



1º DAM/DAW Sistemas Informáticos

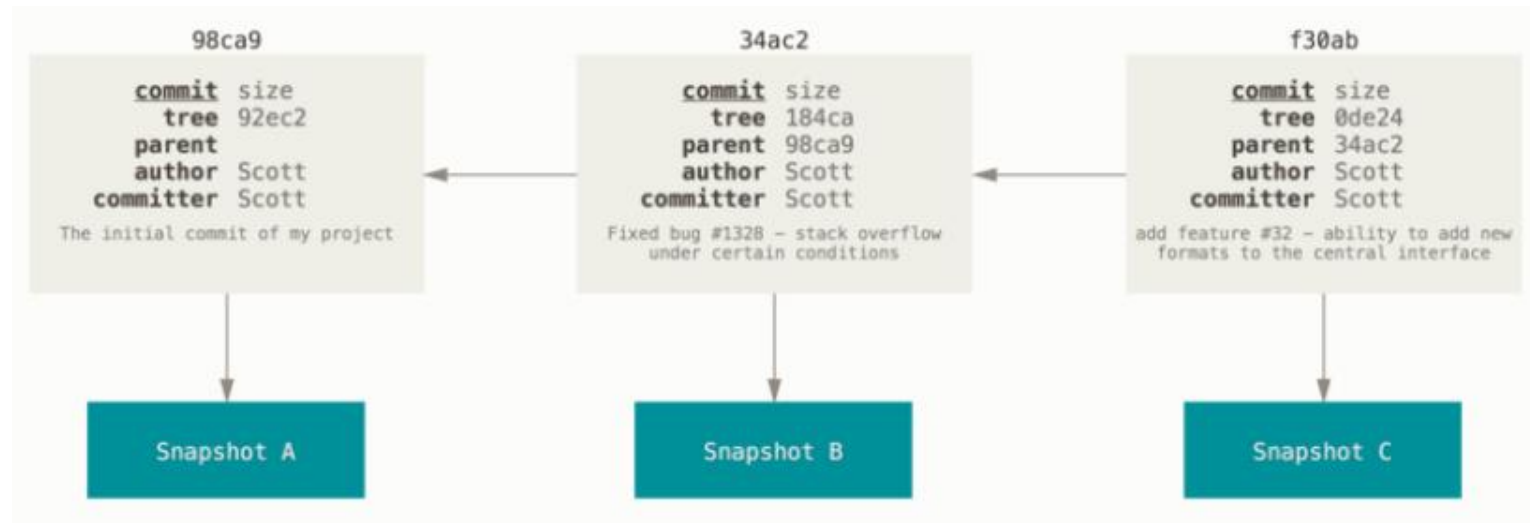
U3. Sistemas de control de versiones - Git

3 - Git. Ramas



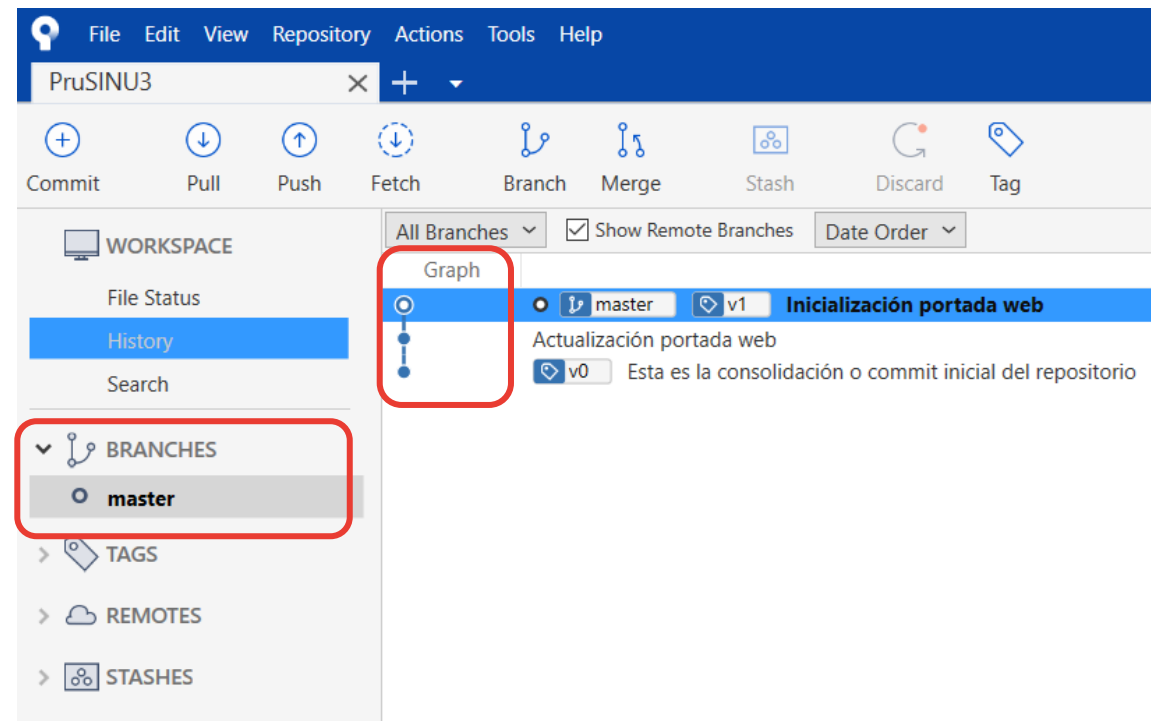
Git. ¿Qué es una Rama?

- Su traducción en inglés es **branch**.
- **Cada** vez que realizamos una **consolidación** o commit, **Git guarda una instantánea** de la situación completa del repositorio (**snapshot**).
- Y una **referencia** a la **consolidación antecesora** (**parent**).



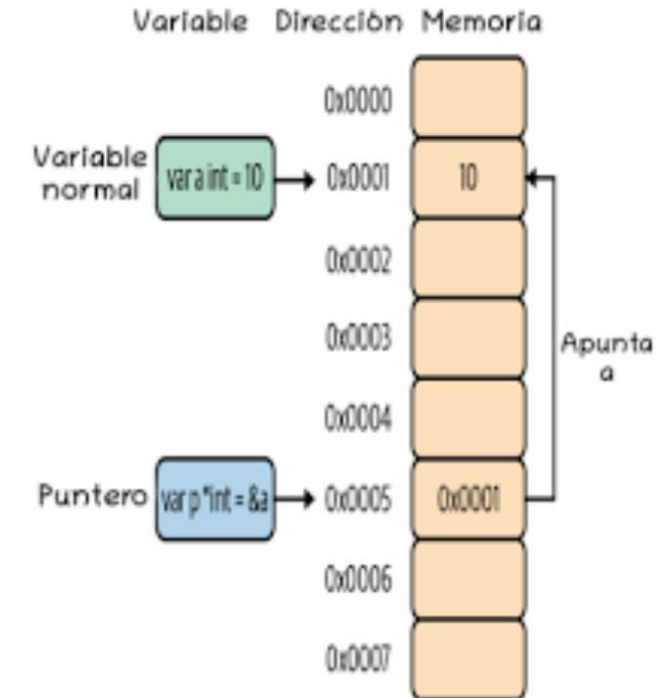
Git. ¿Qué es una Rama?

- A nivel gráfico, SourceTree va generando un **grafo**, con la representación de las ramas. Hasta el momento, hemos trabajado en una única rama (master).



Git. ¿Qué es una Rama?

- Una **rama Git** es un **puntero que apunta** en su creación a **una consolidación** o commit y permite disponer de un **camino paralelo** para hacer evolucionar un proyecto, a partir de dicha consolidación.
- **Nota:** un **puntero** es un tipo de variable que **apunta a otra variable**. En este caso, nos sirve para tener identificada y direccionada una consolidación o commit.



Git. ¿Qué es una Rama?

- La **rama por defecto** de Git es la rama “**master**”. Con la primera confirmación de cambios que realicemos en un repositorio, se creará esta rama principal, “master”.
- Podemos crear tantas ramas como necesitemos. Es recomendable realizar las modificaciones de los proyectos en ramas distintas a la principal.



Git. Ramas. Ejemplo de uso 1

- Vamos a suponer que vamos a gestionar un proyecto de desarrollo compuesto por **4 equipos de trabajo**.
- Cada equipo, debe trabajar en un entorno diferente para que no haya problemas de interferencias entre lo que desarrollan unos y otros, para ello se generarán **4 ramas en total**.
- A nivel de coordinación general del proyecto, alguien se debe encargar de controlar y gestionar cómo debe llevarse a cabo la integración de las diferentes ramas o líneas de trabajo:

Git. Ramas. Ejemplo de uso 1

- **FrontEnd:** desarrollan la capa de interacción con el usuario (interfaces).
- **BackEnd:** desarrollan la lógica de procesos internos del funcionamiento de la aplicación.
- **Testing:** se encarga de realizar las pruebas, antes de dar un desarrollo por finalizado.
- **Master:** coordina la evolución paralela del resto de ramas y la posterior integración.

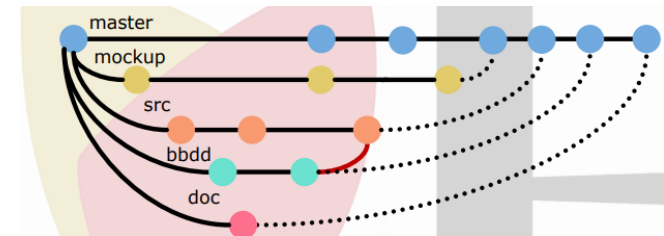


Git. Ramas. Ejemplo de uso 2

- Vamos a suponer ahora que vamos a gestionar un proyecto de desarrollo y **decidimos estructurar el proyecto en 5 secciones.**
- Cada sección, se debe avanzar en un entorno independiente para que no haya problemas de interferencias entre lo que se desarrolla para una sección y para otra.
- **Se generará una estructura de 5 ramas en total:**

Git. Ramas. Ejemplo de uso 2

- **Source:** rama para gestionar el código del proyecto.
- **Mockup:** almacena los elementos y artefactos, fruto del análisis.
- **BBDD:** utilizada para ubicar los ficheros y elementos de la base de datos.
- **Docs:** almacena cualquier creación que no aplique en otra rama. Miscelánea.
- **Master:** coordina la evolución paralela del resto de ramas y la posterior integración.

