



El futuro digital  
es de todos

MinTIC

## Sprints: Desarrollo Aplicaciones Web.



Universidad  
Pontificia  
Bolivariana

Vigilada Mineducación

Mision  
TIC 2022



## Generalidades

- Recuerda reunirte con tu equipo de trabajo para determinar los acuerdos en la ejecución de las actividades.
- Implementen la primera versión de su proyecto con algunas funcionalidades básicas y definan los repositorios compartidos para la integración y despliegue continuo.
- Este sprint tiene como propósito realizar una implementación con una API de prueba.
- Hasta este momento debes tener presente las vistas de tu proyecto, la estructura de manejo de datos y la lógica de operación de las funcionalidades.

## Recomendaciones

1. Entrega únicamente los archivos solicitados dentro del espacio de la actividad.
2. Esta debe hacerse a través del autocalificador codegrade.
3. Verifica los nombres de los archivos de entrega y la extensión.
4. Ingresen al espacio en plataforma “Actividad: Sprint 3” y sigan las instrucciones.
5. Esta actividad tiene como propósito revisar el concepto de API y su uso desde una fuente externa.

## Introducción/Información:

En Colombia, el monitoreo y control de las emisiones de gases es regulado por la norma técnica NTC-4983 que en su numeral 5.2.1 establece los parámetros de medición

### 5.2.1 Parámetros de medición

Los gases de escape cuyas concentraciones se deben determinar y las unidades de medida en que se deben reportar son:

Tabla 1. Parámetros de medición

Parámetro	Símbolo	Intervalo mínimo de medición	Unidad
Monóxido de carbono	CO	0 a 10	% en volumen
Dióxido de carbono	CO <sub>2</sub>	0 a 20	% en volumen
Hidrocarburos (en términos de n-hexano)	HC	0 a 10 000	ppm (partes por millón)
Oxígeno	O <sub>2</sub>	0 a 22	% en volumen

### 5.2.3.4 Puntos de verificación del intervalo de medición.

El equipo debe seguir de manera automática, un procedimiento de verificación y ajuste con dos puntos de gas de referencia como se detalla en la Tabla 3.

Tabla 3. Puntos de verificación

Punto bajo	Punto Alto
300 ppm. de propano	1 200 ppm. de propano
1,0 % de monóxido de carbono (CO)	4,0 % de monóxido de carbono (CO)
6,0 % de dióxido de carbono (CO <sub>2</sub> )	12,0 % de dióxido de carbono (CO <sub>2</sub> )

Se puede emplear el aire ambiente para verificar el sensor de O<sub>2</sub>.

## Objetivos:

- Resolver las historias de usuario propuestas como requisitos del proyecto de aplicaciones web asignado durante el ciclo.
- Repasar el concepto de microservicios, APIs y su implementación dentro de proyectos.
- Implementar el primer prototipo del proyecto según las vistas, estructura de manejo de datos y funcionalidades básicas.

## Contexto:

Como desarrollador de aplicaciones web ha sido contratado para programar la nueva interfaz gráfica para la página web de la secretaria de movilidad, que será un apoyo a los puntos de medición de gases móviles en las vías de la ciudad.

## Historias de usuario:

Identificador Historia#:	HU-01	Título:	Encontrar el porcentaje de los parámetros del analizador de gases
Descripción	COMO:	usuario	
	QUIERO:	Visualizar el porcentaje de los parámetros con su respectiva resolución	
	PARA:	Revisar la calidad del ambiente	
Criterios de aceptación	I. DADO que un usuario se encuentra en el sistema CUANDO se va al dashboard ENTONCES el sistema muestra cada una de las fases del proceso de análisis de gases		
	1. *Una empresa de backend fue contratada para servir un API que retorne los límites de los indicadores de calidad del aire, siéntete libre de explorar su resultado en tu navegador de internet preferido o en cualquier cliente http. <a href="https://misiontic2022upb.vercel.app/api/emission-measurement/limits">https://misiontic2022upb.vercel.app/api/emission-measurement/limits</a>		
	1. Construir una función (async) que se llame calcularPocentajes que reciba 8 parámetros, limiteCO, limiteCO2, limiteHC, limiteO2, valorCO, valorCO2, valorHC, valorO2 y retorne un objeto con los porcentajes de tipo {porcentajeCO: x%,porcentajeCO2:y%, porcentajeHC:z%, porcentajeO2:p%} 2. Si el límite de algún indicador es 0 entonces el porcentaje será 0 ; 3. Llamar a la api <a href="https://misiontic2022upb.vercel.app/api/emission-measurement/limits">https://misiontic2022upb.vercel.app/api/emission-measurement/limits</a> , para recuperar los límites permitidos y así hacer su correspondiente cálculo. 4. Exporte la función empleando la siguiente sentencia de código: <ul style="list-style-type: none"><li>module.exports.calcularPocentajes= calcularPocentajes;</li></ul>		



