

# TP2 – Git y GitHub – Santiago Caiciia Massello

## 2.1)

### - ¿Qué es GitHub?

Es una plataforma para alojar proyectos utilizando el sistema de control de versiones GIT. Permite a los desarrolladores crear, almacenar, editar, administrar y compartir su código.

### - ¿Cómo crear un repositorio en GitHub?

- 1) Ir a GitHub e iniciar sesión en una cuenta de GitHub.
- 2) Hacer clic en botón "New" para crear un nuevo repositorio.
- 3) Asignar un nombre y seleccionar si el repositorio va a ser público (Public) o privado (Private). Opcionalmente se puede elegir si se crea también un archivo README.

### - ¿Cómo crear una rama en Git?

Utilizando el comando `git branch [nombre de la rama]`. Donde en [nombre de la rama] se coloca el nombre que se desea colocarle a la rama.

### - ¿Cómo cambiar a una rama en Git?

Con el comando `git checkout [nombre de la rama]`. Donde en [nombre de la rama] se coloca el nombre de la rama a la que se desea cambiar.

### - ¿Cómo fusionar ramas en Git?

Se utiliza el comando `git merge [nombre de la rama]`. Este comando fusiona [nombre de la rama] con la rama en donde se está posicionado actualmente.

### - ¿Cómo crear un commit en Git?

Se utiliza el comando `git commit`, adicionalmente se agrega `-m "mensaje"` que te permite agregar un mensaje al commit.

- ¿Cómo enviar un commit a GitHub?

1) Se debe agregar el repositorio remoto, si no existe, al repositorio local con el comando `git remote add origin <url>`. Origin es el nombre que se le da al repositorio remoto en nuestro repositorio local, esto se puede cambiar en caso de que necesite configurar un segundo repositorio remoto. Y en `<url>` se coloca la dirección donde se encuentra el repositorio remoto.

2) Una vez agregado el repositorio remoto, enviamos el commit a este con el comando `git push origin master`. Donde origin es el nombre del repositorio remoto donde queremos enviar nuestro commit, y master es la rama de tu repositorio local que quieres enviar al repositorio remoto, es decir, donde se encuentra nuestro commit.

- ¿Qué es un repositorio remoto?

Es una versión de un proyecto Git, almacenado en un servidor o en la nube. Esta puede ser accedida y gestionada por múltiples personas. Estos repositorios normalmente están almacenados en GitHub, GitLab, Bitbucket o un servidor privado.

- ¿Cómo agregar un repositorio remoto a Git?

Se realiza con el comando `git remote add origin <url>`.

- ¿Cómo empujar cambios a un repositorio remoto?

Se utiliza el comando `git push origin master` o `git push origin main` (dependiendo del nombre que se le dio al repositorio remoto y a la rama).

- ¿Cómo tirar de cambios de un repositorio remoto?

Utilizando el comando `git pull origin main`. Donde origin es el nombre que se le dio al repositorio remoto y main es la rama, del repositorio local, donde se quiere aplicar los cambios.

- ¿Qué es un fork de repositorio?

Fork es una copia de un repositorio en tu cuenta de GitHub, GitLab, etc. Es decir, se hace una copia de un repositorio determinado y se almacena en tu cuenta de GitHub, GitLab, etc.

- ¿Cómo crear un fork de un repositorio?

- 1) Iniciar sesión en GitHub.
- 2) Ve al repositorio al que quieres realizar el fork.
- 3) Haz clic en el botón "Fork" en la esquina superior derecha.
- 4) GitHub creará una copia del repositorio en tu cuenta.

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

- 1) Realizar un fork del repositorio.
- 2) Clonar el fork en la máquina local (`git clone <url>`).
- 3) Es recomendable no trabajar en la rama principal, por lo que, se recomienda realizar un branch (`git branch -b <nombre_rama>`). Lo que realiza el `-b` en el `git branch` es automáticamente hacer el checkout de esa rama. Es decir, posicionarse en la rama una vez creada.
- 4) Realizar los cambios necesarios en el repositorio y luego agregar los cambios al repositorio local (`git add .`).
- 5) Confirmar los cambios realizando un commit (`git commit -m <mensaje>`).
- 6) Aplicar los cambios a tu fork con un push (`git push <nombre_remoto> <nombre_rama>`).
- 7) Ir a la página del fork en nuestro GitHub, hacer clic en "Contribute" y luego "Open pull request".
- 8) Verificar que los datos que se compartan con el repositorio original sean correctos.
- 9) Escribir un título y una descripción clara, explicando que se hizo y por qué y hacer clic en "Create pull request".

- ¿Cómo aceptar una solicitud de extracción?

- 1) Iniciar sesión en GitHub e ir a la página del repositorio.
- 2) Hacer clic en la pestaña "Pull request" en la parte superior de la página del repositorio.
- 3) En la lista de solicitudes, hacer clic la que se desea revisar. Se pueden revisar los commits, los cambios en los archivos, etc.
- 4) Opcionalmente, se puede comentar las líneas modificadas o archivos específicos.
- 5) Si se desea realizar una retroalimentación, hay que hacer clic en "Review changes" y escribir el comentario deseado sobre los cambios propuestos.
- 6) Seleccione "Approve" para aprobar la combinación de los cambios propuestos en la solicitud de incorporación de cambios. Se recomienda dejar un comentario.
- 7) Por último, Haz clic en el botón "Merge pull request". Hay tres métodos de merge: Merge commit (Mantiene el historial con un commit adicional, opción por defecto), Squash and merge (Combina todos los commits en uno solo) y Rebase and merge.
- 8) Confirma haciendo clic en "Confirm merge".