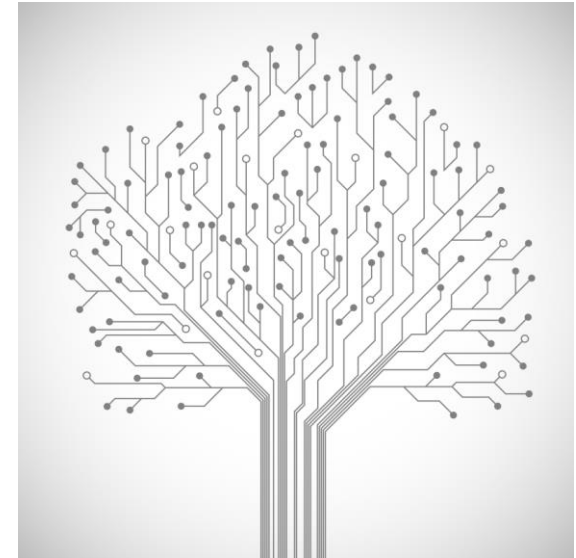


Git. Ramas. Ejemplo de uso 3

- Vamos a suponer ahora que vamos a gestionar el desarrollo de una nueva versión sobre un producto ya existente en el mercado.
- Consistirá en **realizar una serie de mejoras y correcciones**. Por ejemplo, incrementar el rendimiento de las búsquedas, corregir un problema en la entrada de datos, actualizar el diseño de los informes.
- Se generará la siguiente estructura de repositorio:

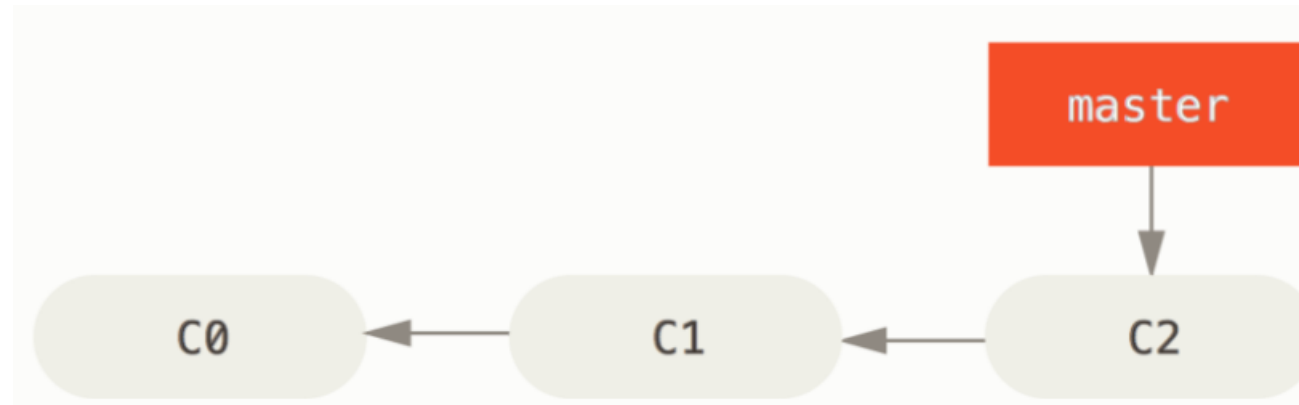
Git. Ramas. Ejemplo de uso 3

- **Rendimiento:** rama para gestionar las modificaciones respecto al rendimiento de las búsquedas.
- **Error:** rama para solucionar el problema detectado.
- **Diseño:** rama que contendrá el nuevo diseño de los informes.
- **Master:** coordina la evolución paralela del resto de ramas y la posterior integración.



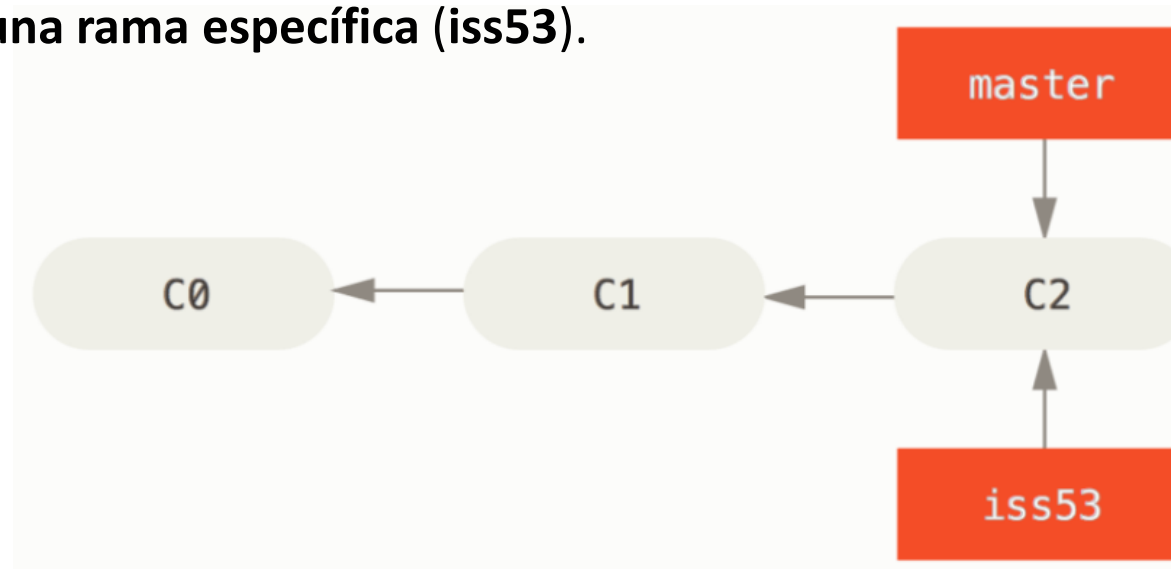
Git. Ramas. Ejemplo de uso 4 (extendido)

- Vamos a suponer ahora que estamos trabajando como desarrolladores/as de un proyecto de **actualización de un sitio web**.
- Lo haremos desde un **punto de vista individual**, en 1ª persona.
- Cuando nos asignan al proyecto vemos que **el repositorio ya dispone de una rama Master**.



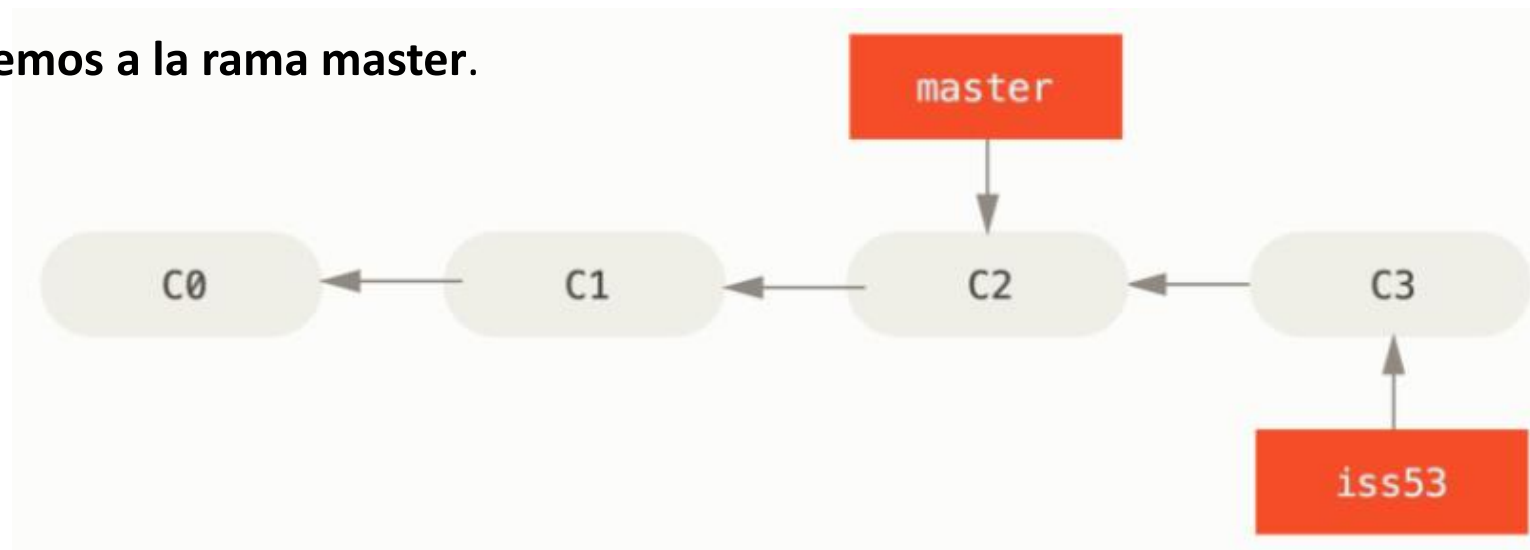
Git. Ramas. Ejemplo de uso 4 (extendido)

- **Nos asignan una tarea de desarrollo**, por ejemplo, la nº 53 (**issue 53**), que consistirá en aplicar una serie de cambios. Por ejemplo, podría ser remodelar el mecanismo para realizar un pedido web, o actualizar el buscador de un catálogo web, etc.
- Así que **creamos una rama específica (iss53)**.



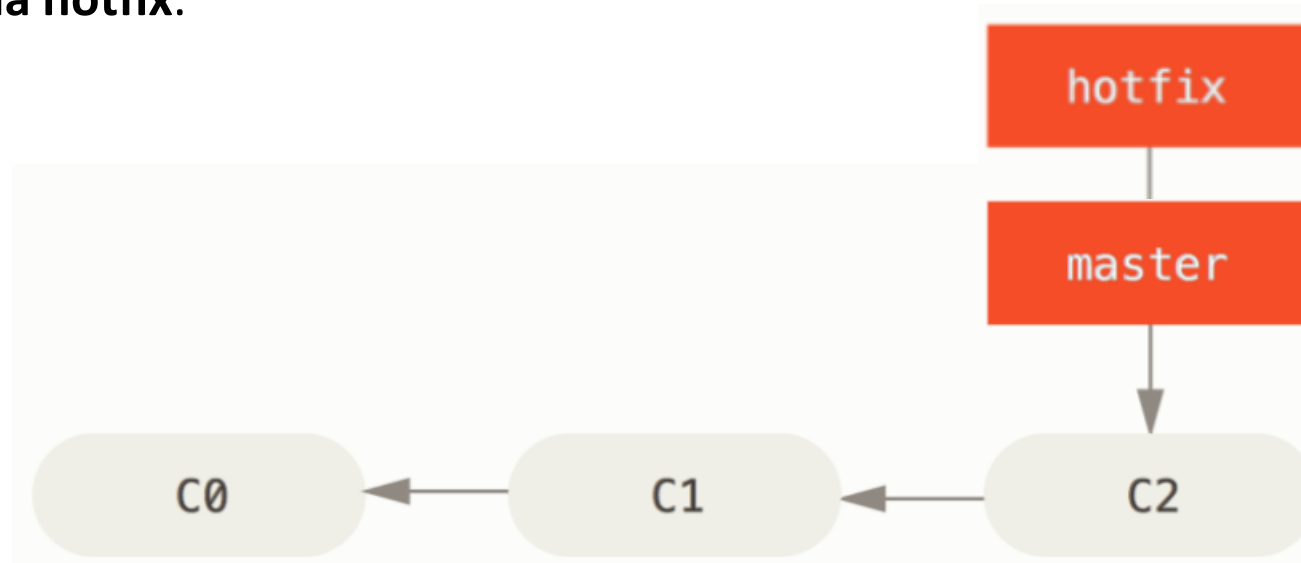
Git. Ramas. Ejemplo de uso 4 (extendido)

- Vamos avanzando en la ejecución de nuestra tarea nº 53 y **durante el desarrollo**, recibimos una **asignación urgente**, debemos **estudiar un error inmediatamente**, para solucionarlo lo antes posible.
- **Consolidaremos** los cambios que llevamos entre manos en la **rama iss53**.
- **Cambiaremos a la rama master.**



