

TP2: GIT Y GITHUB

Alumno: Hernán Gabriel Fernández Molbert

Enlace GitHub: <https://github.com/GabiFernandezMolbert>

- **¿Qué es GitHub?**

GitHub es una página que representa una comunidad en la cual se pueden compartir proyectos de forma pública o privada. En ella se pueden ver repositorios de otros desarrolladores, clonarlos y trabajar en ellos.

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

Para crear un repositorio en GitHub primero hay que crear un usuario en esa comunidad, luego hacemos click en *“create new repository”*, ingresamos el nombre del repositorio, establecemos si va a ser público o privado, si queremos incluir un archivo *“read me”* y finalizamos haciendo click en *“create repository”*.

- **¿Cómo crear una rama en Git?**

Para crear una rama en git lo hacemos con el comando *“git branch”* y colocamos al lado el nombre que vamos a definir para dicha rama. Por ejemplo, `git branch rama1`.

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

Al crear una rama (por ejemplo, rama1) por defecto nos paramos en la rama en la que estabamos. Con el fin de movermos a ella, debemos

usar el comando *git checkout* y el nombre de la rama (en este caso, *git checkout rama1*).

- **¿Cómo fusionar ramas en Git?**

Al tener 2 o mas ramas en Git, estas se pueden y para ello debemos estar en la rama main (*git checkout main*). Una vez que realizamos esto, usamos el comando “*git merge*” (en nuestro ejemplo, *git merge rama1*).

- **¿Cómo crear un commit en Git?**

Para crear un commit en git solo primero debemos hacer un *add* . para agregar los nuevos archivos. Una vez realizado este paso, debemos usar el comando *git commit* y agregar una descripción. Por ejemplo, *git commit -m “Mi primer repositorio”*.

- **¿Cómo enviar un commit a GitHub?**

Para enviar un commit a GitHub debemos usar el comando *git push*. Una vez creado el commit con su respectiva descripción, ingresamos en git bash “*git push origen main*” (en el caso de que el commit se encuentre en esa rama, sino debemos poner su nombre).

- **¿Qué es un repositorio remoto?**

Un repositorio remoto es aquel que se aloja en GitHub o cualquier otro sitio que tenga este mismo fin; a diferencia del repositorio local que se encuentra en nuestra pc, este se encuentra en la nube.

- **¿Cómo agregar un repositorio remoto a Git?**

Para enviar un repositorio a Git debemos dirigirnos al repositorio alojado en GitHub, hacer click en *fork*, *New Fork*. Una vez realizados estos

pasos, en nuestro git bash pegamos la *url* proporcionada para la clonación con el comando *git clone* (*git clone nombredelaurl*).

- **¿Cómo empujar cambios a un repositorio remoto?**

Con el comando *git push* podemos subir los cambios que realizamos a un repositorio remoto (por ejemplo, *git push origin nombredelarama*).

- **¿Cómo tirar de cambios de un repositorio remoto?**

Para tirar de cambios de un repositorio remoto o traer las modificaciones realizadas desde un repositorio remoto a nuestra pc, debemos utilizar el comando *git pull origin nombredelarama*.

- **¿Qué es un fork de repositorio?**

Hacer un *fork* de un repositorio implica una copia de un repositorio remoto a nuestra pc; en él se pueden realizar modificaciones sin afectar el repositorio original.

- **¿Cómo crear un fork de un repositorio?**

Para crear un *fork* de un repositorio debemos dirigirnos a la web de *GitHub*, buscar el repositorio que nos interesa y hacer click en el botón *fork*. De esa manera, se creará una copia del repositorio en nuestro perfil.

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

En la página de *GitHub*, ingresamos a nuestro fork, hacemos click en *pull request*, *New Pull Request* y, por último, *Create Pull Request*.

- **¿Cómo aceptar una solicitud de extracción?**

Buscamos la solicitud de extracción en el repositorio, en *Pull Request* y hacemos click en *Merge Pull Request*

- **¿Qué es un etiqueta en Git?**

Una etiqueta es un manera de identificar commits, esto implica que se pueden realizar checkout utilizando esas etiquetas. Sirven para facilitar la navegación.

- **¿Cómo crear una etiqueta en Git?**

Para crear una etiqueta en git se utiliza el comando *git tag nombredeltag* e incluso se puede agregar una descripción usando *-m*.

- **¿Cómo enviar una etiqueta a GitHub?**

Para subir una etiqueta a GitHub ingresamos en bash: *git push origin nombredeltag*.

- **¿Qué es un historial de Git?**

El historial de git permite acceder a las modificaciones que se hayan realizado en el proceso de creación y desarrollo del repositorio.

- **¿Cómo ver el historial de Git?**

Para ver el historial de Git se puede utilizar el comando *git log*, el cual permite ver los diferentes commits realizados en un repositorio.

- **¿Cómo borrar el historial de Git?**

Se puede utilizar el comando *git reset* para borrar el historial de Git.

- **¿Qué es un repositorio privado en GitHub?**

Un repositorio privado en GitHub implica que este disponible sólo para mi y las personas que yo autorice a verlo a diferencia del público que puede ser visto por cualquier usuario de github.

- **¿Cómo crear un repositorio privado en GitHub?**

Al momento de crear el repositorio debemos marcar la casilla que indique que ese repositorio va a ser privado.

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

Acceder al repositorio privado, *settings*, *collaborators*, *add people* e ingresamos el usuario que queramos que tenga acceso a nuestro repositorio.

- **¿Qué es un repositorio público en GitHub?**

Un repositorio público significa que cualquier persona puede tener acceso a él.

- **¿Cómo crear un repositorio público en GitHub?**

Al momento de crear el repositorio debemos marcar la casilla que indique que ese repositorio va a ser público.

- **¿Cómo compartir un repositorio público en GitHub?**

Acceder al repositorio público, *settings*, *collaborators*, *add people* e ingresamos el usuario que queramos que tenga acceso a nuestro repositorio.