



# PROGRAMACIÓN II

GitHub



**Profesor:** Betancud, Ariel.

**Nombre del grupo:** “Carpinchos Programando”

**Integrantes del Grupo:**

- Aguilar, Melina (10.981).
- Aguilera, Mariana (10.983).
- Atim, Mercedes (10.998).
- Lanatta, Wanda Oriana (11.120).
- Mercado, Nicolás (11.151).
- Molina, Cirano (11.156).
- Ríos, Nelson (11.211).
- Ríos Garín, Ana Paula (11.212).

**Entrega:** Antes de las 23 hs del 25/09

## **ACTIVIDADES**

Error con los tags en git.

### **Investigación:**

Si un tag es imposible generarlo dos veces ¿Cómo es que existe el error de dos tags con el mismo nombre?

¿Cómo se origina este problema o error?

La respuesta debe ser enviada antes de las 23 horas por cada grupo, deben enviar comandos y todo los pasos que harían frente a este conflicto.

## DESARROLLO DE LAS ACTIVIDADES

En Git, un tag es una referencia a un commit específico y se utiliza comúnmente para marcar versiones importantes de un proyecto. En teoría, un tag no puede tener el mismo nombre dos veces en el mismo repositorio. Sin embargo, pueden ocurrir situaciones que generen confusión o errores.

**Tags Locales y Remotos:** Si creamos un tag localmente y luego lo borramos o cambiamos su nombre, pero alguien más en tu equipo ya ha creado un tag con el mismo nombre en el repositorio remoto, al intentar sincronizar, podemos enfrentar un conflicto.

**Fusión de Repositorios:** Si tenemos dos ramas o forks de un proyecto que tiene tags con el mismo nombre, y luego fusionamos estos repositorios, podemos tener problemas con los tags duplicados.

**Errores de Escritura:** A veces, un error humano puede hacer que alguien intente crear un tag que ya existe sin darse cuenta, lo que puede generar mensajes de error.

**Etiquetas Anteriores:** Si borramos un tag y luego lo volvemos a crear, puede parecer que estamos creando un tag nuevo, pero en realidad estamos volviendo a usar un nombre que ya existía.

**Conflictos en repositorios remotos:** Si dos desarrolladores crean un tag con mismo nombre en sus repositorios locales y luego intentan subirlo al repositorio remoto, puede haber un conflicto.

**Forzar la creación de tags:** Usar el comando `git tag -f` puede forzar la creación de un tag con el mismo nombre, sobrescribiendo el existente.

**Problemas de sincronización:** Si el repositorio remoto no está sincronizado correctamente, puede haber inconsistencias que resulten en tags duplicados.

```
MINGW64/c/Users/MyL/Desktop/class-git/class-git
$ git tag
Version1.0
Version2.0
Version3.0
Version4.0

MyL@DESKTOP-06DBG82 MINGW64 ~/Desktop/class-git/class-git (main)
$ git tag -a Version4.0 -m "Creo etiqueta Version4.0"
fatal: tag 'Version4.0' already exists

MyL@DESKTOP-06DBG82 MINGW64 ~/Desktop/class-git/class-git (main)
$ git tag -d Version4.0
Deleted tag 'Version4.0' (was ba0f105)

MyL@DESKTOP-06DBG82 MINGW64 ~/Desktop/class-git/class-git (main)
$ git tag -a Version4.0 -m "Creo etiqueta Version4.0"

MyL@DESKTOP-06DBG82 MINGW64 ~/Desktop/class-git/class-git (main)
$ git push origin Version4.0
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 179 bytes | 179.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:MerAtim/class-git.git
 * [new tag]          Version4.0 -> Version4.0

MyL@DESKTOP-06DBG82 MINGW64 ~/Desktop/class-git/class-git (main)
$ git fetch --tags

MyL@DESKTOP-06DBG82 MINGW64 ~/Desktop/class-git/class-git (main)
$ git tag | grep Version4.0
Version4.0
```

**Verificamos los tags existentes**

**Creamos una etiqueta**

**Borramos el tag existente**

**Despues de haber creado el tag subimos el nuevo al remoto.**

**Traemos los tags del remoto**

**grep <nombre del tag> nos permite verificar si existe un tag específico en el repositorio.**

Resolver el conflicto:

A. Si los tags apuntan a diferentes commits

Decide cuál de los tags es el correcto y elimina el duplicado:

**git tag -d <tag-name>**

Luego, si es necesario, vuelve a crear el tag apuntando al commit correcto:

**git tag <tag-name> <commit-hash>**

**B. Si los tags apuntan al mismo commit**

Simplemente elimina uno de los tags duplicados:

```
git tag -d <tag-name>
```

**5. Actualizar el repositorio remoto**

Después de eliminar los tags duplicados localmente, actualiza el repositorio remoto:

```
git push origin :refs/tags/<tag-name>
```

```

MINGW64/c/Users/MyL/Desktop/class-git/class-git
MyL@DESKTOP-O6DBG82 MINGW64 ~/Desktop/class-git/class-git (main)
$ git push origin :refs/tags/Version4.0
To github.com:MerAtim/class-git.git
- [deleted]          Version4.0

MyL@DESKTOP-O6DBG82 MINGW64 ~/Desktop/class-git/class-git (main)
$ git log --oneline
0155d8d (HEAD -> main, tag: Version4.0, origin/main) Agrego clase 6 de git
42fb750 (tag: Version2.0) Clase 5 de Git
e38b2ad Update SegundoSemestre.md Clase 5
7a4d430 Update SegundoSemestre.md Clase 4
7885bd5 (tag: Version3.0) Merge branch 'main' of github.com:MerAtim/class-git
61ece61 Actualizo cambios del remoto al local
b5011fd Update SegundoSemestre.md
68536d6 Update SegundoSemestre.md
d434ea0 Update SegundoSemestre.md
5849dfa Update SegundoSemestre.md
b973482 Agrego Archivo MD
e9687b7 Update README.md
4f6f7f9 (tag: Version1.0, origin/segunda, segunda) Merge branch 'segunda'
06dbbd1 Agregue suscripción, cambie el código y puse todo azul en css
6ecb44c Modifique el archivo portafolio
ec49f10 Para guardar estos cambios en el README.md
9cff5d9 Finalizado el cambio en rama segunda
a183464 Añado cambios del portafolio en html
1802bfb Guardo archivos creados log.txt
cdca8e1 modifico archivo html
27af9e5 Agrego comandos en README de la clase 7 de Git
acf4003 Agrego comandos en README de la clase 7 de Git
d41daa3 Agrego comandos al archivo README
36409ef cambio en la última línea del historia.txt
9e8d31b Creamos el css para darle algo de estilo a nuestro portafolio
0f6f93c Agregamos el html para nuestro portafolio
2684c15 Agregando comandos en README de la clase 4
04e8bc4 Agregando lineas al archivo historia.txt
db74cf5 Agregando el archivo historia.txt
bfcbedc Modifico el README dentro del directorio class-git
3c68393 Cargamos el README dentro del directorio class-git
0fa82f3 Mi segundo commit
910720e Agregando modificaciones del proyecto

MyL@DESKTOP-O6DBG82 MINGW64 ~/Desktop/class-git/class-git (main)
$ git push origin Version4.0
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 179 bytes | 179.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:MerAtim/class-git.git
* [new tag]          Version4.0 -> Version4.0

MyL@DESKTOP-O6DBG82 MINGW64 ~/Desktop/class-git/class-git (main)
$ git log --oneline --decorate
0155d8d (HEAD -> main, tag: Version4.0, origin/main)

```

**Elimina el tag del repositorio Remoto**

**Agregamos nuevamente el tag al remoto apuntando al commit específico.**

**Ya figura en el remoto apuntando al commit.**

**Prevenir futuros duplicados:**

Para evitar que este problema ocurra nuevamente, considera implementar las siguientes medidas:

- **Revisar los procesos de creación de tags:** Asegúrate de que todos los desarrolladores sigan un proceso claro y documentado para crear tags.
- **Automatización y validación:** Implementa scripts de validación en tu pipeline de CI/CD para verificar la unicidad de los tags antes de crearlos.
- **Educación y comunicación:** Asegúrate de que todos los miembros del equipo entiendan la importancia de mantener la unicidad de los tags y cómo verificarlo.