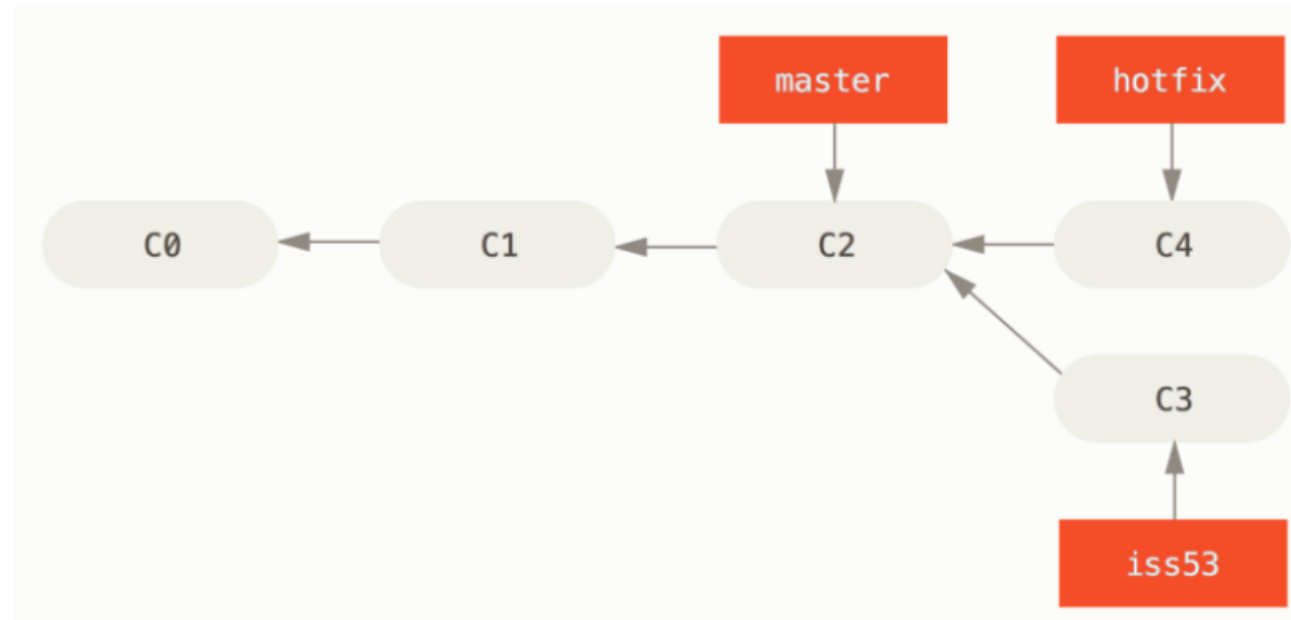
Git. Ramas. Ejemplo de uso 4 (extendido)

- En este punto, el **directorio de trabajo** del proyecto es exactamente **como estaba antes** de comenzar a trabajar en la tarea nº 53.
- Antes de comenzar a trabajar en la revisión de error, **creamos una nueva rama denominada hotfix**.



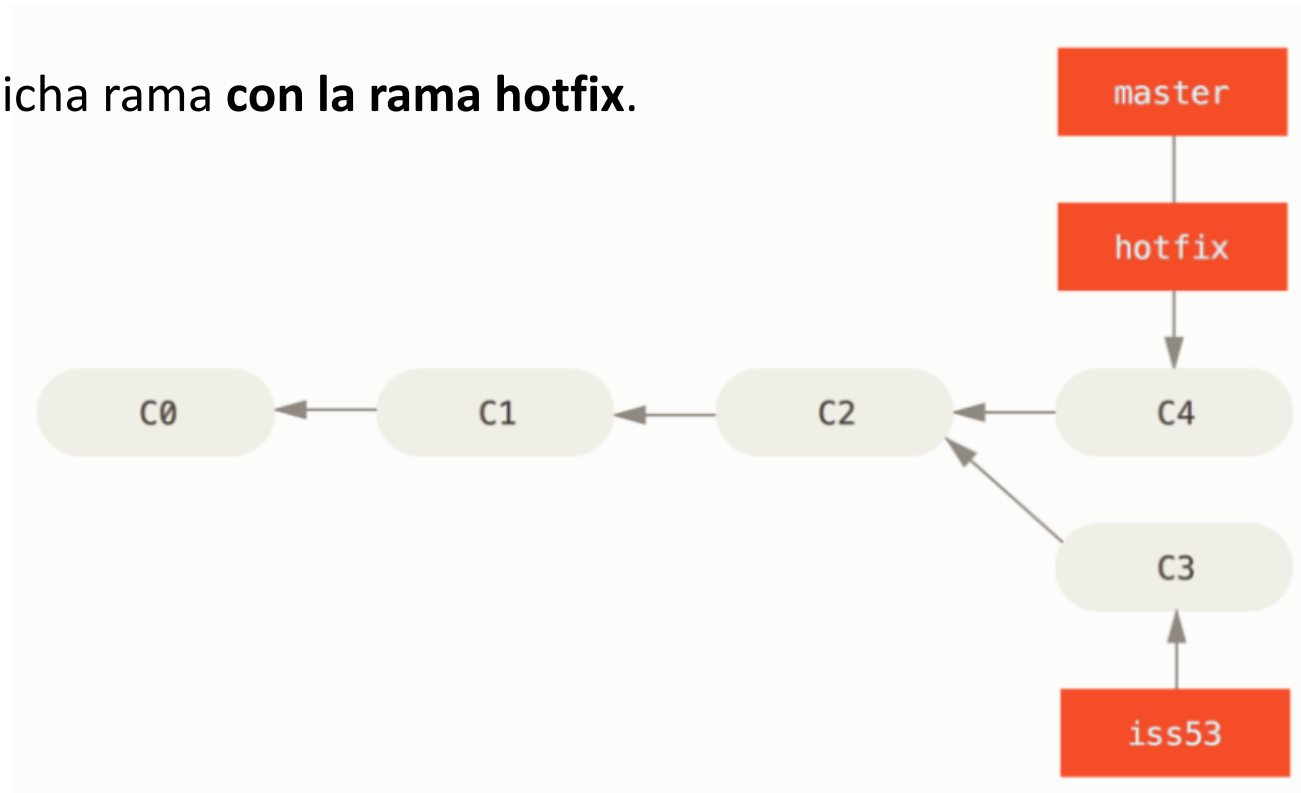
Git. Ramas. Ejemplo de uso 4 (extendido)

- Una vez realizadas las pruebas correspondientes, **localizamos el error y lo corregimos** mediante las modificaciones que correspondan.
- Una vez esté resuelto el error, **confirmamos los cambios en una nueva consolidación** de la rama hotfix.



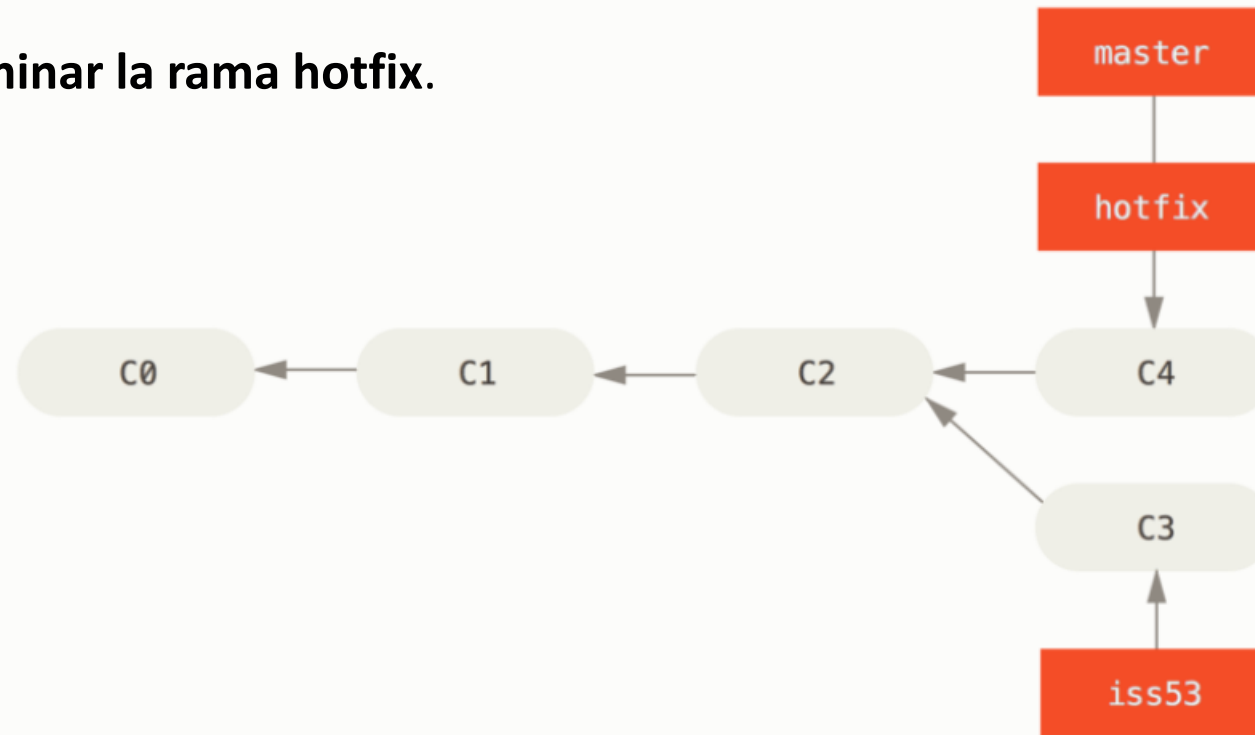
Git. Ramas. Ejemplo de uso 4 (extendido)

- El **error** está **resuelto** en la rama **hotfix**.
- **Volvemos a** la rama **master**.
- **Fusionamos** dicha rama **con la rama hotfix**.



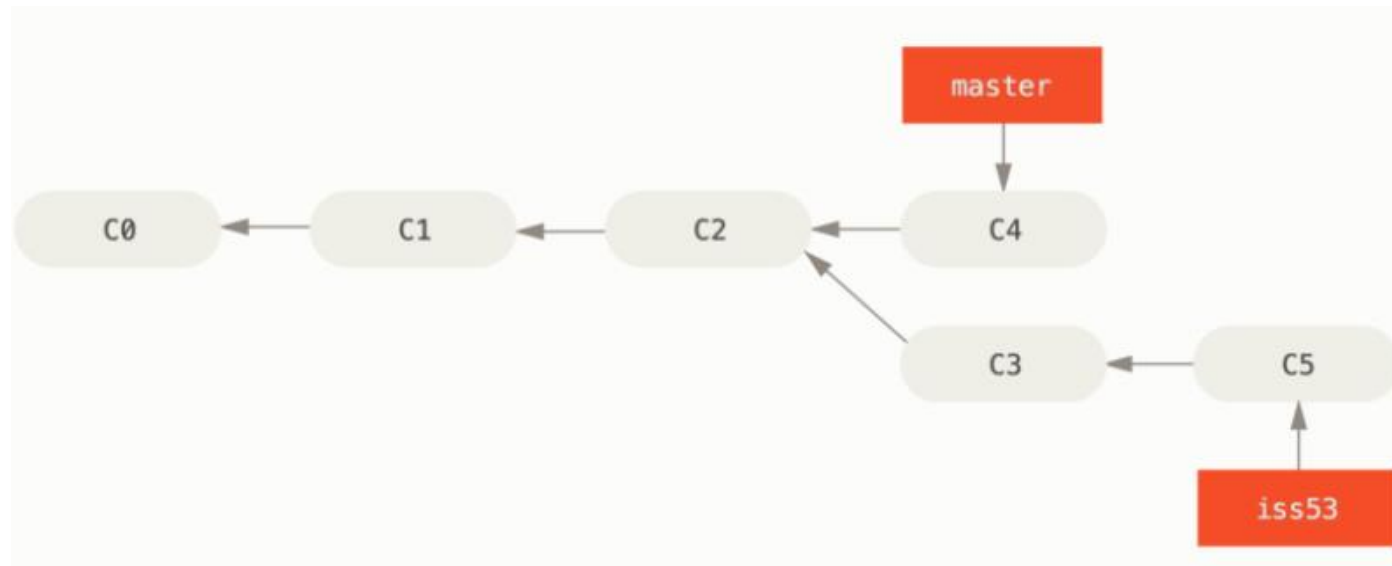
Git. Ramas. Ejemplo de uso 4 (extendido)

- Básicamente, **el puntero master avanza**.
- Ambas ramas quedan en la misma situación evolutiva.
- Podríamos **eliminar la rama hotfix**.



