

- 1) GitHub es una plataforma en la nube que permite almacenar, compartir y editar código.
- 2) Debemos tener una cuenta en GitHub, elegimos nuevo repositorio, elegimos si es público o privado.
- 3) Podemos crear una rama utilizando **git branch** nombre_de_rama_nueva
- 4) Podemos cambiarnos a una rama utilizando el comando **git checkout** rama-destino
- 5) Para fusionar una rama en git debemos colocarnos en la rama que queremos que sea la que se desean hacer la fusión ejemplo **main** y usar el comando **git merge** rama-a-fusionar
- 6) Paso 1 para crear un commit es: Agregar los archivos con **git add** . (todo) o **git add** *nombrearchivo* y luego hacer un **git commit -m** "Hola probando el commit "
- 7) Podemos subir el archivo a github utilizando **git push** origin nombre-de-la-rama
- 8) Un repositorio remoto en Git es una versión del repositorio que se almacena en un servidor de internet ejemplo: EN Github.
- 9) Se puede añadir un directorio remoto utilizando: **git remote add** origin *url*
- 10) Para subir cambios al directorio remoto se utiliza: **git push -u origin master**
Y **git push** (Luego de la primera vez)
- 11) Podemos tirar de los cambios utilizando: **git pull origin master**
- 12) Un **fork** en Git es una copia de un repositorio que se crea en nuestra cuenta.
- 13) Crear un fork: Ir a github hacer click en el botón "**Fork**".
Nos va a pedir elegir en qué cuenta hacer el fork, elegir una
Esperar y se va a crear.
- 14) Ir al repositorio en GitHub (<https://github.com/tu-usuario/mi-repo-forkeado>).
Hay un menú de Pull Request. Compare y Request
Asegurarse de que la rama de origen sea la de tu fork y la rama de destino sea la del repositorio original (ejemplo: main).
Escribe un título y una descripción clara de tus cambios.
Haz clic en "Create Pull Request".
- 15) Busco el Pull Request que deseo.
Hago clic para revisar los cambios propuestos, reviso lo necesario y cuando quiero aceptarlo aplico el merge.
- 16) Una etiqueta Git es una referencia fija a un punto específico en el historial de commits, se usa para marcar versiones importantes de un proyecto
- 17) El tag se crea a través de **git tag** nombre-del-tag
- 18) Se puede enviar a través de **git push origin** nombre-del-tag
- 19) El historial de Git es el conjunto de todos los commits realizados en un repositorio a lo largo del tiempo
- 20) Se puede ver utilizando **git log**.
- 21) Se puede buscar de varias maneras, por autor (**git log --author="Juan Pérez"**), por palabra clave(**git log --grep="texto a buscar"**), en un archivo específico (**git log -- nombre-del-archivo**)
- 22) Esto reiniciara el historial manteniendo los archivos:

```
rm -rf .git  
git init  
git add .  
git commit -m "Reinicio del historial"
```

- 23) Un repositorio privado en GitHub es un repositorio que solo pueden ver y acceder las personas que el usuario dueño autorice.
- 24) Hay que ir a nuevo -> Repositorio-> llenar los datos solicitados y elegir la opción Private con el candado.
- 25) Vamos a nuestro repositorio privado, luego vamos arriba a settings y seleccionamos Access->Collaborators y en Add people seleccionamos al colaborador deseado