



El futuro digital  
es de todos

MinTIC

## Ciclo 3:

## Sprints de evaluación.



Universidad  
Pontificia  
Bolivariana

Vigilada por el Ministerio de Educación

Mision  
TIC 2022



## IMPORTANTE:

La información contenida en esta presentación es de carácter pedagógico, el material presentado obedece a situaciones que simulan el desarrollo de un proyecto en un entorno organizacional a partir de una metodología ágil como lo es SCRUM.

En este sentido, no se exigirá dominar los artefactos, herramientas y cargos de la metodología, pero si establecer acuerdos para llevar a buen fin el proyecto.

¿Cómo afrontar los sprints durante este ciclo 3?



## Recuerda...

¿Cuál es el alcance del proyecto que se debe desarrollar durante todo el ciclo?

1. Un sitio web, para un sector económico de relevancia actual, en el que se aplique el conocimiento adquirido durante ciclo 3
2. El sitio debe estar soportado por una base de datos en la que se pueda gestionar la información asociada al proyecto.
3. El sitio debe cumplir con una interfaz gráfica para ser usada en diferentes navegadores.
4. El sitio debe ser desplegado en un servicio cloud para el acceso durante la feria de proyectos.

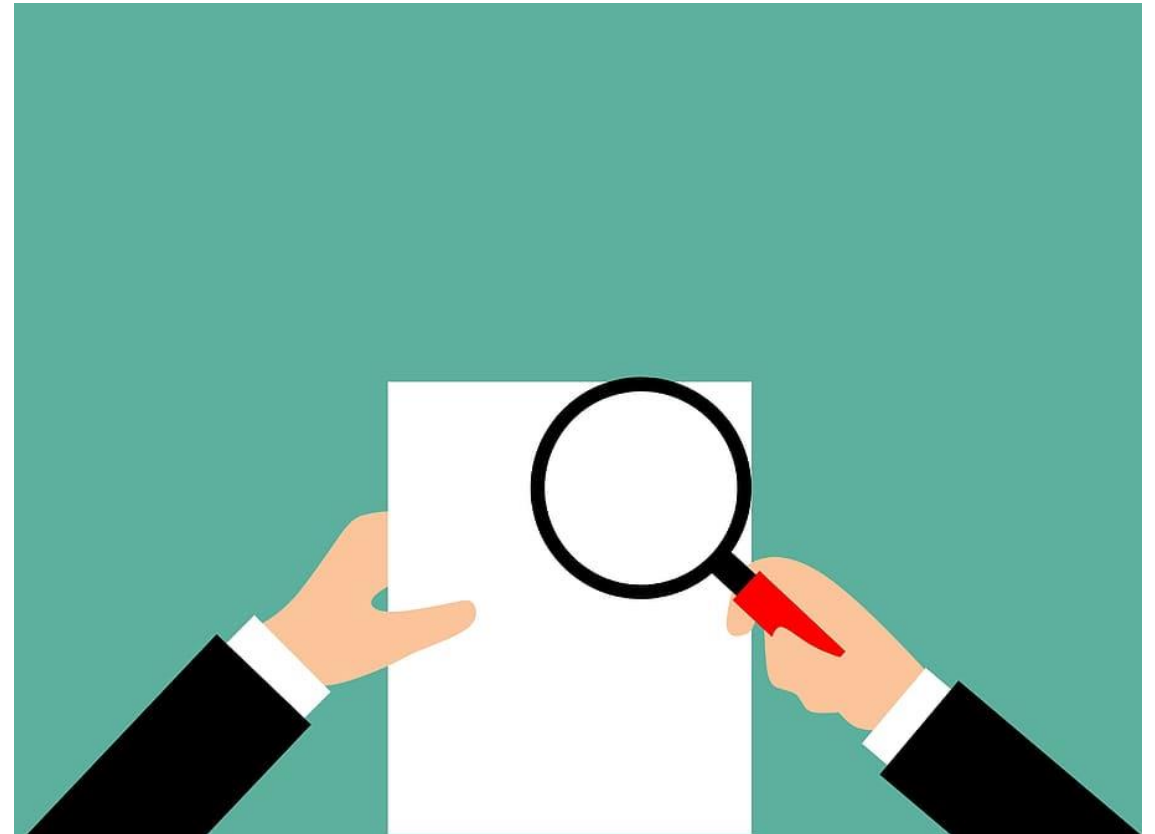
Los sprints están definidos bajo historias de usuario con requerimientos genéricos de sitios web. Su propósito es validar los conocimientos en los temas centrales del ciclo y se publicarán semanalmente.