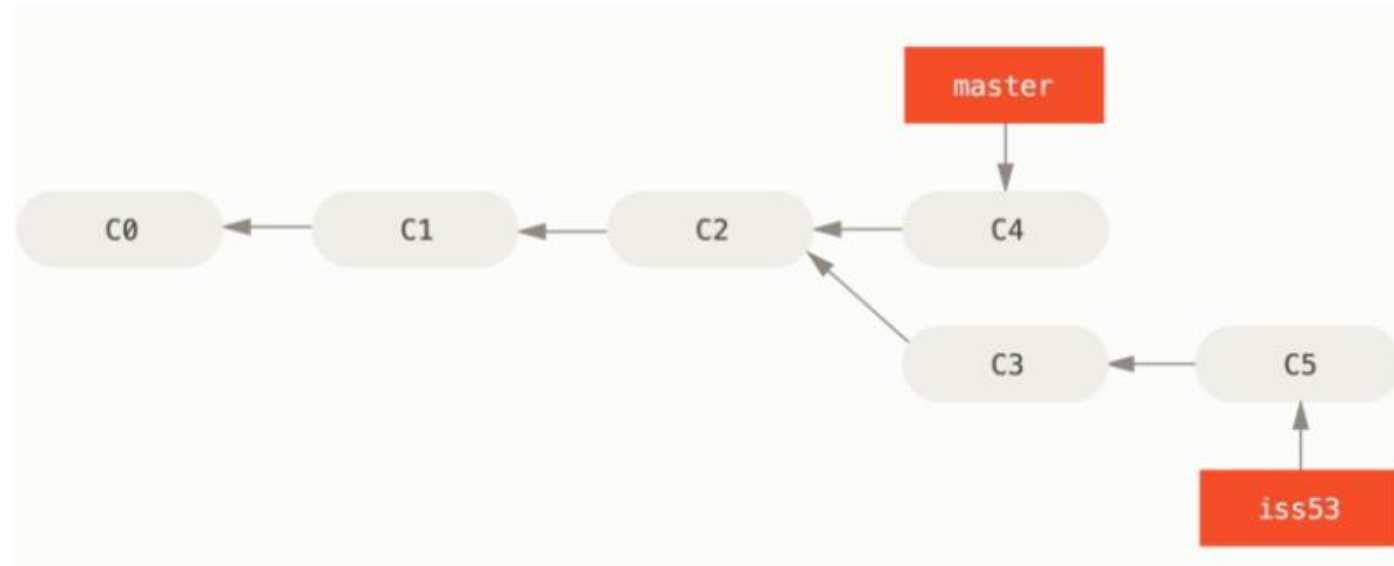
Git. Ramas. Ejemplo de uso 4 (extendido)

- **Volvemos a la rama iss53** para continuar con la tarea nº 53 que habíamos dejado aparcada.
- Una vez **terminamos** las cuestiones de **la tarea nº 53, consolidamos los cambios** (es importante tener en cuenta que **la solución del error no está aplicada en la rama iss53**).



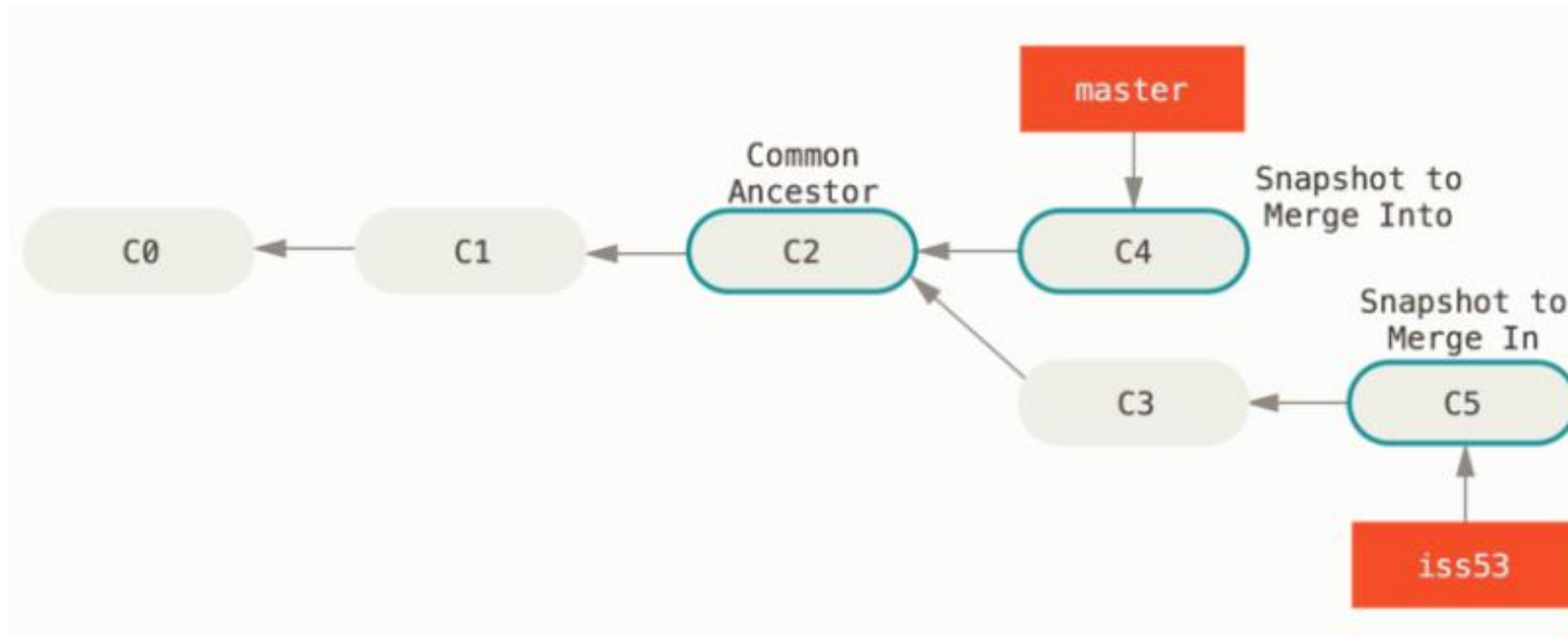
Git. Ramas. Ejemplo de uso 4 (extendido)

- Es el momento de **fusionar los cambios realizados en la rama iss53 con la rama master**.
- **Cambiamos a la rama master**, que es donde queremos realizar la fusión.



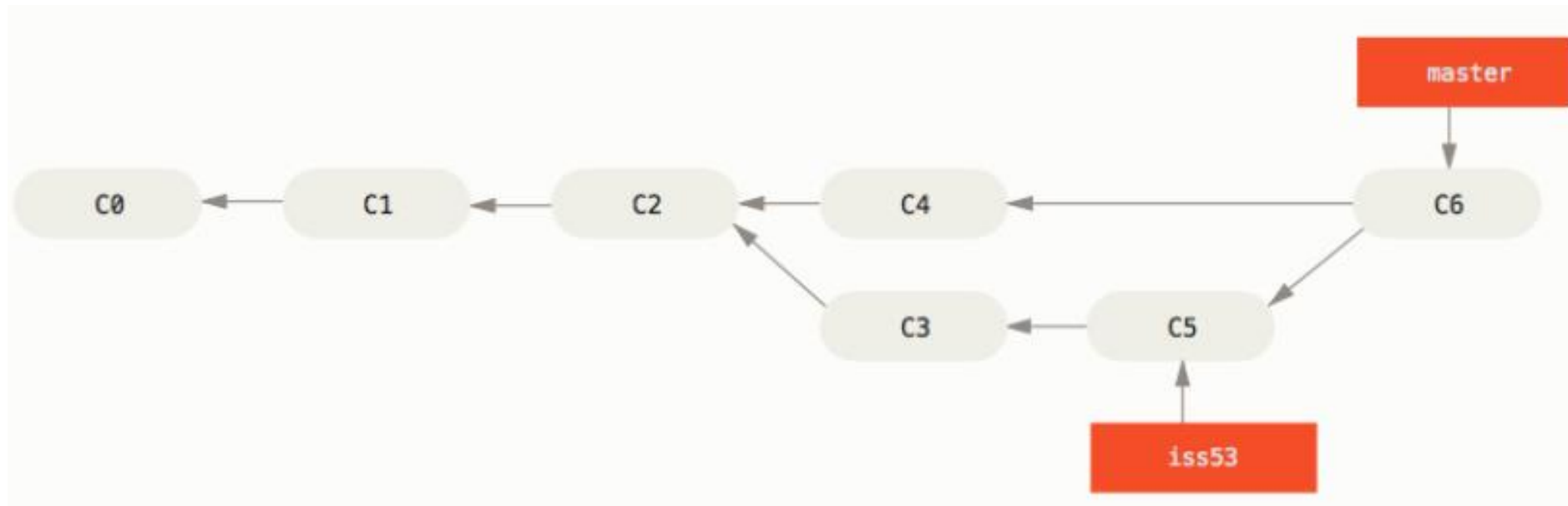
Git. Ramas. Ejemplo de uso 4 (extendido)

- **El contenido de ambas ramas no es lineal, es divergente.** Es decir, ambas ramas han evolucionado de manera distinta desde la consolidación **C2**, su **ancestro común**.



Git. Ramas. Ejemplo de uso 4 (extendido)

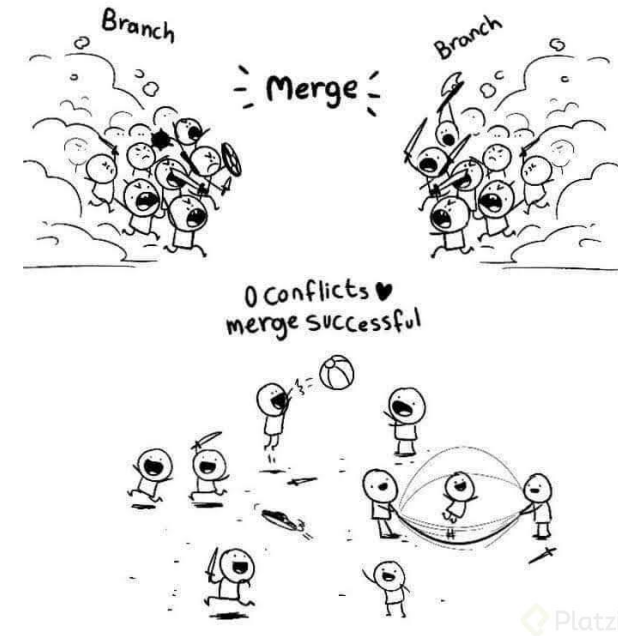
- Este tipo de fusión requiere que Git haga su “magia”, de modo que se creará una **nueva consolidación** que **fusionará la última consolidación de las ramas iss53 y master**.
- El **resultado** es una **consolidación con 2 padres (C6)**.



