accesibilidad 90%
 - Best Practices 90%
- **Animaciones:** Al abrir el detalle de la miniatura se despliega con alguna animación.
- Configurar **linter** que controle estandarización del código (ej: tabs de 4 espacios, single quotes o double quotes, uso o no uso de punto y coma al final de cada línea, etc.).
- **Unitary testing** (Jest, Cypress, etc.).
- **Component testing** (react-testing-library, Cypress, etc.).

Se evaluará

- Escritura de código limpio y buenas prácticas.
- Evaluar el diseño visual para asegurar que se siga la estética de la marca.
- Revisar la navegación para comprobar la usabilidad y accesibilidad.
- Comprobar la compatibilidad con diferentes navegadores y dispositivos.
- Verificar la responsividad para asegurar que la interfaz se adapte a diferentes tamaños de pantalla.
- Arquitectura.
- Patrones de diseño.
- Prueba de rendimiento para medir el tiempo de carga de cada página web.