

RAMAS

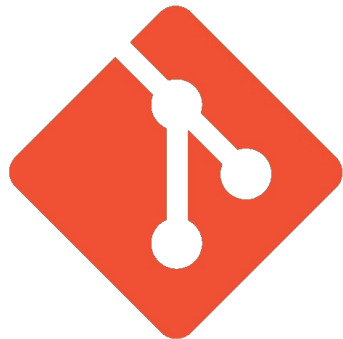


Uso de ramas en git

En proyectos de software con planificación ágil es común tener múltiples líneas de trabajo al mismo tiempo en las que diferentes miembros del equipo trabajan en diferentes desarrollos e integran los cambios de forma gradual.

Para facilitar esta dinámica de trabajo en git existe el concepto de **ramas** o **branches**.

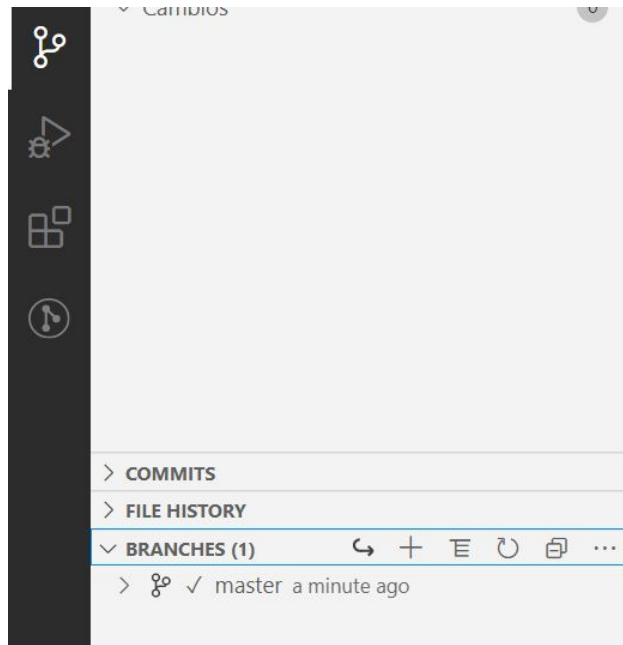
Una rama es un flujo de trabajo copia del flujo de trabajo original. En este nuevo flujo un miembro introduce nuevos desarrollos que una vez terminados y verificada su calidad se integran con el flujo de trabajo original. De esta forma se evita mezclar el código principal con cambios que forman parte de una tarea grande sin terminar, pudiendo desestabilizar el flujo de trabajo principal ocasionando problemas a otros miembros del equipo.



Uso de ramas en git

La extensión **GitLens** de Visual Studio Code nos permite una manera muy sencilla e intuitiva de crear y trabajar con ramas.

Desde la pestaña de control de código fuente encontramos el panel Branches donde podemos ver las ramas del proyecto y crear nuevas.



Uso de ramas en git

Una vez creada más de una rama podemos trabajar en cada una de manera independiente.

Una vez terminado el trabajo de una de las ramas se puede fusionar sobre la rama principal (comúnmente master o main).

> **COMMITTS**

> **FILE HISTORY**

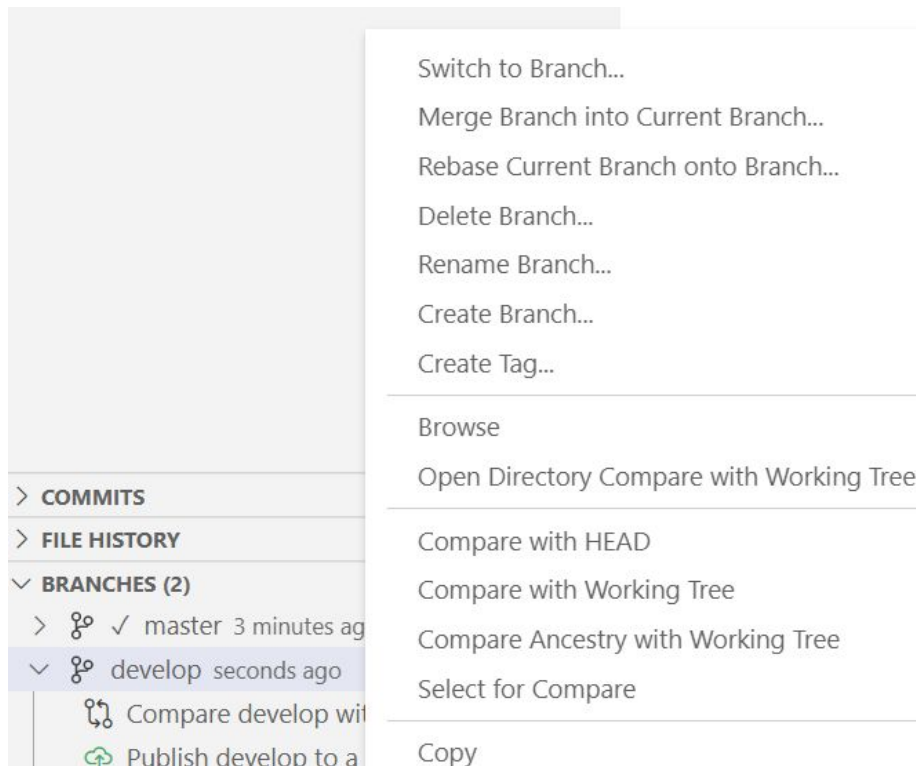
✓ **BRANCHES (2)**

>  master 2 minutes ago

>  ✓ develop 2 minutes ago

Uso de ramas en git

Con **merge** fusionamos el trabajo de una rama en otra y con **delete** podemos eliminar esa rama.



Uso de ramas en git



En Git Bash podemos trabajar con ramas haciendo uso de los comandos:

- **Ver las ramas:** `git branch -a`
- **Crear una nueva rama:** `git branch testing`
- **Cambiar a una rama:** `git checkout testing`
- **Crear y cambiar a una rama:** `git checkout -b testing`
- **Fusionar una rama:** `git merge testing` (primero git checkout a la rama donde queremos fusionar)
- **Borrar una rama:** `git branch -d testing`