Git. Ramas. Ejemplo de uso 4 (extendido)

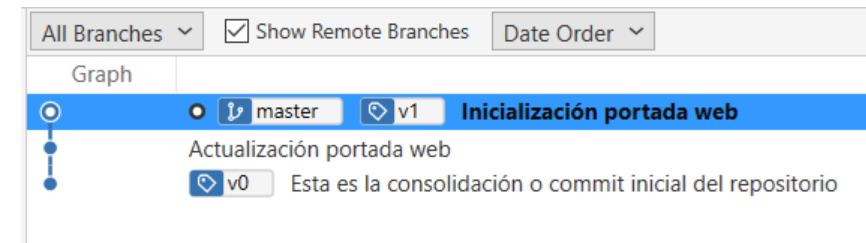
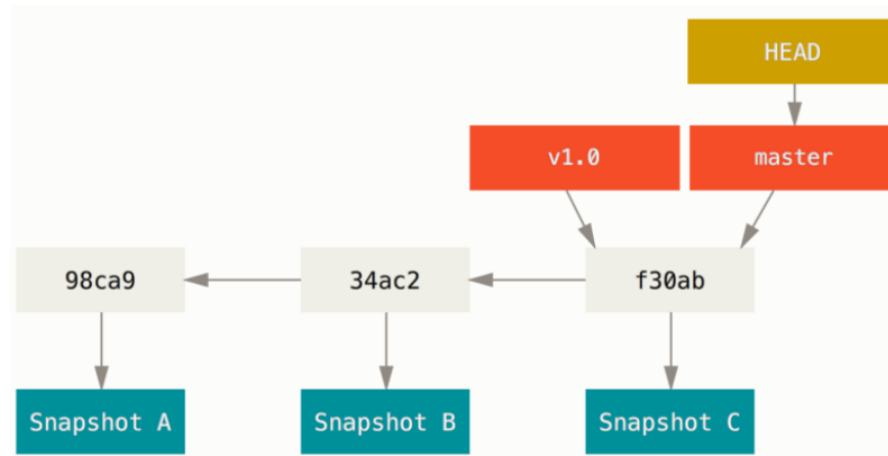
- Si en **ambas ramas**, iss53 y master, se hubiera **modificado el mismo elemento** (la misma línea y el mismo fichero), este tipo de fusión hubiera generado un **conflicto** que debemos resolver para que la fusión sea funcionalmente correcta.

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Nuevo Titulo</title>
  </head>
  <body>
    <p>Contenido de la web</p>
    <p>Nuevo párrafo de la página</p>
  </body>
</html>
```



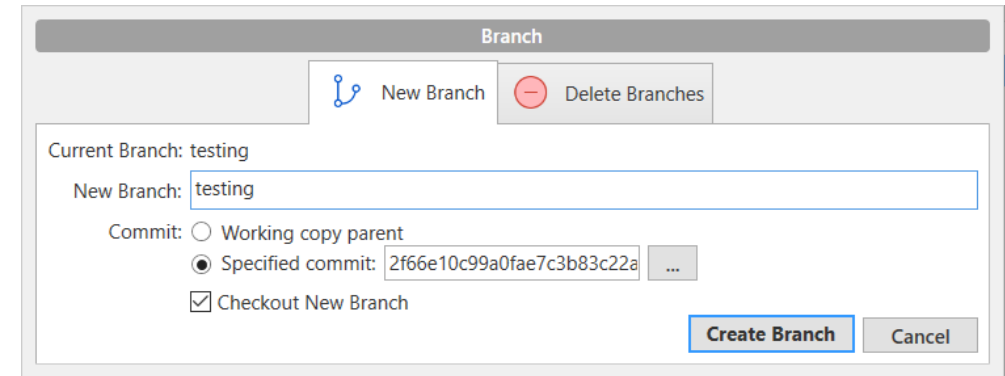
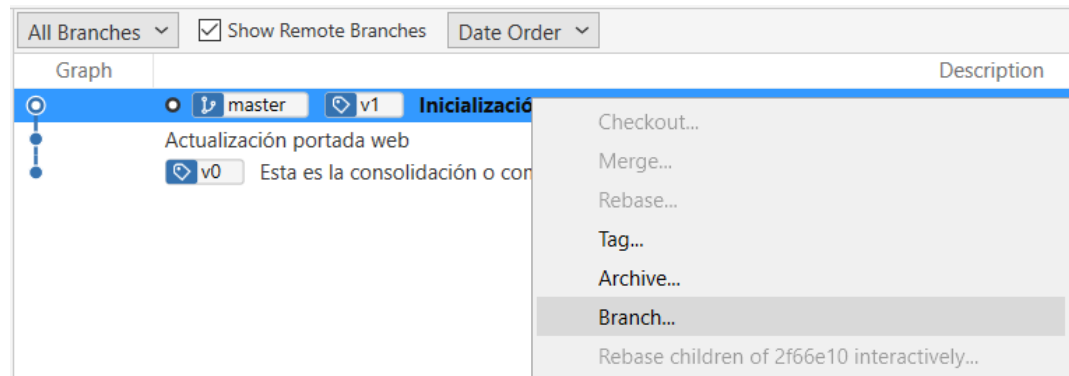
Git. Nuevas ramas.

- Cuando inicias un repositorio se crea el puntero “master”, la rama por defecto de Git. Como mínimo siempre existirá una rama.
- Puedes crear etiquetas (punteros), para localizar una consolidación fácilmente. Se pueden usar para indicar números de versión.
- El **puntero especial HEAD**, siempre apunta a la **rama activa** en cada momento.

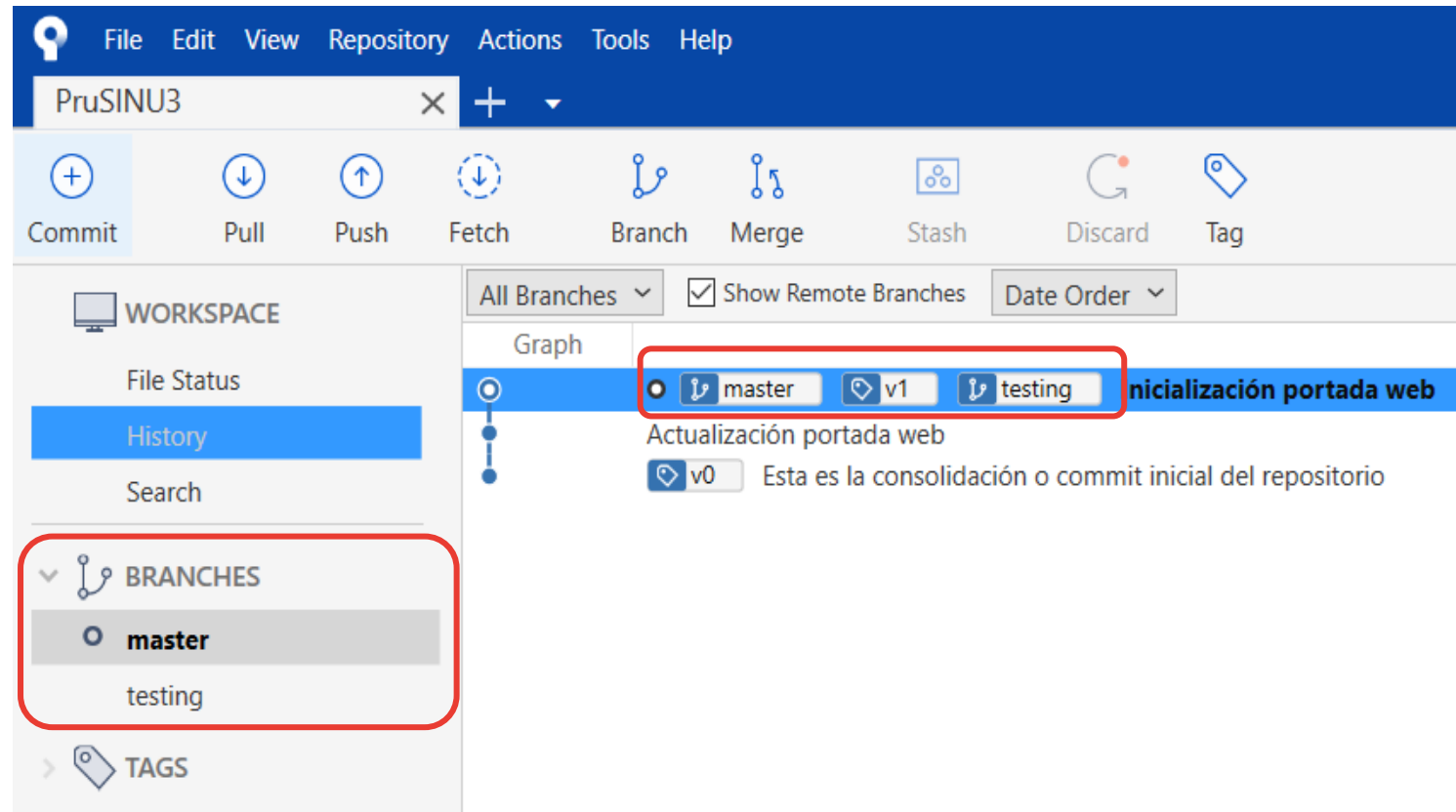


Git. Nuevas ramas.

- Cuando creas una **rama nueva**, se crea un **puntero**, que apunta inicialmente a la consolidación que hayamos seleccionado.
- Si desde SourceTree, pulsamos botón derecho sobre una consolidación, podemos generar una **nueva rama** (branch). Le pondremos un nombre y la crearemos.



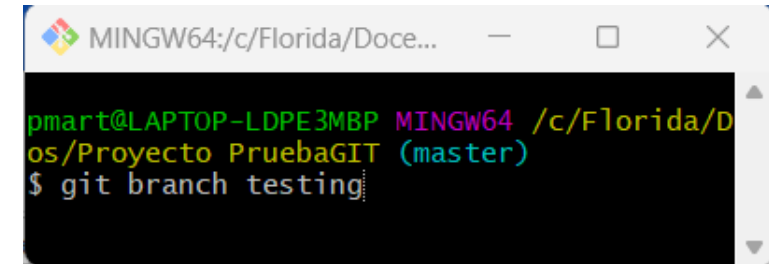
Git. Nuevas ramas.



Git. Nuevas ramas.

- También podemos crear nuevas ramas en la consolidación activa (HEAD) desde consola:

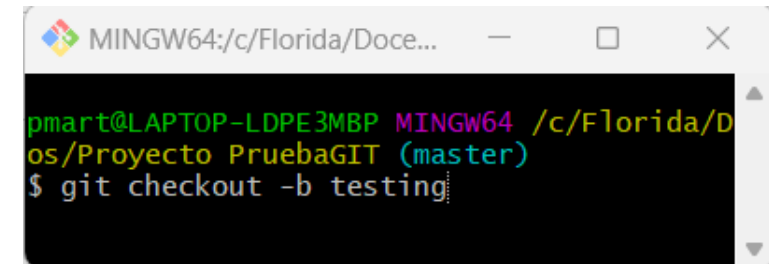
\$ git branch testing



