

GUÍA PARA TRABAJAR CON GIT Y GITHUB DESDE MÚLTIPLES PC

1. INTRODUCCIÓN

Esta guía explica cómo trabajar correctamente en el mismo repositorio de GitHub desde varias computadoras, evitando conflictos y errores de permisos. Incluye procedimientos para repositorios propios, colaboraciones y reglas de sincronización.

2. REGLA DE ORO AL TRABAJAR EN VARIAS PC

Siempre seguir esta secuencia:

ANTES de trabajar:

```
git pull
```

DESPUÉS de trabajar:

```
git add .
```

```
git commit -m "Cambios"
```

```
git push
```

Esto evita conflictos y asegura que cada PC tenga la última versión.

3. TRABAJAR DESDE OTRA PC SI EL REPOSITORIO ES TUYO

Pasos:

1. Clonar el repositorio:

```
git clone https://github.com/usuario/repositorio.git
```

2. Configurar tu usuario (solo la primera vez):

```
git config --global user.name "TuNombre"
```

```
git config --global user.email "tu-correo@example.com"
```

3. Trabajar normalmente:

```
git pull
```

```
git add .  
git commit -m "Cambios"  
git push
```

4. SI EL REPOSITORIO ES TUYO PERO OTRA PERSONA QUIERE TRABAJAR

Esa persona NO puede hacer push hasta que la agregues como colaborador.

Pasos:

- Ir a GitHub Settings Collaborators
- Agregar su usuario
- La otra persona acepta la invitación
- Ahora podrá hacer:

```
git pull  
git push  
git commit
```

5. SI EL REPOSITORIO NO ES TUYO

Solo puedes hacer:

```
git clone  
git pull
```

No podrás hacer push.

Solución:

Crear un fork y trabajar en tu copia.

1. Click en "Fork" en GitHub

2. Clonar tu fork:

```
git clone https://github.com/tuUsuario/tuFork.git
```

3. Trabajar y hacer push normalmente

Para enviar cambios al proyecto original:

Crear un Pull Request (PR).

6. CÓMO ACTUALIZAR TU FORK

Si creaste un fork y quieres traer los cambios del proyecto original:

Configurar el repositorio remoto original (solo una vez):

```
git remote add upstream https://github.com/original/Proyecto.git
```

Actualizar tu fork:

```
git pull upstream main
```

Y luego actualizar tu GitHub:

```
git push origin main
```

7. RESOLVER CONFLICTOS ENTRE PC

Si en una PC hiciste cambios que chocan con los de otra, aparecerán conflictos.

Debes editar los archivos que muestran:

<<<<< HEAD

Tu versión

=====

Versión remota

>>>>>

Después:

```
git add .
```

```
git commit -m "Resolver conflictos"
```

```
git push
```

8. RESUMEN FINAL

- Usa siempre pull antes de tocar archivos
- Haz commit + push después de trabajar

- Si es tu repo, puedes usarlo en varias PCs sin límite
- Si otra persona colaborará, debes agregarla como colaborador
- Si el repo no es tuyo, haz un fork

FIN.