

## Git y GitHub Objetivo:

El estudiante desarrollará competencias para trabajar con Git y GitHub, aplicando conceptos fundamentales de control de versiones, colaboración en proyectos y resolución de conflictos, en un entorno simulado y guiado.

## Resultados de aprendizaje:

1. Comprender los conceptos básicos de Git y GitHub: Identificar y explicar los principales términos y procesos asociados con Git y GitHub, como repositorios, ramas, commits, forks, etiquetas y repositorios remotos.
2. Manejar comandos esenciales de Git: Ejecutar comandos básicos para crear, modificar, fusionar y gestionar ramas, commits y repositorios, tanto en local como en remoto.
3. Aplicar técnicas de colaboración en GitHub: Configurar y utilizar repositorios remotos, realizar forks, y gestionar pull requests para facilitar el trabajo colaborativo.
4. Resolver conflictos en un entorno de control de versiones: Identificar, analizar y solucionar conflictos de merge generados en un flujo de trabajo con múltiples ramas.

## Actividades

### 1) Contestar las siguientes preguntas utilizando las guías y documentación proporcionada (Desarrollar las respuestas):

#### 1. ¿Qué es GitHub?

GitHub es una plataforma web basada en Git que permite almacenar, gestionar y colaborar en proyectos de software. Facilita el control de versiones, la gestión de cambios en el código y la cooperación entre desarrolladores mediante herramientas como issues, pull requests y GitHub Actions.

#### 2. ¿Cómo crear un repositorio en GitHub?

- 1\_ Iniciar sesión en [GitHub](#).
- 2\_ Hacer clic en el ícono + en la esquina superior derecha y seleccionar New repository.
- 3\_ Ingresar un nombre para el repositorio y, opcionalmente, una descripción.
- 4\_ Seleccionar la visibilidad del repositorio (público o privado).
- 5\_ (Opcional) Inicializar el repositorio con un README, un .gitignore y una licencia.
- 6\_ Hacer clic en Create repository.

#### 3. ¿Cómo crear una rama en Git?

Para crear una nueva rama y cambiar a ella inmediatamente, se utiliza el siguiente comando:

**git checkout -b nombredelarama** (Este comando permite desarrollar nuevas funcionalidades sin afectar la rama principal)

#### 4. ¿Cómo cambiar a una rama en Git?

Para cambiar a una rama existente, se usa:

**git checkout nombredelarama**

También es posible utilizar el siguiente comando en versiones recientes de Git:

**git switch nombredelarama**

#### 5. ¿Cómo fusionar ramas en Git?

Para fusionar una rama en otra, se debe seguir el siguiente procedimiento:

Cambiar a la rama donde se desea hacer la fusión:

**git checkout main**

Ejecutar el comando de fusión:

**git merge nombredelarama**

Si existen conflictos, Git solicitará resolverlos antes de completar la fusión.

#### 6. ¿Cómo crear un commit en Git?

Para registrar cambios en el repositorio, se deben seguir estos pasos:

Agregar los archivos al área de preparación:

**git add .**

Crear el commit con un mensaje descriptivo:

**git commit -m "Descripción de los cambios"**

#### 7. ¿Cómo enviar un commit a GitHub?

Para subir un commit al repositorio remoto, se usa:

**git push origin nombredelarama**

Si es la primera vez que se sube la rama, se debe utilizar:

**git push -u origin nombredelarama**

#### 8. ¿Qué es un repositorio remoto?

Un repositorio remoto es una versión del repositorio almacenada en un servidor externo, como GitHub. Permite sincronizar cambios entre diferentes desarrolladores y dispositivos.

#### 9. ¿Cómo agregar un repositorio remoto a Git?

Para vincular un repositorio local con un remoto, se ejecuta:

**git remote add origin https://github.com/usuario/nombre-repositorio.git**

#### 10. ¿Cómo empujar cambios a un repositorio remoto?

Para subir los cambios locales al repositorio remoto, se utiliza:

**git push origin main**

Si se trabaja en otra rama, se debe reemplazar main con el nombre de la rama correspondiente.

#### 11. ¿Cómo traer cambios de un repositorio remoto?

Para descargar y fusionar los cambios del repositorio remoto, se emplea:  
bash

#### 12. ¿Qué es el historial de Git?

El historial de Git es el registro de todos los commits realizados en el repositorio, mostrando qué cambios se hicieron, quién los realizó y cuándo.