```
mingw64/c/Florida/Doce...  
pmart@LAPTOP-LDPE3MBP MINGW64 /c/Florida/D  
os/Proyecto PruebaGIT (master)  
$ git branch testing
```

- Con este comando, creamos una nueva rama y, además, cambiamos a ella:

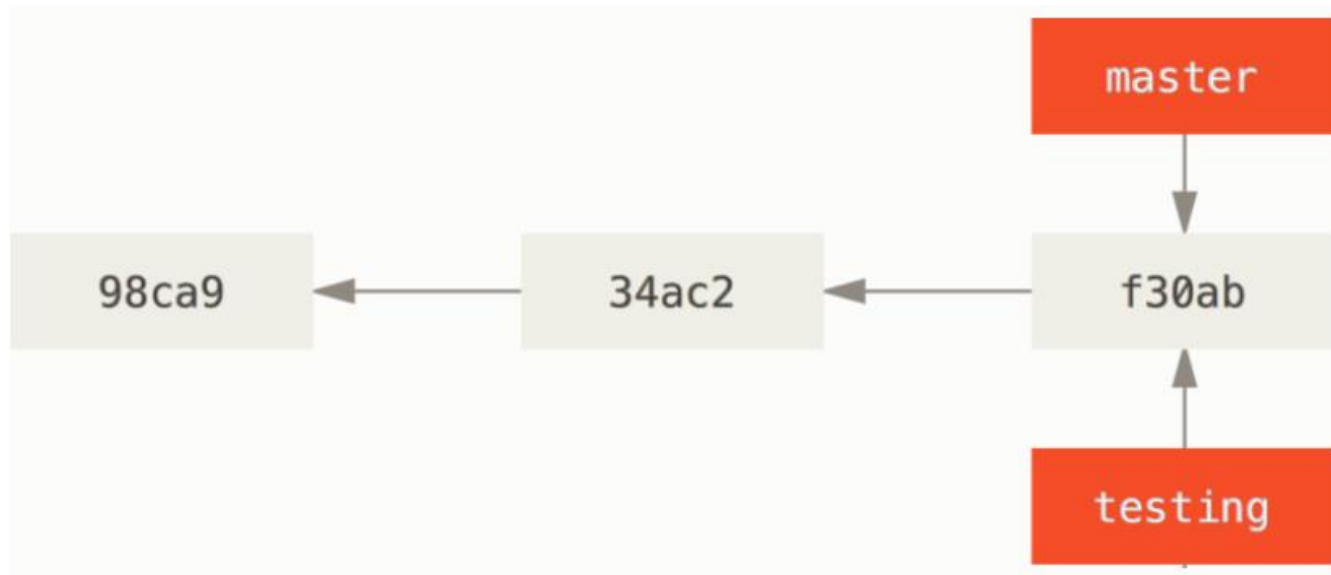
\$ git checkout -b testing



```
mingw64/c/Florida/Doce...  
pmart@LAPTOP-LDPE3MBP MINGW64 /c/Florida/D  
os/Proyecto PruebaGIT (master)  
$ git checkout -b testing
```

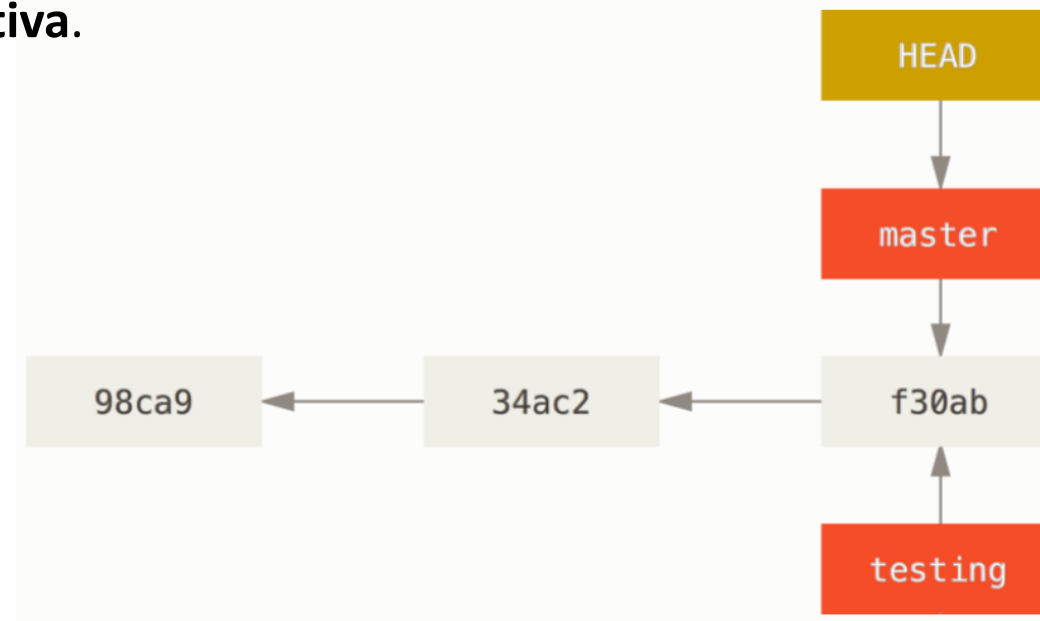
Git. Nuevas ramas.

- **Tras crear la nueva rama “testing”,** podemos ver que la tercera y última consolidación, dispone en este punto del proyecto de **2 punteros** apuntándole.
- **¿Cómo sabe Git, a partir de qué puntero, rama o camino puede ir evolucionando el proyecto?**



Git. Rama activa. HEAD

- El puntero especial **HEAD**, nos **indica qué rama evolucionará** cuando volvamos a hacer una consolidación. Es decir, cuál es la **rama activa**.
- Además, también nos indica **a partir de qué consolidación evolucionará la rama**. Es decir, la **consolidación activa**.

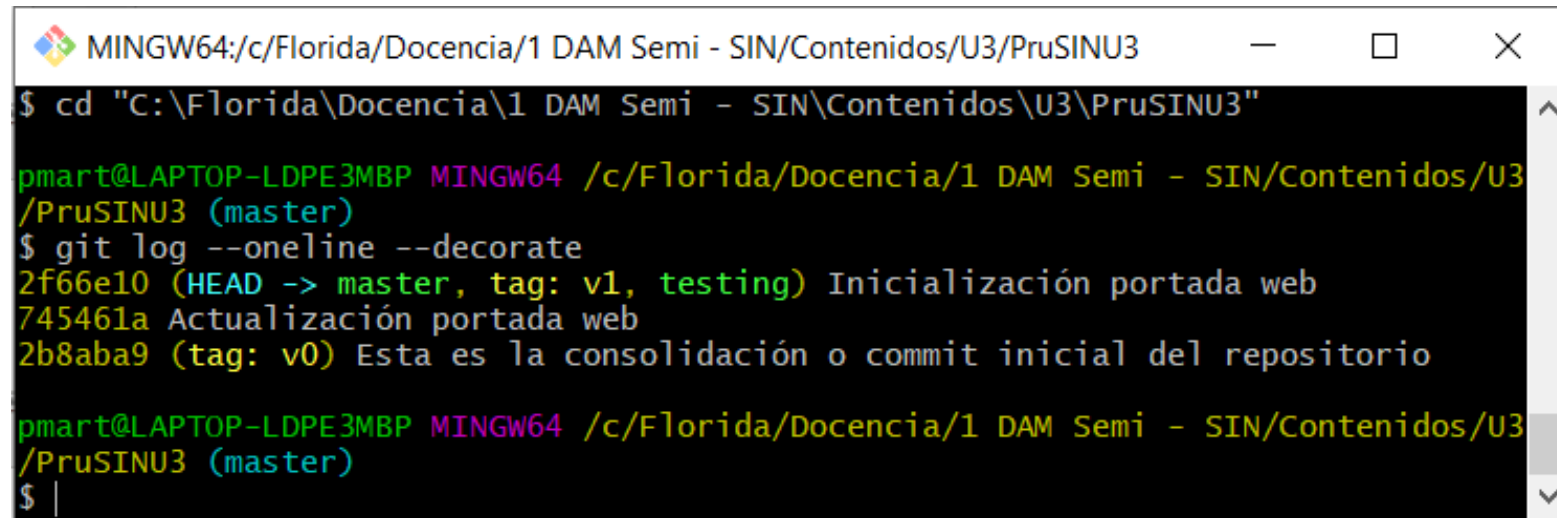


Git. Rama activa. HEAD

- En consola, podemos consultar cuál es la rama activa mediante:

\$ git log

(**la opción **--oneline** nos muestra cada consolidación en una línea)



```
MINGW64:/c/Florida/Docencia/1 DAM Semi - SIN/Contenidos/U3/PruSINU3
$ cd "C:\Florida\Docencia\1 DAM Semi - SIN\Contenidos\U3\PruSINU3"

pmart@LAPTOP-LDPE3MBP MINGW64 /c/Florida/Docencia/1 DAM Semi - SIN/Contenidos/U3/PruSINU3 (master)
$ git log --oneline --decorate
2f66e10 (HEAD -> master, tag: v1, testing) Inicialización portada web
745461a Actualización portada web
2b8aba9 (tag: v0) Esta es la consolidación o commit inicial del repositorio

