



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:

Los coronavirus (CoV) son virus que surgen periódicamente en diferentes áreas del mundo y que causan Infección Respiratoria Aguda (IRA), es decir gripe, que pueden llegar a ser leve, moderada o grave.

El nuevo Coronavirus (COVID-19) ha sido catalogado por la Organización Mundial de la Salud como una emergencia en salud pública de importancia internacional (ESPII). Se han identificado casos en todos los continentes y, el 6 de marzo se confirmó el primer caso en Colombia.

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:

Ante la pandemia del Covid-19, es prudente recurrir a la tecnología y a la innovación para salvar la mayor cantidad de vidas posibles. Un ejemplo de esto es la aplicación colombiana CoronApp, que con datos del Instituto Nacional de Salud (INS), es una de las fuentes de información y captura de datos más completa para la toma de decisiones por parte de las autoridades.

Historias de usuario:

Identificador Historia#:	HU-05	Título:	Registrar municipios en el API
Descripción	COMO:	Administrador	
	QUIERO:	Registrar la información de los municipios	
	PARA:	Tener un histórico de los datos del comportamiento del virus.	
Criterios de aceptación	I. DADO que un usuario se encuentra en el sistema CUANDO va a registrar la información de un municipio ENTONCES el sistema muestra un formulario para agregar los datos de un municipio		
	1. *Una empresa backend fue contratada para construir un API que retorne para hacer el CRUD de las operaciones Siéntete libre de explorar su resultado en algún cliente http (usando el método post). https://misiontic2022upb.vercel.app/api/epidemiologic-monitor/villages		
	1. Crear una función (async) registrarMunicipio que reciba 6 parámetros, municipio, departamento, numeroCasos, numeroMujeres,numeroHombres, numeroRecuperados. 2. Enviaremos un objeto por método POST a la API https://misiontic2022upb.vercel.app/api/epidemiologic-monitor/villages y su estructura será mas o menos así { municipio:"Algeciras", departamento:"Huila", numeroCasos:"2342",numeroMujeres:"342", numeroHombres: "2000", numeroRecuperados:"2124" } usando fetch ASYNC/AWAIT 3. Retornar el string resultado del item 2. 4. Exporte la función empleando la siguiente sentencia de código: • module.exports.registrarMunicipio= registrarMunicipio;		

Identificador Historia#:	6	Título:	Clasificador de noticias Covid
Descripción	COMO:	Usuario	
	QUIERO:	Ver noticias relacionadas con los temas de covid19	
	PARA:	Estar atento de novedades en el cambio del comportamiento viral.	
Criterios de aceptación	I. DADO que un usuario se encuentra en el sistema CUANDO está en la sección de noticias ENTONCES el sistema le mostrará sólo las noticias relacionadas con covid		

Continua en la siguiente página



Criterios de aceptación	<ol style="list-style-type: none">1. Se ha implementado un scrapper, que recoge noticias de montones de sitios, y la tarea es crear un clasificador básico de las noticias.2. Una empresa de backend fue contratada para servir un API y retorna el resultado del scrapper del anterior punto, siéntete libre de explorar su resultado en tu navegador de internet preferido o en cualquier cliente http. https://misiontic2022upb.vercel.app/api/epidemiologic-monitor/news <ol style="list-style-type: none">1. Crear una función (async) que se llame filtrarEInsertarNoticias, que no reciba parámetros y devuelva las noticias que tienen relación con la pandemia por Covid-19.2. Llamar a la api https://misiontic2022upb.vercel.app/api/epidemiologic-monitor/news haciendo uso de la función fetch con ASYNC/AWAIT.3. Recorrer el array de llamado de la API (item 2) para comenzar la clasicación de las noticias.4. Crear un array que se llame noticiasFiltradas que empiece vacío.5. Si el título o el contenido tienen las palabras “coronavirus” o “covid19”, se agregará un objeto de tipo {titulo:”título de noticia de covid19”, contenido: ”contenido de la noticia de coronavirus”, url:”https://path-to-noticia”} al array noticiasFiltradas.6. (opcional) como alternativa al la estructura condicional que puedes hacer en el paso 5 puedes usar la función filter contenida dentro de cualquier array. O usas sentencia “if” o la función filter7. Retorne el array noticiasFiltradas.8. Exporte la función empleando la siguiente sentencia de código:<ul style="list-style-type: none">• <code>module.exports.filtrarEInsertarNoticias= filtrarEInsertarNoticias;</code>
-------------------------	--

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 (“news.json”) de lo contrario no se calificará. Para el uso de fetch usar las funciones ASYNC/AWAIT

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

```
const registrarMunicipio = async (
  municipio,
  departamento,
  numeroCasos,
  numeroMujeres,
  numeroHombres,
  numeroRecuperados
) => {
  .....
  let response = await fetch(
    "https://misiontic2022upb.vercel.app/api/epidemiologic-monitor/villages",
    {
      method: "POST",
      headers: {
        Accept: "application/json",
        "Content-Type": "application/json",
      },
      body: .....,
    }
  );
  let result = await response.json();
  .....
};

const filtrarEInsertarNoticias = async () => {

  let response = await fetch(
    "https://misiontic2022upb.vercel.app/api/epidemiologic-monitor/news"
  );
  let noticiasAPI = await response.json();
  .....

};

module.exports.filtrarEInsertarNoticias = filtrarEInsertarNoticias;
module.exports.registrarMunicipio = registrarMunicipio;
```

Material de apoyo:

https://play.google.com/store/apps/details?id=ffz.it.airquality&hl=es_CO&gl=US

<https://www.ambientebogota.gov.co/calidad-del-aire>

<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_objects.asp

https://developer.mozilla.org/es/docs/Web/JavaScript/Reference/Global_Objects/Array/filter