Identificador Historia#:	HU-02	Título:	Niveles de parámetros NTC-4983
Descripción	COMO:	Usuario	
	QUIERO:	Evidenciar el valor cualitativo de un valor NTC	
	PARA:	Saber en qué rango se encuentra el parámetro y si supera la prueba de medición	
Criterios de aceptación	I. DADO que un usuario se encuentra en el sistema CUANDO está en el home ENTONCES el sistema le mostrará una etiqueta diciéndole el valor del parámetro.		
	II. Hay Variables en el ambiente de pruebas que se llama rangosCO, rangosCO2, rangosHC, rangosO2, siéntete libre de hacerle console.log(para ver que tiene), no es necesario definirlo. OJO sólo está en el ambiente de pruebas (podemos verlo en la sección “entrega”)		
	1. *Una empresa de backend fue contratada para servir un API que retorne los rangos cualitativos de los niveles de parámetros de la revisión técnica de gases , siéntete libre de explorar su resultado en tu navegador de internet preferido o en cualquier cliente http. <a href="https://misiontic2022upb.vercel.app/api/emission-measurement/ranges-parameters">https://misiontic2022upb.vercel.app/api/emission-measurement/ranges-parameters</a>		
	1. Crear una función (async) que se llame registrarCO, que reciba un parámetro valor, y retorne el valor cualitativo según el rango en el que esté.		
	2. Recorrer el arreglo rangosCO		
	3. Llamar a la API <a href="https://misiontic2022upb.vercel.app/api/emission-measurement/ranges-parameters">https://misiontic2022upb.vercel.app/api/emission-measurement/ranges-parameters</a> usando la función fetch y guardar su valor en una variable llamada rangoCO.		
	4. Encárgate de verificar el rango en el que se encuentra la variable valor,		
	5. Devolver la propiedad etiqueta (valor equitativo). DEL RANGO “Parámetro CO en rango estandar”, “Parametro CO fuera de rango”...		
	6. Si el valor es menor que 0 o mayor que 10 la función deberá retornar el string “fuera_de_rango”		
	7. Exporte la función empleando la siguiente sentencia de código: module.exports.registrarCO= registrarCO;		
	Lo mismo se hace para las demás parámetros registrarCO2, registrarHC, registrarO2, rangosCO2, rangosHC, rangosO2		

## Sprint backlog

1. Construya un tablero de apoyo para la asignación y estado de las actividades.
2. Buscar la fórmula de depreciación lineal, que está sugerida en los materiales de apoyo de este enunciado.
3. Revisar como llamar una API con fetch (usando async/await), trabajar con objetos, recorrer un Array, clonar objetos (spread operator o Object.assign) y adicionar elementos en un array guiándose de los materiales de apoyo
4. Preparar y codificar las funciones y variables.
5. Revisar muy bien los strings de retorno, nombre de funciones y variables, con un gran nivel de detalle.
6. Probar por aparte su código, para su comodidad (opcional).
7. Establezcan una revisión periódica de los avances.
8. Realicen el montaje del primer prototipo de manera local en sus máquinas.





El futuro digital  
es de todos

MinTIC



## Entrega:

Suba a la plataforma los archivos JS, que empleó para dar solución a cada una de las actividades de las historias de usuario planteadas en este sprint; el archivo debe de llamarse **src.js**. Adicionalmente, incluya el archivo en formato JSON disponible en la descripción del sprint (“range\_parameters.json” – “limits.json”) de lo contrario no se calificará. Para el fetch hacer uso de ASYNC/AWAIT

Para la evaluación de este sprint es **muy importante** seguir muy detalladamente las instrucciones, los valores de retorno, las variables, strings.

```
const calcularPocentajes = async (
  valorC0,
  valorC02,
  valorHC,
  valor02
) => {
  //make a fetch request to the api
  let response = await fetch("https://misiontic2022upb.vercel.app/api/emission-measurement/limits");
  let limits = await response.json();
  .....
  return {
    porcentajeC0,
    porcentajeC02,
    porcentajeHC,
    porcentaje02,
  };
};

const registrarC0 = async (value) => {
  let response = await fetch("https://misiontic2022upb.vercel.app/api/emission-measurement/ranges-parameters");
  let rangosC0 = await response.json();
  let candidate = rangosC0[0]
  .....
};

const registrarC02 = async (value) => {
  let response = await fetch("https://misiontic2022upb.vercel.app/api/emission-measurement/ranges-parameters");
  .....
};

const registrarHC = async (value) => {
  let response = await fetch("https://misiontic2022upb.vercel.app/api/emission-measurement/ranges-parameters");
  .....
};

const registrar02 = async (value) => {
  let response = await fetch("https://misiontic2022upb.vercel.app/api/emission-measurement/ranges-parameters");
  .....
};

module.exports.registrarC0 = registrarC0;
module.exports.registrarC02 = registrarC02;
module.exports.registrarHC = registrarHC;
module.exports.registrar02 = registrar02;
module.exports.calcularPocentajes = calcularPocentajes;
```



## Material de apoyo:

<https://www.innovacionesiac.com/NTC-4983.pdf>

<https://www.skypack.dev/view/react-circle-progress-bar>

[https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Loops\\_and\\_iteration](https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Loops_and_iteration)

[https://www.w3schools.com/jsref/jsref\\_foreach.asp](https://www.w3schools.com/jsref/jsref_foreach.asp)

<https://www.jose-aguilar.com/blog/5-formas-diferentes-para-recorrer-un-array-con-javascript/>

<https://dmitripavlutin.com/javascript-fetch-async-await/#2-fetching-json>

[https://www.w3schools.com/js/js\\_async.asp](https://www.w3schools.com/js/js_async.asp)

[https://www.w3schools.com/js/js\\_objects.asp](https://www.w3schools.com/js/js_objects.asp)