Los sprints por si solos no completan el proyecto, por lo tanto, el trabajo de cada subgrupo será refinar al máximo su propuesta, durante las semanas de formación, para la postulación y participación en el evento de cierre de ciclo.

## Product Backlog:

### Historias de usuario:

1. Yo como administrador requiero poder guardar la información ingresada en el formulario de registro en una estructura de datos.
2. Yo como administrador requiero manipular la información guardada, con el fin de hacer un análisis de la misma.



## Sprint Backlog:

### Historia de usuario

Yo como administrador requiero poder guardar la información ingresada en el formulario de registro en una estructura de datos.

### Actividades

1. Crear un arreglo de objetos que contenga la información consignada en el formulario de registro

### Instrucciones:

- Declarar un arreglo global en donde se almacenen cada uno de los registros. Se almacenarán tantos registros como la cantidad de veces que se haya enviado el formulario.
  - NOTA IMPORTANTE: Debes exportar el arreglo global declarado con la sentencia
  - **module.exports.registros = registros;**
- Crear la función **agregarRegistro()**, la cual estará encargada de realizar las operaciones necesarias para tomar la información ingresada en el formulario de registro y guardarla en el arreglo.
- **Opcional:** Emplee el objeto **console** para imprimir el contenido que se está ingresando en el arreglo, tome de referencia la siguiente documentación: <https://developer.mozilla.org/es/docs/Web/API/Console>
- Exporte la función empleando la siguiente sentencia de código:
  - **module.exports.agregarRegistro = agregarRegistro;**

## Sprint Backlog:

### Historia de usuario

Yo como administrador requiero manipular la información guardada, con el fin de hacer un análisis de la misma.

### Actividades

1. Ordenar el contenido del arreglo a partir del campo **Apellido** de manera ascendente

### Instrucciones:

- Implementar la función **ordenarArreglo(arreglo)**, donde se ordenará el arreglo de registros a partir del campo **Apellido**.
- La función debe retornar el arreglo ordenado e imprimir por consola el contenido del mismo.
- Exporte la función empleando la siguiente sentencia de código:
  - **Module.exports.ordenarArreglo = ordenarArreglo;**
- Lectura recomendada: [https://www.w3schools.com/js/js\\_array\\_methods.asp](https://www.w3schools.com/js/js_array_methods.asp)

## Sprint Backlog:

### Historia de usuario

Yo como administrador requiero manipular la información guardada, con el fin de hacer un análisis de la misma.

### Actividades

2. Filtrar aquellos registros que contengan en el campo **Correo**, correos con direcciones de gmail.com

### Instrucciones:

- Implementar la función **filtrarCorreo(arreglo)**.
- La función debe de retornar el listado de los registros que cumplen las características del filtro e imprimir en consola dicho listado.
- Exporte la función empleando la siguiente sentencia de código:
  - **Module.exports.filtrarCorreo = filtrarCorreo;**
- Lectura recomendada: [https://www.w3schools.com/js/js\\_array\\_methods.asp](https://www.w3schools.com/js/js_array_methods.asp)

## 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 **registro.js**, de lo contrario no se calificará.

Para la evaluación de este sprint es **muy importante** que los valores de los atributos **id** de los elementos sean **exactamente iguales** a los indicados en las actividades del sprint 1. De igual manera, nombre las funciones **exactamente igual** a como se nombran en el instructivo de cada actividad del sprint actual.

```
let registros = [];  
  
function ordenarArreglo(arreglo) {  
  
}  
  
function filtrarCorreo(arreglo) {  
  
}  
  
function agregarRegistro() {  
  
}  
  
module.exports.registros = registros;  
module.exports.filtrarCorreo = filtrarCorreo;  
module.exports.ordenarArreglo = ordenarArreglo;  
module.exports.agregarRegistro = agregarRegistro;
```





[www.upb.edu.co/es/mision-tic](http://www.upb.edu.co/es/mision-tic)  
#MisiónTICSomosTodos