



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 segunda versión de su proyecto con algunas funcionalidades básicas y las pruebas del valor agregado.
- Este sprint tiene como propósito continuar realizando pruebas sobre la implementación backend.
- Ya es momento que vayas pensando como será el despliegue de tu proyecto. Accede a tus servidores locales y los servidores en la instancia.

## 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 4” y sigan las instrucciones.

## Introducción/Información:

A través de la Resolución 1552 del 14 de mayo de 2013, el Ministerio de Salud y Protección Social, reglamentó que las Entidades Promotoras de Salud (EPS), de ambos regímenes, directamente o a través de la red de prestadores que definan, deberán tener siempre agendas abiertas para la asignación de citas de medicina especializada la totalidad de días hábiles del año. Dichas entidades en el momento en que reciban la solicitud, informarán al usuario la fecha para la cual se asigna la cita, sin que les sea permitido negarse a recibir la solicitud y a fijar la fecha de la consulta requerida.

Las medidas hacen parte de la reglamentación de los artículos 123 y 124 del [Decreto Ley 019 de 2012](#) conocido como Decreto Antitrámites.

## Objetivos:

- Resolver las historias de usuario propuestas como requisitos del proyecto de aplicaciones web asignado durante el ciclo.
- Repasar el concepto de stack MERN y manejo de tecnologías.
- Implementar el segundo prototipo del proyecto según las vistas, estructura de manejo de datos, funcionalidades básicas y funcionalidades de valor agregado

## Contexto:

Como desarrollador de aplicaciones web ha sido contratado para diseñar la nueva interfaz gráfica para la página web de la EPS COLSALUD, donde se podrán agendar las citas medicas por parte de sus afiliados

## Historias de usuario:

Identificador Historia#:	HU-07	Título:	API para retornar las citas médicas
Descripción	COMO:	Usuario	
	QUIERO:	Buscar citas disponibles, de acuerdo a mis preferencias	
	PARA:	Ordenar mejor mi tiempo.	
Criterios de aceptación	1. DADO que un usuario se encuentra en el sistema, CUANDO le da click en la sección Solicitar Cita médica ENTONCES el sistema le mostrará un formulario para poder llenar sus datos de preferencia.		
Precondiciones	1. Ahora tu eres parte de la empresa de backend contratada para el sistema de Calidad del aire, y te piden que desarrolles un API para retornar las citas médicas disponibles. 2. Hay un array que se llama appointments a tu disposición que contiene citas médicas en la base de datos, puedes usarla en el entorno de pruebas para ver que todo está bien. OJO Sólo está en el entorno de pruebas.		
Instrucciones	1. Haz la inicialización de express. Con la configuración inicial que está en la sección de entrega. 2. No olvides en express aceptar jsons como cuerpo de la petición. (ver sección entrega) 3. Crear en Express un manejador de rutas con que reciba la ruta relativa /api/medical-appointments/apointments con método GET. 4. El api tiene que retornar el array de los appointments, como json (ver materiales de apoyo). 5. Exporte la variable app: <ul style="list-style-type: none"><li>module.exports=app;</li></ul>		

## Historias de usuario:

Identificador Historia#:	HU-08	Título:	API para confirmar cita
Descripción	COMO:	Usuario	
	QUIERO:	Confirmar la cita escogida	
	PARA:	Poder guardarla y posteriormente hacer uso de ella	
Criterios de aceptación	1. DADO que un Usuario se encuentra en el sistema, CUANDO le da click en el botón confirmar cuenta. ENTONCES el sistema envía los datos a la API para que se agende y confirme la cita		

Precondiciones	<ol style="list-style-type: none"><li>1. Ahora tu eres parte de la empresa de backend contratada para el sistema de Calidad del aire, y te piden que desarrolles un API para la confirmación de citas.</li><li>2. Hay un array que se llama appointments a tu disposición que contiene citas médicas en la base de datos, puedes usarla en el entorno de pruebas para ver que todo está bien. OJO Sólo está en el entorno de pruebas.</li></ol>
Instrucciones	<ol style="list-style-type: none"><li>1. Haz la inicialización de express. Con la configuración inicial que está en la sección de entrega.</li><li>2. No olvides en express aceptar jsons como cuerpo de la petición. (ver sección entrega)</li><li>3. Crear en Express un manejador de rutas con que reciba la ruta relativa <code>/api/medical-appointments/confirm/:appointment_id</code> con método POST. Donde <code>appointment_id</code> es el id de la cita a confirmar.</li><li>4. Recuperar el <code>appointment_id</code> de <code>req.params</code>. (ver materiales de apoyo)</li><li>5. Para poder comparar el <code>appointment_id</code> con el id de cada una de las citas necesitamos hacer la conversión del <code>appointment_id</code> (que viene de la ruta como string) a Entero (ver materiales de apoyo).</li><li>6. Buscar la cita en el array cuyo <code>appointment.id</code> concuerde con el <code>appointment_id</code> del punto 6 y le ponemos la propiedad <code>appointment.status = "confirmed"</code>.</li><li>7. Sobreescribimos el array <code>appointments</code> que tenga la nueva cita confirmada. (Hint ver la sección entrega)</li><li>8. El API tiene que retornar el array <code>appointments</code>, como json (ver materiales de apoyo)</li></ol> <p>Exporte la variable app:</p> <ul style="list-style-type: none"><li>• <code>module.exports=app;</code></li></ul>



## Entrega:

Suba a la plataforma el 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 (“appointments.json”) de lo contrario no se calificará.

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

```
const express = require("express");
const app = express();
app.use(express.urlencoded({ extended: false }))
app.use(express.json());

app.get("su ruta para recuperar las citas va aquí", (req, res) => {
  //su código aquí
});

app.post("su ruta para confirmar va a qué", (req, res) => {
  //....
  //...
  appointments=appointments.map(appointment => {
    //lógica para confirmar la cita
  })
  //.....
});
module.exports = app;
```



## Material de apoyo:

<https://expressjs.com/es/starter/hello-world.html>

<https://www.geeksforgeeks.org/express-js-req-params-property/>

<https://www.geeksforgeeks.org/express-js-express-json-function/?ref=lbp>

<https://expressjs.com/es/starter/basic-routing.html>

<https://www.geeksforgeeks.org/express-js-res-json-function/>

[https://www.w3schools.com/js/js\\_number\\_methods.asp](https://www.w3schools.com/js/js_number_methods.asp)

[https://developer.mozilla.org/es/docs/Web/JavaScript/Reference/Global\\_Objects/Array/push](https://developer.mozilla.org/es/docs/Web/JavaScript/Reference/Global_Objects/Array/push)