

Conflictos y cómo resolverlos

IT BOARDING

BOOTCAMP



// ¿Qué es un Conflicto?

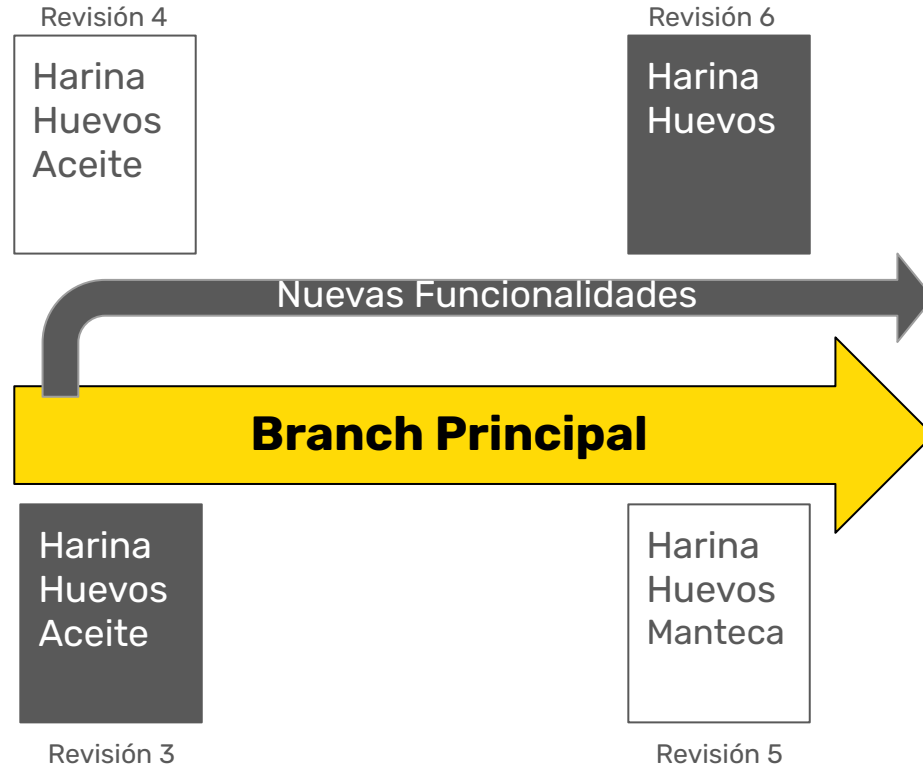
- Sea que usemos la consola de comandos o un GUI, el error más frecuente es cuando dos o más usuarios tocan el mismo archivo en la misma línea de código.
- Para evitar esto, la recomendación más eficiente es dividir bien el trabajo y no tocar al mismo tiempo con varias personas el mismo archivo.

// Resolviendo Conflictos

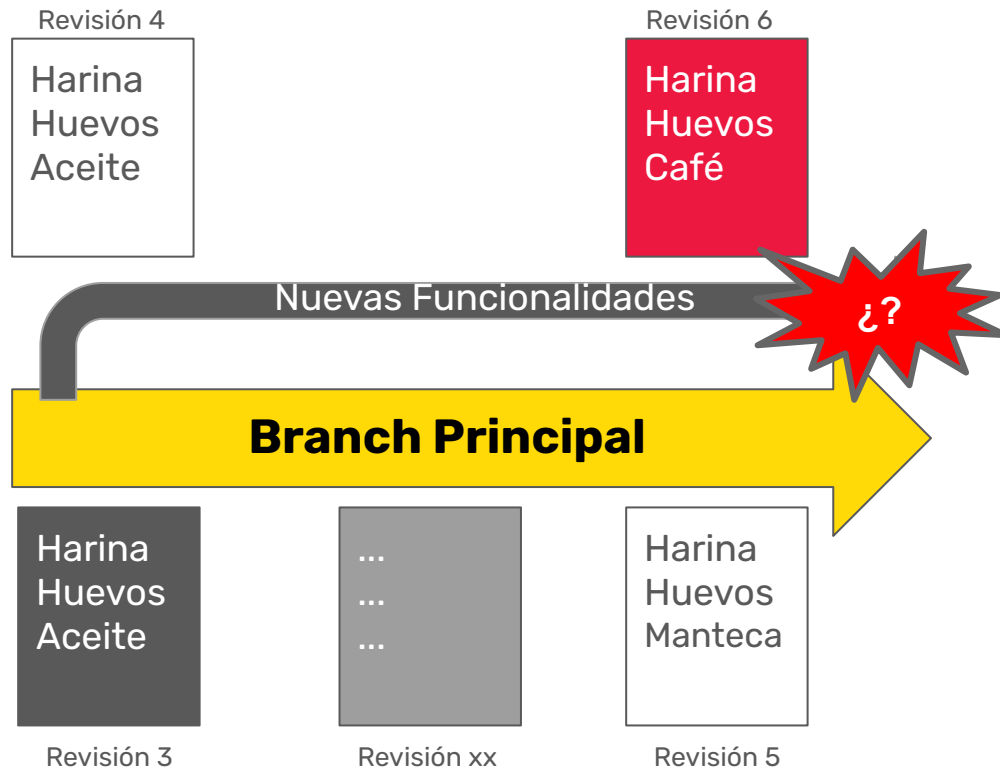
Cuando estamos trabajando con varios colaboradores es posible que tengamos conflictos cuando actualicemos o bajemos los archivos que tengamos en el repositorio remoto a nuestro repositorio local. Esto ocurre por lo general cuando se tocan uno o varios archivos en diferentes commits.

- Git a menudo puede resolver diferencias entre ramas y fusionarlas automáticamente. Por lo general, los cambios se realizan en diferentes líneas, o incluso en diferentes archivos, lo que hace que la combinación sea fácil de entender para Git.
- A veces hay cambios en competencia que Git no puede resolver sin nuestra ayuda.

// Ejemplo de Conflictos



// Ejemplo de Conflictos



// Resolviendo conflictos

Si el conflicto es causado por cambios en la misma línea del mismo archivo los mismos se pueden resolver utilizando el editor de código y decidiendo cuáles cambios se quedan y cuales no. Por ejemplo tenemos conflictos en un archivo.txt

> _

```
> git status
# On branch branch-b
# You have unmerged paths.
#   (fix conflicts and run "git commit")
#
# Unmerged paths:
#   (use "git add ..." to mark resolution)
#
# both modified:      archivo.txt
#
no changes added to commit (use "git add" and/or
"git commit -a")
```

// Resolviendo conflictos

Debemos en el editor de código buscar el marcador de conflicto <<<<<<

Los cambios de la rama HEAD o base aparecerán después de la línea <<<<<< HEAD. A continuación, veremos =====, que divide nuestros cambios de los cambios en la otra rama, seguido de >>>>>> BRANCH-NAME. Lo que resta es decidir qué se queda y qué debemos sacar.

Luego agregar los cambios al stage `git add .`, los confirmamos `git commit -m "solucionando conflictos"` y por último los subimos `git push origin main`

Este es un ejemplo de conflictos en un archivo

```
{ <<<<<< HEAD
  cambios desde otro usuario
  =====
  mis cambios
  >>>>>> branch-a
```



Gracias.

IT BOARDING

BOOTCAMP

