



## Actividad

### Sprint 1. Creación de un repositorio en Github

### Desarrollo de aplicaciones web

1. **Título:** Sprint 1. Creación de un repositorio en Github
2. **Modalidad:** Célula de desarrollo (trabajo en grupo de 5 integrantes)
3. **Resultado de aprendizaje:** se espera que, al finalizar la actividad, los estudiantes aprendan a manejar los comandos básicos para administrar los cambios en el proyecto sobre el repositorio de manera local mediante la herramienta Git, así como realizar mezclas de los cambios locales con la rama principal del proyecto en GitHub usando los pull request.
4. **Recursos:** Para lograr la creación de un repositorio en Github, debes ingresar a los enlaces y acceder a los recursos indicados que se presentan a continuación, como los que se les compartieron en la semana 1 en el material de estudio y el material de apoyo.

- Enlace para el manejo de repositorios públicos:  
<https://github.com/>
- Enlace para la herramienta Git del equipo local:  
<https://git-scm.com/download/win>
- Enlace para comandos básicos de Git:  
<https://www.freecodecamp.org/espanol/news/10-comandos-de-git-que-todo-desarrollador-deberia-saber/>

### 5. Indicaciones

- Conformar un equipo de cinco integrantes y ponerle un nombre al equipo.
- Distribuye el rol para cada uno de los integrantes, teniendo presente siempre que todos los del equipo deben ser desarrolladores. Los roles están definidos en la siguiente tabla:



Rol	Función del rol
1. Desarrollador	Generar las fuentes para el frontend y el backend
2. Analista	Levantar las necesidades del negocio para convertirlas en requerimientos del sistema
3. Product Owner	Conocer el producto <i>software</i> que se va a elaborar en el proyecto
4. Administrador de BD	Gestionar y disponer el motor de base de datos para el proyecto de desarrollo de <i>software</i>
5. Scrum master	Gestionar todas las ceremonias y actividades de la metodología Scrum, y asignar los recursos a cada rol

- Cada estudiante debe desarrollar las actividades correspondientes para cumplir con los objetivos.

Como analista de sistemas **necesito** realizar el proceso de integración de cada uno de los artefactos que se generen en el desarrollo de la aplicación web **y** almacenar, gestionar, versionar y realizar el seguimiento a los componentes creados en el proyecto sobre directorios clasificando los tipos de fuentes en el proyecto.

Se requiere construir un nuevo repositorio sobre la herramienta GitHub y definir sobre este repositorio las carpetas en las que se alojarán los artefactos que se generen a lo largo del ciclo de vida del proyecto de desarrollo de *software*, adicionando en la raíz del proyecto archivos iniciales como README y .gitignore para evitar versionar elementos que no requieren rastreo, seleccionando la plantilla (template) del lenguaje de programación con el que haremos nuestros códigos, que debe ser Node.



☒ **Add a README file**

This is where you can write a long description for your project.

☒ **Add .gitignore**

Choose which files not to track from a list of templates.

.gitignore template: **Node** ▼

The screenshot shows a GitHub repository interface. At the top, there are tabs for 'main' (selected), '1 branch', and '0 tags'. Below these are buttons for 'Go to file', 'Add file', and 'Code'. The main content area shows a commit history table with columns for the commit message, commit hash, time ago, and number of commits. The table lists three commits: 'RobinCG Update README.md' (3f3326e, 3 minutes ago, 2 commits), '.gitignore Initial commit' (5 minutes ago), and 'README.md Update README.md' (3 minutes ago). Below the table, there is a section for the 'README.md' file, which contains the text 'CursoMinTic Desarrollo de aplicaciones Web'.

Commit Message	Commit Hash	Time Ago	Commits
RobinCG Update README.md	3f3326e	3 minutes ago	2 commits
.gitignore Initial commit		5 minutes ago	
README.md Update README.md		3 minutes ago	

README.md

**CursoMinTic Desarrollo de aplicaciones Web**

Localmente, en el equipo de cómputo se debe tener instalado el Git para gestionar los cambios locales mediante el explorador de archivos de Windows o mediante comandos desde consola y así poder subir los cambios al gestor de repositorios GitHub en la nube.

En pocas palabras, el equipo debe:

- El scrum master debe crear un repositorio en la nube y dar permisos a cada integrante
- Cada integrante debe clonar el repositorio creado en su respectivo equipo local
- El administrador de base de datos debe crear una rama (Branch) y la carpeta "Presentación", en su equipo local y hacer pull request.
- Cada integrante debe adicionar su presentación al repositorio haciendo pull request

Nota: Se recomienda que el equipo se reúna por medio de una herramienta de vídeo llamada para realizar esta actividad con el fin de que cada estudiante conozca los detalles de las tareas realizadas.



- En Scrum se realizan entregas parciales y regulares de un producto final, y las actividades se priorizan de acuerdo a las necesidades para lograr la entrega de un producto completo.
- Para llevar a cabo dichas entregas debes primero registrar cada semana los objetivos trazados por cada integrante del grupo según las historias de usuario, para lo cual se empleará el programa Trello. De igual forma, todos los días se deben reunir los integrantes de manera virtual para diligenciar el formulario en el cual deben cumplir con el objetivo del Dailies, que es una reunión de 15 minutos donde cada uno de los integrantes deberá responder las siguientes preguntas:
  - ¿Qué hiciste ayer?
  - ¿Qué harás hoy?
  - ¿Hay impedimentos en tu camino? (blockers)
- En cada uno de los cinco sprints debes desarrollar la temática planteada. De igual manera, el equipo debe entregar el sprint en la semana asignada. Esta entrega recibirá una nota que será el soporte para la certificación, pero, aunque el trabajo sea grupal, la nota será de manera individual.

**6. Criterios de valoración de la evidencia:** Deberás entregar un repositorio creado en GitHub de tus proyectos de desarrollo mostrando las diferentes interacciones/aportes que se han tenido a lo largo del curso con la historia de cambio por parte de cada uno de los integrantes de la célula de desarrollo. Esta actividad tiene un valor del 20% y debes entregarla en la semana 2.

## **7. Protocolo de entrega**

**1) Formato:** el documento a entregar debe contener de lo siguiente:

- Portada con los nombres de los integrantes del grupo
- Descripción del proceso que se llevó a cabo para cumplir con los requerimientos de este sprint
- Ruta (URL) en la que se encuentra el repositorio GitHub



Se requiere compartir el enlace de Trello, en el que se deberán evidenciar los diferentes aportes de cada uno de los integrantes del proyecto.

Debes nombrar el archivo entregado indicando Sprint1\_Equipo\_XX (XX depende del nombre que le asignaron al equipo).

## 2) Medio

- Recurso **tarea**, disponible en el aula virtual, semana 2, en la plataforma Moodle.

## 3) Plazo máximo de entrega

- De acuerdo con el cronograma de actividades (entrega de semana 2).