#### 13. ¿Cómo ver el historial de Git?

Para visualizar el historial de commits, se usa:

**git log**

Para una vista más resumida:

**git log --oneline**

#### 14. ¿Cómo buscar en el historial de Git?

Para encontrar commits que contengan una palabra clave en su mensaje, se ejecuta:

**git log --grep="palabra-clave"**

#### 15. ¿Cómo borrar el historial de Git?

El historial de Git no se puede borrar directamente, pero se puede reescribir con:

**git reset --hard commit-id**

Este comando restaura el repositorio al estado del commit especificado, eliminando cambios posteriores.

#### 16. ¿Qué es un repositorio privado en GitHub?

Un repositorio privado es aquel que solo puede ser accedido por usuarios autorizados. No es visible públicamente y requiere permisos para su consulta y modificación.

#### 17. ¿Cómo crear un repositorio privado en GitHub?

Durante la creación de un repositorio, se debe seleccionar la opción Private en la configuración de visibilidad.

#### 18. ¿Cómo invitar a alguien a un repositorio privado en GitHub?

1\_ Ingresar a la pestaña Settings del repositorio.

2\_ Acceder a Manage access.

3\_ Hacer clic en Invite a collaborator e ingresar el usuario de GitHub o su correo electrónico.

4\_ La persona recibirá una invitación para acceder al repositorio.

### 19. ¿Qué es un repositorio público en GitHub?

Un repositorio público es accesible para cualquier usuario de GitHub. Puede ser visualizado, clonado y descargado por cualquier persona.

### 20. ¿Cómo crear un repositorio público en GitHub?

Al crear un nuevo repositorio, se debe seleccionar la opción Public en la configuración de visibilidad.

### 21. ¿Cómo compartir un repositorio público en GitHub?

Para compartir un repositorio público, se puede:

- \_ Copiar y compartir la URL del repositorio desde GitHub.
- \_ Usar el comando `git remote -v` para obtener la URL y compartirla con otros usuarios.

## 2) Realizar la siguiente actividad:

- Crear un repositorio.
  - Dale un nombre al repositorio.
  - Elije el repositorio sea público.
  - Inicializa el repositorio con un archivo.

## Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere?  
[Import a repository.](#)

---

*Required fields are marked with an asterisk (\*).*


Owner \*

Repository name \*

 joaquinriveros

/

GitHub\_Objetivo


 GitHub\_Objetivo is available.

Great repository names are short and memorable. Need inspiration? How about **potential-palm-tree** ?


**Description** (optional)

---

☒

 **Public**  
Anyone on the internet can see this repository. You choose who can commit.

☐

 **Private**  
You choose who can see and commit to this repository.

---

**Initialize this repository with:**

☒

**Add a README file**  
This is where you can write a long description for your project. [Learn more about READMEs.](#)

**Add .gitignore**

.gitignore template: **None**

- **Agregando un Archivo**
  - Crea un archivo simple, por ejemplo, "mi-archivo.txt".
  - Realiza los comandos `git add .` y `git commit -m "Agregando mi-archivo.txt"` en la línea de comandos.
  - Sube los cambios al repositorio en GitHub con `git push origin main` (o el nombre de la rama correspondiente).

```
C:\Users\Administrador>cd OneDrive/Desktop

C:\Users\Administrador\OneDrive\Desktop>git clone https://github.com/joaquinriveros/GitHub_Objetivo.git
Cloning into 'GitHub_Objetivo'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.

C:\Users\Administrador\OneDrive\Desktop>_
```

