

TP N°2

GIT Y GITHUB

Alumno: Tolaba Victor

Materia: Programación 1

- ¿Qué es GitHub?

Es una comunidad donde podemos compartir nuestros repositorios de forma publica o privada. Creada una cuenta en github se va a disponer de un perfil parecido a una red social donde se pueden cargar nuestros repositorios, seguir a otros usuarios, ver lo que suben e incluso proponer cambios y acoplarlos al repositorio original

- ¿Cómo crear un repositorio en GitHub?

- 1- Ir a <https://github.com> e ingresar a la cuenta
- 2- Hacer click en el botón "+" ubicado en la esquina superior derecha
- 3- Seleccionar "New repository"
- 4- Escribir el nombre del repositorio
- 5- Se puede seleccionar "Public" para que todos puedan verlo o "Private" para que solo el autor y a quienes uno autorice pueda verlo
- 6- Hacer click en "Create repository" y listo, ya estará creado

- ¿Cómo crear una rama en Git?

Se puede hacer desde la terminal con los siguientes comandos:

`git branch nombre-de-la-rama`

Por ejemplo:

`git branch nueva-funcionalidad`

- ¿Cómo cambiar a una rama en Git?

Se puede cambiar a una rama siguiendo el siguiente comando:

`git checkout nombre-de-la-rama`

O en una sola línea, mas común hoy en día:

`git checkout -b nombre-de-la-rama`

- ¿Cómo fusionar ramas en Git?

Ir a la rama donde se quiere hacer el merge:
git checkout main

Fusionar la rama que contiene los cambios:
Git merge nombre-de-la-rama

- ¿Cómo crear un commit en Git?

Es el proceso de guardar los cambios realizados en el repositorio local. Se realiza siguiendo estos pasos básicos:

Verifica los archivos modificados:
git status

Añadir los archivos al área de preparación (staging area):

git add archivo.txt

O todos los archivos modificados:

git add .

Crear el commit con un mensaje:

git commit -m "Mensaje del cambio realizado"

Ejemplo:

git commit -m "Corrige errores en validación de formulario"

- ¿Cómo enviar un commit a GitHub?

Se hace mediante un push:

git push origin nombre-de-la-rama

- ¿Qué es un repositorio remoto?

Un repositorio remoto en Git es una versión de tu proyecto que está alojado en un servidor externo como GitHub, GitLab, Bitbucket, etc.

Sirve para compartir código con otras personas, hacer respaldo del trabajo de uno y colaborar en equipo

- ¿Cómo agregar un repositorio remoto a Git?

- 1- Crear un repositorio vacío en GitHub:

Ir a <https://github.com/bew> y crear un repositorio sin inicializar

- 2- En el proyecto local, abrir la terminal e ir a la carpeta del proyecto:

cd /ruta/de/tu/proyecto

- 3- Inicializar el repositorio local:
git init
- 4- Agregar el repositorio remoto con git remote add:
Git remote add origin https://github.com/tu_usuario/tu_repositorio.git
- 5- Verificar que se haya agregado correctamente:
git remote -v

- ¿Cómo empujar cambios a un repositorio remoto?

git push origin nombre-de-la-rama

- ¿Cómo tirar de cambios de un repositorio remoto?

git pull origin nombre-de-la-rama

- ¿Qué es un fork de repositorio?

Es una copia completa de un repositorio hecha en la cuenta propia. Permite trabajar de forma independiente sobre el proyecto original, sin afectar el repositorio fuente.

- ¿Cómo crear un fork de un repositorio?

- 1- Ir al repositorio original en GitHub. Ejemplo:
https://github.com/otro_usuario/proyecto-ejemplo
- 2- Hacer click en el botón "Fork"
- 3- GitHub creará una copia exacta del repositorio original en la cuenta de uno
- 4- Clonar el fork en la computadora:
git clone https://github.com/tu_usuario/proyecto-ejemplo.git
cd proyecto-ejemplo

- ¿Cómo enviar una solicitud de extracción (pull request) a un repositorio?

- 1- Hacer un fork del repositorio
- 2- Clonar el fork en la computadora:
git clone https://github.com/tu_usuario/repositorio-forkeado.git
cd repositorio-forkeado
- 3- Crear una nueva rama para los cambios:
Git checkout -b nombre-de-la-rama
- 4- Hacer cambios, guardar y hacer commit:
Git add .
Git commit -m "Corrige errores ortográficos"
- 5- Empujar la rama al fork en github:

git push origin nombre-de-la-rama

6- Abrir el pull request en github:

Ir al fork en github

Click en el botón "Compare & pull request"

Escribir un título y descripción breve de los cambios

Asegurarse de que el PR se está enviando a la rama correcta del repositorio original

Hacer click en "Create pull request"

- ¿Cómo aceptar una solicitud de extracción?