- ¿Qué es un etiqueta en Git?

Es un marcador inmóvil que se usa para señalar puntos importantes en la historia de un repositorio, como versiones de lanzamiento. A diferencia de las ramas, las etiquetas no cambian una vez creadas. A diferencia de las ramas, las etiquetas no cambian una vez creadas.

- ¿Cómo crear una etiqueta en Git?

1) Etiquetas ligeras:

- Son simplemente un puntero a un commit.
- No contienen metadatos adicionales.
- `git tag <nombre_etiqueta>`

2) Etiquetas anotadas:

- Guardan información adicional como autor, fecha y un mensaje.
- Se almacenan en la base de datos de Git de forma más permanente.

- `git tag -a <nombre_etiqueta> -m "Descripción"`

- ¿Cómo enviar una etiqueta a GitHub?

`git push origin <nombre_etiqueta>`

- ¿Qué es un historial de Git?

El historial de Git es el registro de todos los cambios realizados en un repositorio, incluyendo commits, ramas, fusiones (merges) y etiquetas. Permite rastrear la evolución del proyecto y revertir cambios si es necesario.

- ¿Cómo ver el historial de Git?

Para ver el historial git, se utiliza el comando: `git log`.

- ¿Cómo buscar en el historial de Git?

1) Buscar commits por palabra clave: `git log --grep="palabra clave"`

2) Buscar commits por autor: `git log --author="Nombre del Autor"`

3) Buscar commits por fecha: `git log --since="2024-01-01"` o

`git log --until="2024-03-26"`

4) Buscar commits que afecten a un archivo específico:

`git log -- nombre-del-archivo`

5) Buscar una línea específica en un archivo:

`git blame nombre-del-archivo`

(Esto mostrará cada línea del archivo junto con el commit y el autor que la modificó)

6) Combinar filtros para búsquedas más precisas.

- ¿Cómo borrar el historial de Git?

- Eliminar el historial de commits recientes:

1. Hacer un reset a n commit anteriores: `git reset --hard HEAD~n`.

2. Para aplicar el cambio en el repositorio remoto hay que forzar un push: `git push origin <nombre_rama> --force`

- Eliminar un archivo del historial completo del repositorio:
  1. Eliminar el archivo de todo el historial del repositorio local:  
`git filter-branch --index-filter "git rm -rf --cached --ignore-unmatch ruta/al/archivo" HEAD`  
 (Reemplazar ruta/al/archivo por la ruta al archivo que deseas eliminar).
  2. Después, forzar el push al repositorio remoto: `git push origin <nombre_rama> --force`

- ¿Qué es un repositorio privado en GitHub?

Un repositorio privado en GitHub es un repositorio cuyo contenido solo es accesible para los propietarios y colaboradores autorizados. A diferencia de los repositorios públicos, no está visible para el público en general.

- ¿Cómo crear un repositorio privado en GitHub?

- 1) Ir a GitHub e iniciar sesión en una cuenta de GitHub.
- 2) Hacer clic en botón "New" para crear un nuevo repositorio.
- 3) Asignar un nombre y seleccionar repositorio privado (Private).  
Opcionalmente se puede elegir si se crea también un archivo README.

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

- 1) Iniciar sesión en GitHub.
- 2) Ir a la página del repositorio privado.
- 3) Seleccionar la pestaña Settings en la zona superior.
- 4) En la sección "Access", en la barra lateral, seleccionar "Collaborators".
- 5) En "Manage access" hacer clic en "Add people".
- 6) Escribir el nombre de la persona que se quiere agregar y luego en "Add to repository".
- 7) La persona invitada recibirá un correo y una vez aceptada la invitación, tendrá acceso al repositorio.

- ¿Qué es un repositorio público en GitHub?

Un repositorio público en GitHub es un repositorio cuyo contenido está disponible para cualquier persona en Internet. Cualquier usuario puede ver su código, descargarlo y, en algunos casos, contribuir mediante forks o pull requests.

- ¿Cómo crear un repositorio público en GitHub?

- 1) Ir a GitHub e iniciar sesión en una cuenta de GitHub.
- 2) Hacer clic en botón "New" para crear un nuevo repositorio.
- 3) Asignar un nombre y seleccionar repositorio público (Public).  
Opcionalmente se puede elegir si se crea también un archivo README.

- ¿Cómo compartir un repositorio público en GitHub?

Simplemente compartiendo la dirección web del repositorio o con el siguiente comando:

```
git clone https://github.com/direccion_del_repositorio.git
```

(reemplazar dirección del repositorio que desea compartir)

2) [https://github.com/scaiciia/TUPaD\\_TP2E2](https://github.com/scaiciia/TUPaD_TP2E2)

3) <https://github.com/scaiciia/conflict-exercise>