

TECHNICAL TEST FE

V2.0.0

June, 2021

PRERREQUISITOS

- Tener Git instalado
- Tener Node js Instalado (v 12.18.1 o superior)
- Clonar el repositorio con la url que has recibido vía email
- Desde la consola, Ve a la carpeta raíz y ejecuta el comando: `npm install`
- Antes de ejecutar el comando test debes ir al archivo `src/config/index.js` y modificar la URL del servidor, por la url y puerto de tu proyecto
- Desde la consola, ejecuta el comando: `npm run test, and happy code! :)`

HTML FEATURE

A partir de este [mockup](#) diseñado por nuestro equipo, requerimos implementar un sitio web sencillo con algunos de estos widgets en su interfaz. Ten en cuenta que las letras en color Verde, y las secciones “Elements” y “Primary colors” son instructivas y no hacen parte del diseño.

- Para lo cual vas a requerir consumir los siguientes Endpoints:
 1. Sedes: [link](#), retorna un listado de nuestras sedes.
 2. Pronóstico extendido: [link](#), retorna el pronóstico extendido del clima de nuestra sede principal.
 3. Ciudades del mundo: [link](#), retorna una lista de ciudades del mundo con información básica.
- - Recomendamos la fuente "weathericons" para los iconos del estado del tiempo. Weather Icons.
 - Recomendamos "Font Awesome" para el uso de otros iconos - Fontawesome.
 - Recomendamos la tipografía "Lato". Google fonts.
 - Recomendamos vía CDN las fuentes.

Se requiere el widget #3 y debe mostrar el pronóstico de nuestras otras dos sedes. Se requiere cumplir con lo siguiente:

1. El widget debe contener un **id="widget-favorite"**
2. Los datos de cada sede deben estar contenidos en un elemento html con la clase **"widget-card"**
3. El elemento html que contenga el nombre de la ciudad de la sede debe tener la clase **"widget-headquarter-name"**
4. El elemento html que contenga el weather icon de cada sede debe tener la clase **"widget-icon"**.

Se requiere el widget #4 y debe mostrar cual es el mejor día para realizar nuestra actividad al aire libre, según el pronóstico extendido de nuestra sede principal. El mejor día es aquel cuya temperatura sea agradable (**entre 25°C y 28°C**) y cuyo porcentaje de humedad sea el más bajo. Se requiere cumplir con lo siguiente:

1. El widget debe contener un **id="widget-suggest"**
2. El elemento html que contenga el nombre de la ciudad de la sede debe tener la clase **"widget-headquarter"**
3. El elemento html que contenga el weather icon de cada sede debe tener la clase **"widget-icon"**
4. El elemento html que contenga el nombre de la ciudad de la sede debe tener la clase **"widget-day"**
El nombre del día debe ser calculado a partir de la propiedad **date** retornada en la respuesta del endpoint.

Se requiere el widget #5 y debe mostrar 3 ciudades del mundo recomendadas para visitar en la temporada, a partir de los datos obtenidos del endpoint **Ciudades del mundo**. Las ciudades del mundo recomendadas, son aquellas ciudades cuya temperatura está entre **24°C y 30°C**. Se requiere cumplir con lo siguiente:

1. El widget debe contener un **id="widget-cities"**
2. Los datos de cada ciudad deben estar contenidos en un elemento html con la clase **"widget-card"**
3. El elemento html que contenga el nombre de la ciudad debe tener la clase **"widget-city-name"**
4. El botón "Visit Place" de cada ciudad debe ser un elemento html **"a"** con el hipervínculo a otra pestaña.

Layout

- Hacer uso de la semántica y la especificidad para la maquetación.
- No está permitido el uso de librerías o frameworks como Bootstrap, Bulma entre otras para este fin.
- El responsive y la distribución de los elementos queda a criterio del desarrollador asegurando que la renderización sea legible.

- Browser modernos en últimas versiones

Interacción

- Usa Javascript
- Hacer uso de alguna tecnología JS (Angular, React, Vue.)
- Usa sintaxis ES6.
- Buenas prácticas (SOLID, DRY entre otras).

Bonus track

- Animaciones.
- Prettier o Linters.
- Preprocesadores.
- Unit Testing.

Entrega Final

- La estructura de tu proyecto debe tener un directorio **dist/** donde debes incluir el build de su proyecto.
- Debes subir el código fuente (incluyendo el build) al repositorio de Github indicado.
- Con cada push al repositorio, se disparará el despliegue automático del contenido del directorio **dist/** (build) a un servidor accesible con la URL indicada.