Ve al repositorio

- Ejemplo: https://github.com/tu_usuario/tu_repositorio

Haz clic en la pestaña "Pull requests"

- Se encuentra justo debajo del nombre del repositorio.

Selecciona la pull request que quieres revisar

- Verás una lista con el título y la rama de cada PR enviada.

Revisa los cambios

- Puedes ver los *commits*, los *archivos cambiados* y los comentarios.
- También puedes dejar comentarios o solicitar cambios.

Haz clic en "Merge pull request"

- Aparecerá un botón verde si no hay conflictos.

Confirma el merge

- Puedes editar el mensaje del merge o dejar el que aparece por defecto.
- Luego haz clic en "Confirm merge".

- ¿Qué es una etiqueta en Git?

Es un marcador que se usa para identificar un punto específico en la historia del repositorio, normalmente para señalar una versión estable o un lanzamiento importante. Las etiquetas no cambian, apuntan a un commit fijo

- ¿Cómo crear una etiqueta en Git?

1-Crear una etiqueta ligera:

`git tag nombre-etiqueta`

2-Crear una etiqueta anotada:

`git tag -a nombre-etiqueta -m "Mensaje de la etiqueta"`

- ¿Cómo enviar una etiqueta a GitHub?

1- Una etiqueta específica:

`git push origin nombre-etiqueta`

2- Todas las etiquetas:

`git push origin --tags`

- ¿Qué es un historial de Git?

Es un registro completo de todos los commits realizados en un repositorio.

Muestra qué cambios se hicieron, quién los hizo, cuándo y con qué mensaje.

- ¿Cómo ver el historial de Git?

1- De forma completa:

`git log`

2- De forma compacta:

`git log --oneline`

- ¿Cómo buscar en el historial de Git?

1- Buscar commits por palabras clave en el mensaje

`git log --grep="palabra"`

2- Buscar commits que modificaron un archivo específico:

`git log -- archivo.txt`

3- Buscar por autor:

`git log --author="nombre"`

4- Buscar por contenido de los cambios:

`git log -S"texto-buscado"`

- ¿Cómo borrar el historial de Git?

Opcion 1:

1. Borra todo el historial y crea un nuevo commit inicial

`rm -rf .git`

`git init`

`git add .`

`git commit -m "Nuevo inicio del historial"`

2. Conecta al remoto (si aplica)

```
git remote add origin https://github.com/tu_usuario/tu_repo.git
```

3. Fuerza el push (sobrescribe todo en remoto)

```
git push -f origin main
```

Opcion 2:

```
git filter-branch --force --index-filter \  
"git rm --cached --ignore-unmatch ruta/al/archivo" \  
--prune-empty --tag-name-filter cat -- --all
```

Opcion 3:

```
java -jar bfg.jar --delete-files archivo_secreto.txt
```

- ¿Qué es un repositorio privado en GitHub?

Es un proyecto de código al que solo tienen acceso el autor y las personas autorizadas por él. No es visible para el público en general.

- ¿Cómo crear un repositorio privado en GitHub?

- 1- Ir a github.com/new
- 2- Poner un nombre al repositorio
- 3- Marcar la opción "Private"
- 4- Crear el repositorio

- ¿Cómo invitar a alguien a un repositorio privado en GitHub?

- 1- Ir al repositorio privado en github
- 2- Click en la pestaña "Settings"
- 3- Click en "Collaborators"
- 4- Click en "Add people" o "Invite a collaborator"
- 5- Escribir el nombre de usuario o el correo de github de la persona que se quiere invitar
- 6- Click en "Add" o "Send invitation"

- ¿Qué es un repositorio público en GitHub?

Es un proyecto de código que cualquier persona puede ver, clonar y descargar en github. Es ideal para proyectos de código abierto, portafolios, colaboraciones públicas o para aprender de otro

- ¿Cómo crear un repositorio público en GitHub?

- 1- Ir a <https://github.com/new>
- 2- Rellenar los campos "Repository name" y "Description"
- 3- Seleccionar la opción "Public"
- 4- Click en "Create repository"

- ¿Cómo compartir un repositorio público en GitHub?

Se copia la URL del repositorio y se comparte por donde uno quiera

Ejemplo: https://github.com/tu_usuario/nombre-repositorio