

# **TP #1 INTRODUCCIÓN SOBRE GIT Y GITHUB**

**Institución:** Instituto Superior Politécnico de Córdoba

**Módulo:** Proyecto Integrador.

**Autor:** Mayrene Colmenares.

**Tutor:** Gonzalo Vera.

Abril, 2024

# **TRABAJO PRÁCTICO #1**

## **Resumen sobre el tutorial de Git y GitHub**

### **Parte 2: Introducción a Git y GitHub**

#### **1. Creación de una Cuenta en GitHub: Si aún no tienes una, crea una cuenta en GitHub.**

Cuenta creada: <https://github.com/mayrenecolmenares>

#### **2. Tutorial de Git/GitHub: Completar un tutorial básico de Git y GitHub. <https://classroom.github.com/a/iAxy1U65>**

Realizado.

#### **3. Informe de Aprendizaje: Escribir un resumen de lo aprendido en el tutorial.**

### **GIT**

Git es un sistema de control de versiones distribuido (DVCS) que permite a los desarrolladores rastrear los cambios en sus proyectos. Fue desarrollado originalmente por Linus Torvalds, el famoso creador del kernel del sistema operativo Linux, en 2005. En Git, la copia de trabajo del código de cada desarrollador es también un repositorio que puede albergar el historial completo de todos los cambios.

Git se utiliza para una variedad de propósitos, incluyendo:

**Seguimiento de cambios en el código:** Git permite a los desarrolladores rastrear los cambios en el código a lo largo del tiempo.

**Colaboración:** Git facilita la colaboración en proyectos de software. Los desarrolladores pueden trabajar en diferentes ramas del código al mismo tiempo y luego fusionar sus cambios para crear una versión final.

**Revisión de código:** Git se puede utilizar para la revisión de código. Los desarrolladores pueden compartir sus cambios con otros desarrolladores para obtener comentarios.

**Automatización:** Git se puede utilizar para automatizar tareas, como la construcción de software y la distribución de actualizaciones.

Los principales comandos de Git incluyen:

**git init:** Crea un nuevo repositorio local GIT.

**git clone:** Se usa para copiar un repositorio.

**git add:** Se usa para agregar archivos al área de preparación.

**git commit:** Crea una instantánea de los cambios y la guarda en el directorio git.

**git config:** Puede ser usado para establecer una configuración específica de usuario, como el email, nombre de usuario y tipo de formato, etc.

**git status:** Muestra la lista de los archivos que se han cambiado junto con los archivos que están por ser preparados o confirmados.

**git branch:** Se usa para crear, listar y eliminar ramas.

**git checkout:** Se usa para cambiar de una rama a otra.

## GITHUB

GitHub es una plataforma de desarrollo colaborativo que aloja proyectos utilizando el sistema de control de versiones Git. Fue comprada por Microsoft en 2018. Permite a los desarrolladores almacenar y administrar el código, llevando un registro de cambios. Generalmente, el código es abierto, lo que permite realizar proyectos compartidos y mantener el seguimiento detallado de su progreso. La plataforma GitHub también funciona como red social, conectando a los desarrolladores con los usuarios.

GitHub se utiliza para una variedad de propósitos, incluyendo:

**Almacenamiento de código:** Los desarrolladores pueden subir el código de sus aplicaciones y herramientas.

**Colaboración:** GitHub facilita la colaboración en proyectos de software. Los desarrolladores pueden trabajar en diferentes ramas del código al mismo tiempo y luego fusionar sus cambios para crear una versión final.

**Revisión de código:** GitHub se puede utilizar para la revisión de código. Los desarrolladores pueden compartir sus cambios con otros desarrolladores para obtener comentarios.

**Automatización:** GitHub se puede utilizar para automatizar tareas, como la construcción de software y la distribución de actualizaciones.

Las principales características de GitHub incluyen:

**Funciones colaborativas:** GitHub se caracteriza sobre todo por sus funciones colaborativas que ayudan a que todos puedan aportar su granito de arena para mejorar el código.

**Proyectos privados:** Si prefieres que tu código no se vea, también pueden crearse proyectos privados.

**Multiplataforma:** GitHub es una de las más utilizadas del mundo por los desarrolladores. Es multiplataforma, y tiene multitud de interfaces de usuario.

Los principales comandos de GitHub incluyen:

**git clone:** Se usa para clonar un repositorio.

**git branch:** Crea, lista y elimina ramas.

**git checkout:** Se usa para cambiar de una rama a otra.

**git status:** Muestra la lista de los archivos que se han cambiado junto con los archivos que están por ser preparados o confirmados.

**git add:** Se usa para agregar archivos al área de preparación.

**git commit:** Crea una instantánea de los cambios y la guarda en el directorio git.

**git push:** Sube todos los commits de la rama local al GitHub.

**git pull:** Actualiza tu repositorio local al commit más nuevo.