pmart@LAPTOP-LDPE3MBP MINGW64 /c/Florida/Docencia/1 DAM Semi - SIN/Contenidos/U3/PruSINU3 (master)
$ |
```

Git. Cambio de rama activa.

- **Para cambiar** de una rama a otra, bastará con hacer “**checkout**” hacia la **rama destino**, convirtiendo ésta en la rama activa. También podemos usar el comando “**switch**”.
- Al cambiar de rama, puede cambiar el contenido del repositorio, reflejando el estado de la consolidación activa de la rama a la que cambiamos.



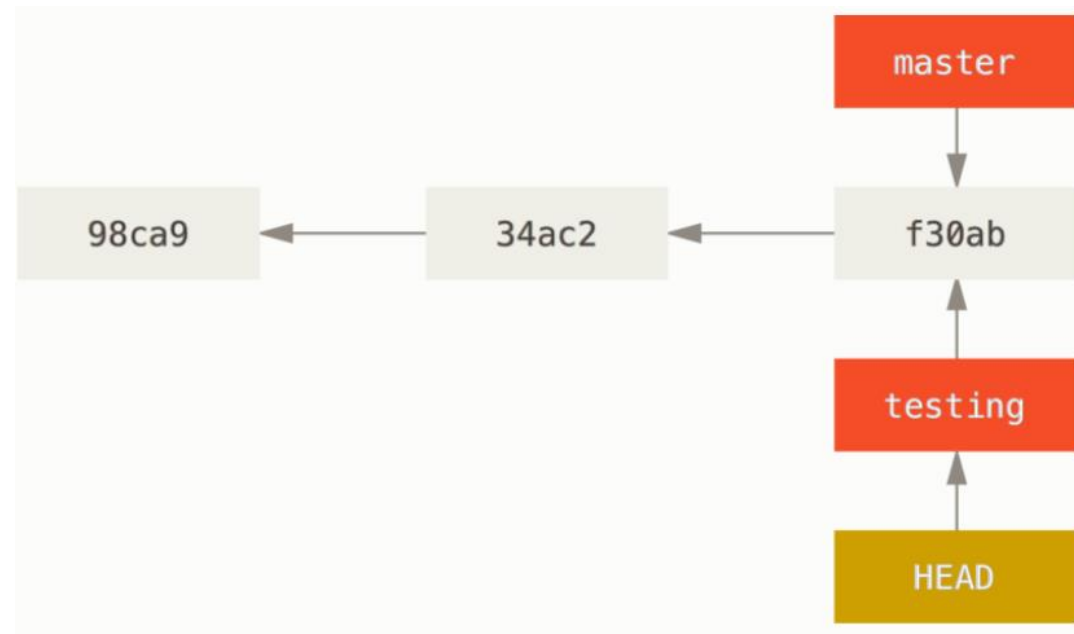
Git. Cambio de rama activa.

- Estando la rama “master” como rama activa, si introduzco el siguiente comando en consola, la rama activa pasará a ser “testing”:

\$ git checkout testing

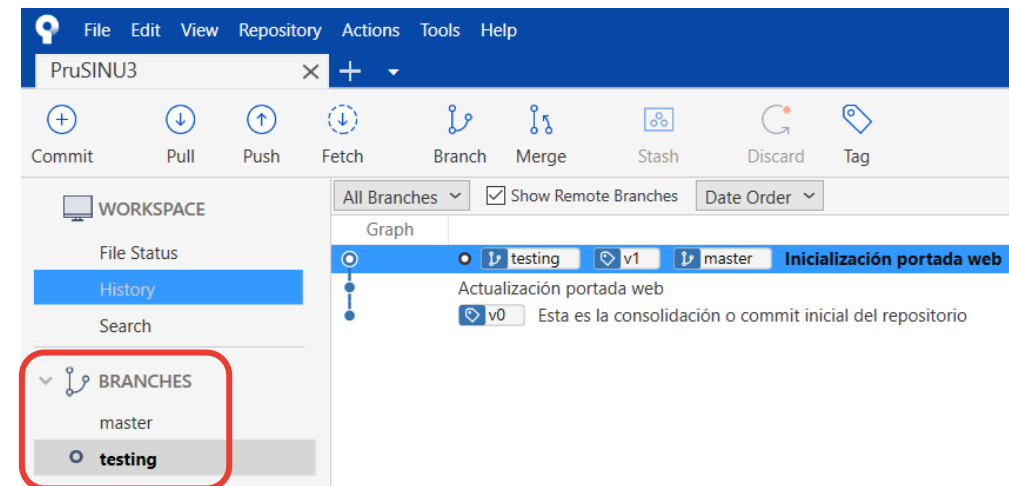
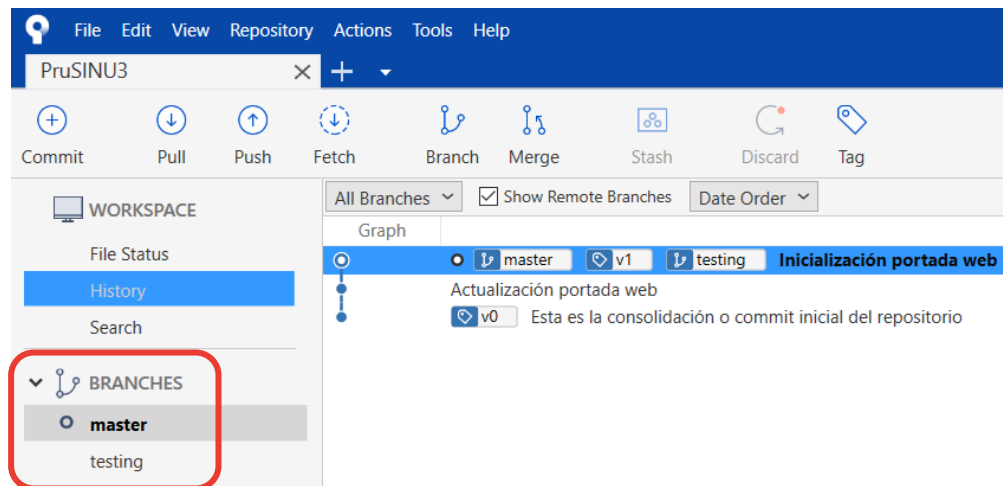
=

\$ git switch testing



Git. Cambio de rama activa.

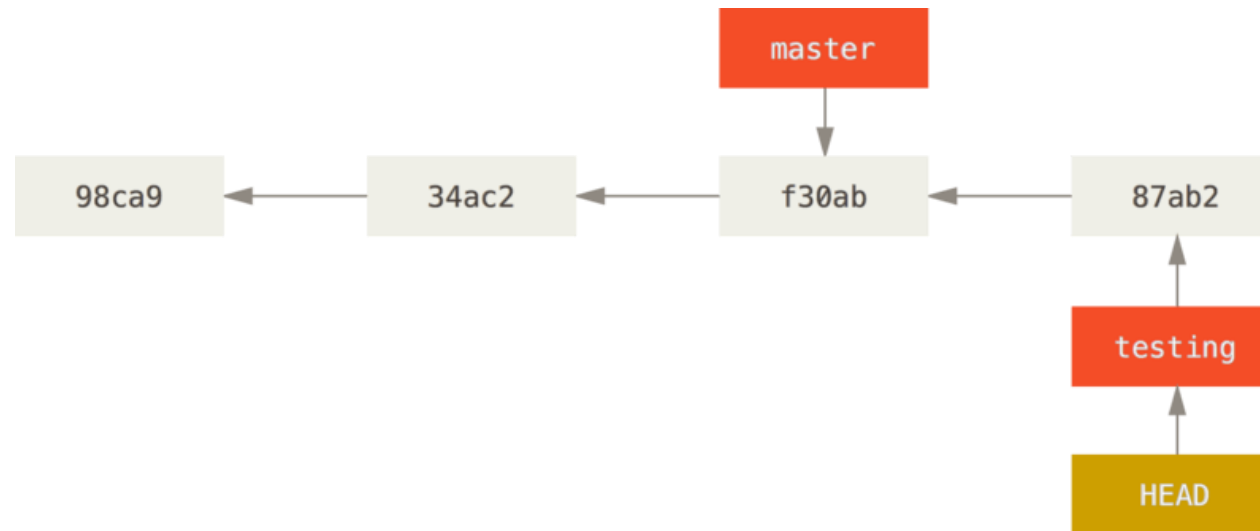
- A nivel gráfico, desde SourceTree, podemos cambiar la rama activa haciendo **doble click en el nombre de la rama** desde el menú de la izquierda.



Git. Cambio de rama activa.

- Si siendo **testing** la rama activa, hacemos una nueva consolidación. Será testing la rama que evolucionará.

\$ git commit -m "Consolidación en testing"



Git. Ramas. Ciclos de ejemplo: 1

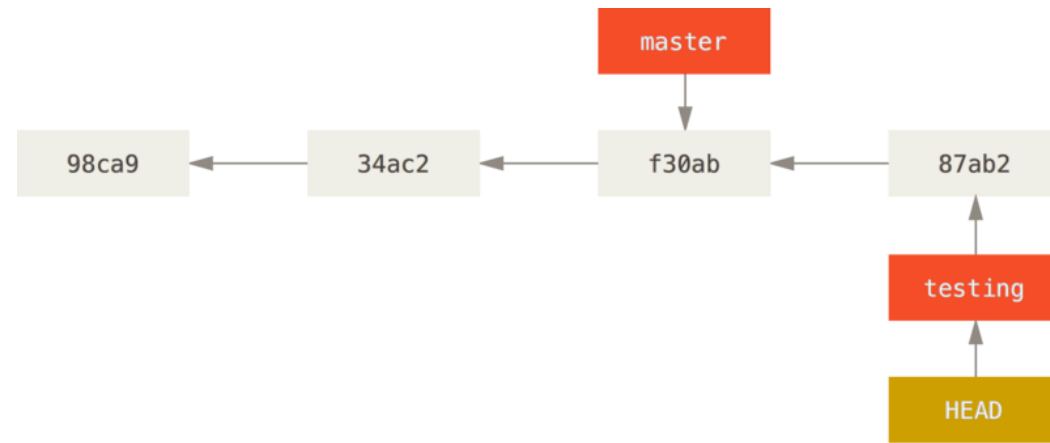
Ciclo de ejemplo 1:

- Partimos de la situación en que master y testing están en la misma consolidación.
- Es habitual y recomendable realizar las modificaciones de un proyecto en ramas distintas a la principal, master.
- En nuestro ejemplo, disponemos de la rama testing para esta finalidad.