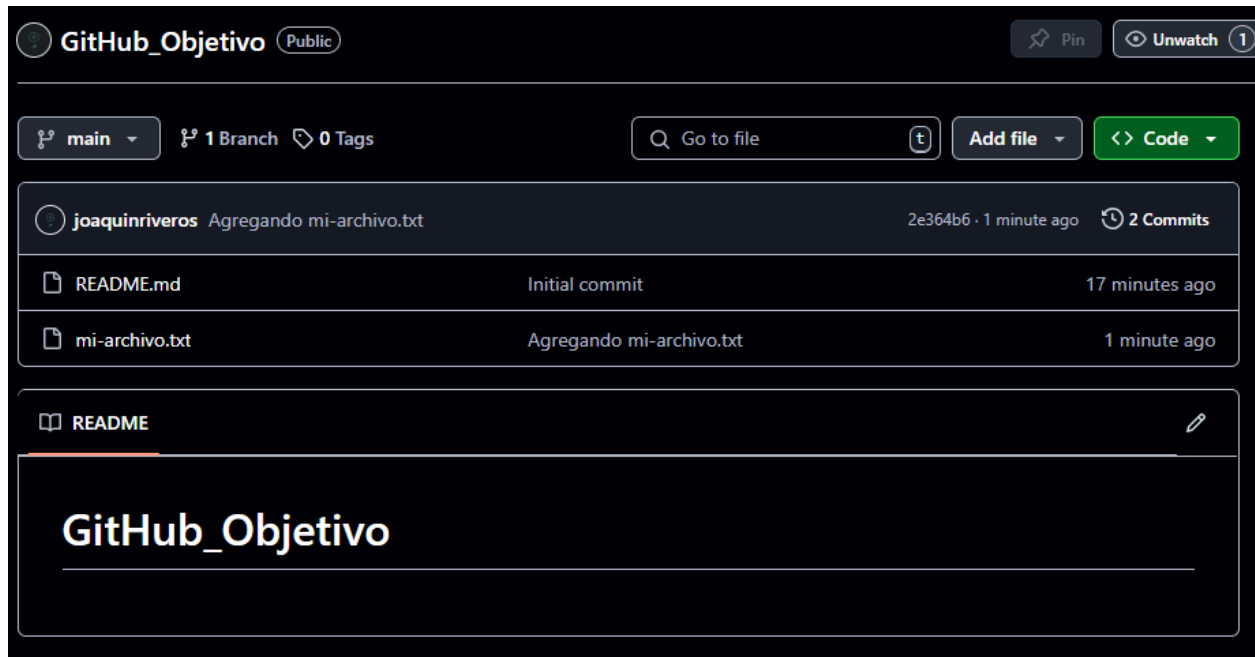
```
Directorio de C:\Users\Administrador\OneDrive\Desktop\GitHub_Objetivo

01/04/2025  22:06    <DIR>          .
01/04/2025  22:06    <DIR>          ..
01/04/2025  22:06                32 mi-archivo.txt
01/04/2025  22:01                17 README.md
                2 archivos                49 bytes
                2 dirs 14.542.278.656 bytes libres
```

```
C:\Users\Administrador\OneDrive\Desktop\GitHub_Objetivo>git add .

C:\Users\Administrador\OneDrive\Desktop\GitHub_Objetivo>git commit -m "Agregando mi-archivo.txt"
[main 2e364b6] Agregando mi-archivo.txt
 1 file changed, 1 insertion(+)
 create mode 100644 mi-archivo.txt
```

```
C:\Users\Administrador\OneDrive\Desktop\GitHub_Objetivo>git push origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 326 bytes | 326.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/joaquinriveros/GitHub_Objetivo.git
 48c97b2..2e364b6  main -> main
```



- **Creando Branchs**
  - **Crear una Branch**
  - **Realizar cambios o agregar un archivo**
  - **Subir la Branch**

```
C:\Users\Administrador\OneDrive\Desktop\GitHub_Objetivo>git checkout -b nueva-rama
Switched to a new branch 'nueva-rama'

C:\Users\Administrador\OneDrive\Desktop\GitHub_Objetivo>git branch
* main
  nueva-rama

C:\Users\Administrador\OneDrive\Desktop\GitHub_Objetivo>echo "Este es un archivo en la nueva rama." > archivo-rama.txt

C:\Users\Administrador\OneDrive\Desktop\GitHub_Objetivo>git add .

C:\Users\Administrador\OneDrive\Desktop\GitHub_Objetivo>git commit -m "Agregando archivo-rama.txt en nueva-rama"
[nueva-rama 8160521] Agregando archivo-rama.txt en nueva-rama
 1 file changed, 1 insertion(+)
 create mode 100644 archivo-rama.txt

C:\Users\Administrador\OneDrive\Desktop\GitHub_Objetivo>git push origin nueva-rama
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 379 bytes | 379.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'nueva-rama' on GitHub by visiting:
remote:   https://github.com/joaquinriveros/GitHub_Objetivo/pull/new/nueva-rama
remote:
To https://github.com/joaquinriveros/GitHub_Objetivo.git
 * [new branch]      nueva-rama -> nueva-rama
```

GitHub\_OjetivoPublic

PinUnwatch1

main2 Branches0 Tags

Go to filetAdd fileCode

Switch branches/tags

Find or create a branch...

BranchesTags

maindefault

nueva-rama

View all branches

joaquinriveros/nueva-rama7834e7c · 1 minute ago4 Commits

Initial commit21 minutes ago

Agregando archivo-rama.txt en nueva-rama2 minutes ago

Agregando mi-archivo.txt6 minutes ago

GitHub\_Ojetivo

GitHub\_OjetivoPublic

PinUnwatch1

main2 Branches0 Tags

Go to filetAdd fileCode

joaquinriveros Merge pull request #1 from joaquinriveros/nueva-rama7834e7c · now4 Commits

README.mdInitial commit21 minutes ago

archivo-rama.txtAgregando archivo-rama.txt en nueva-rama1 minute ago

mi-archivo.txtAgregando mi-archivo.txt5 minutes ago

README

GitHub\_Ojetivo