- Teniendo **testing como rama activa** modificamos cualquier fichero del proyecto y **realizamos un commit** o consolidación.

Git. Ramas. Ciclos de ejemplo: 1

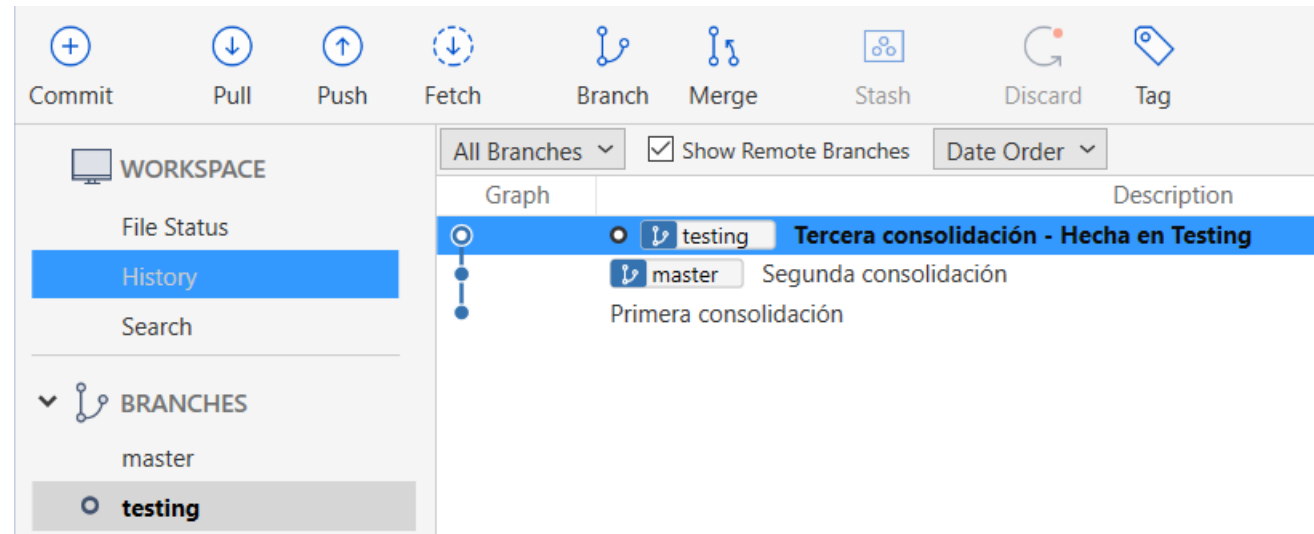
Ciclo de ejemplo 1:

- En este momento, **cada rama** de las existentes en el repositorio tiene **diferente información**, pero la rama master no ha sido modificada.
- **El contenido de ambas ramas es lineal**, es decir, la rama testing es como la rama master más los cambios aplicados en la última consolidación.



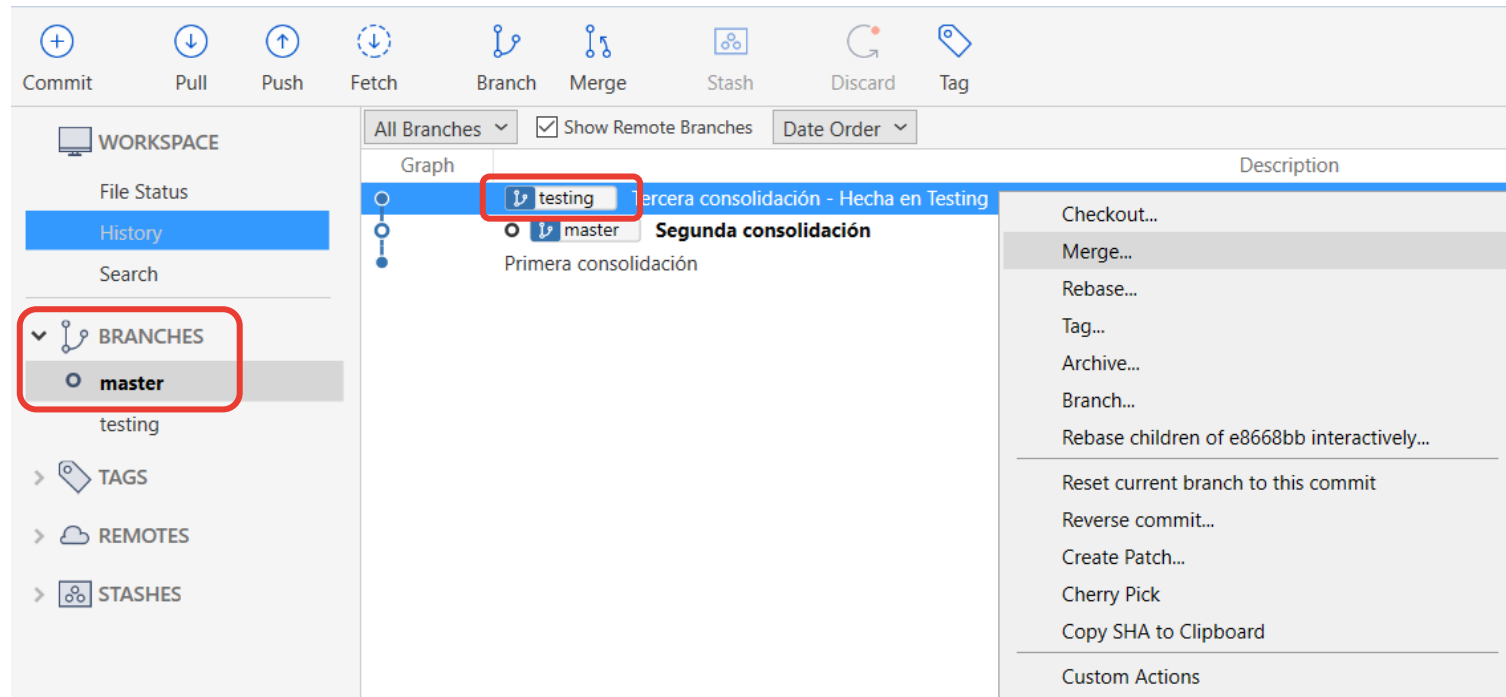
Git. Ramas. Ciclos de ejemplo: 1

- Este tipo de fusión de ramas se denominaría **merge de tipo fast forward**.



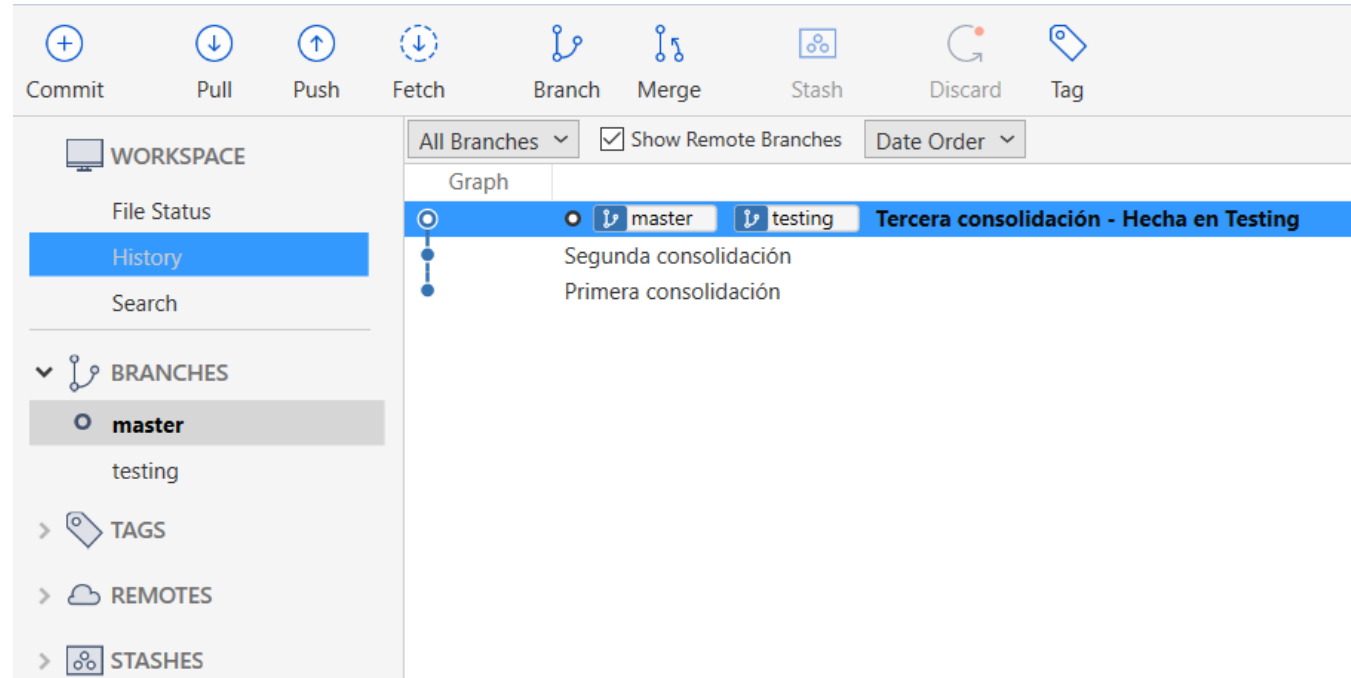
Git. Ramas. Ciclos de ejemplo: 1

- Para fusionar y consolidar las ramas, con **master** como rama activa, pulsaremos **botón derecho** sobre la consolidación de la rama **testing** y ejecutaremos “**merge**”.



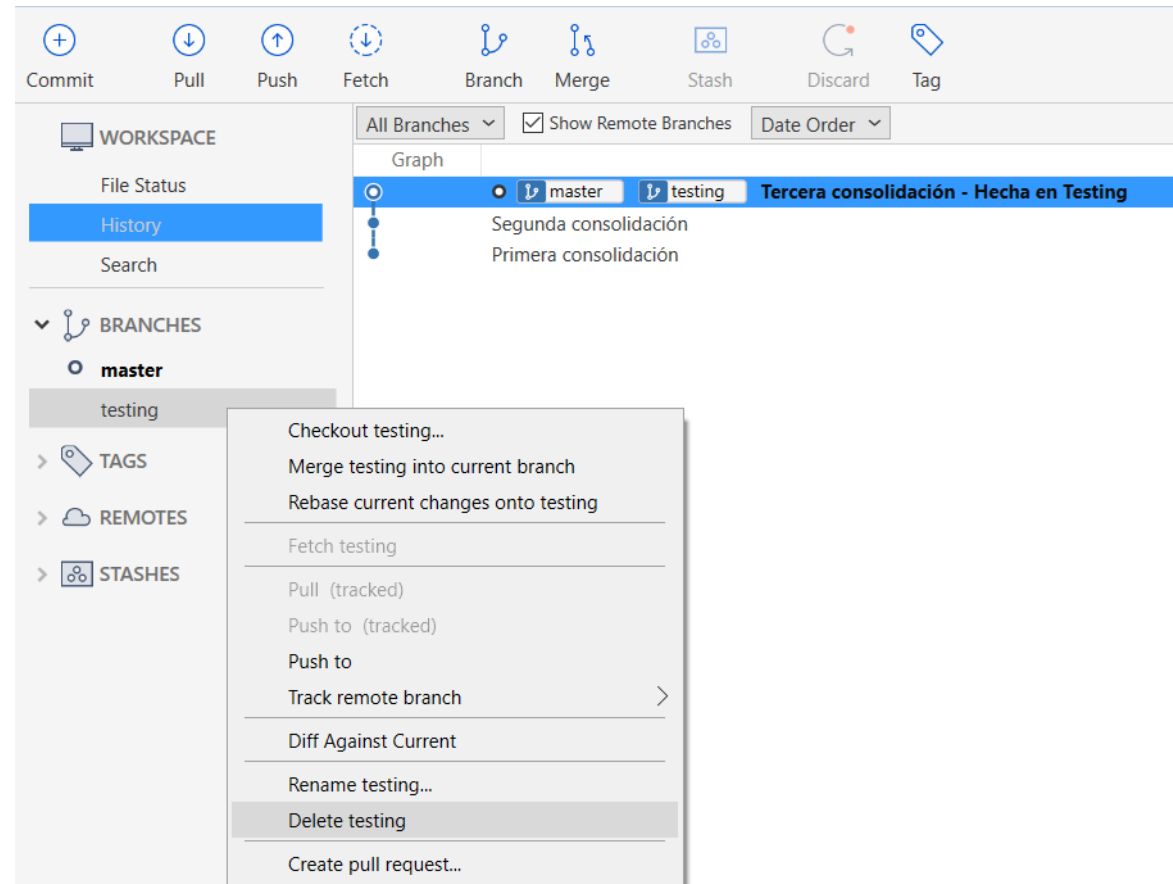
Git. Ramas. Ciclos de ejemplo: 1

- Tras aplicar la **fusión (merge) de tipo fast forward**, las ramas quedan fusionadas en una **nueva consolidación con el cambio integrado en la rama master**.



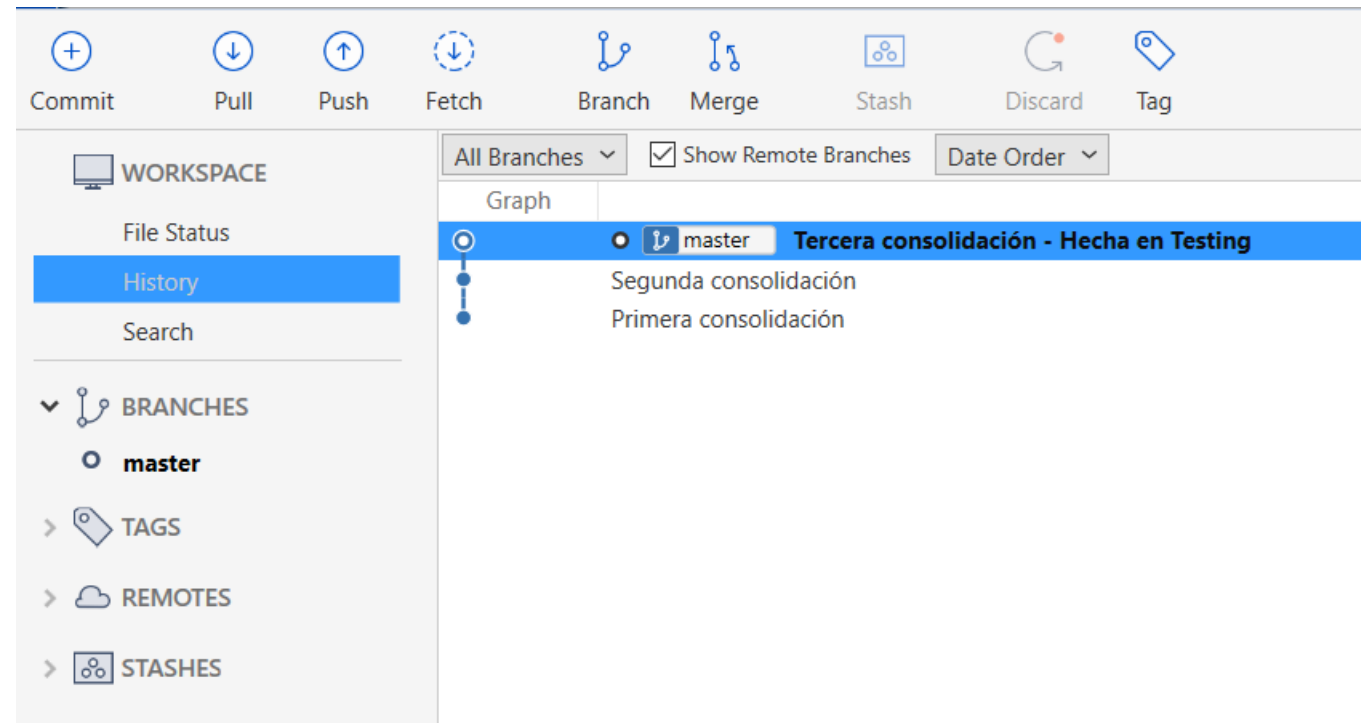
Git. Ramas. Ciclos de ejemplo: 1

- Podemos dar por concluido el trabajo en la rama testing y eliminar la rama.
- Para eliminarla, pulsaremos con el botón derecho sobre la rama y ejecutaremos la acción “Delete”.



Git. Ramas. Ciclos de ejemplo: 1

- En este punto, volvemos a disponer de una única rama, master, con el cambio integrado en ella.



Git. Ramas. Ciclos de ejemplo: 2

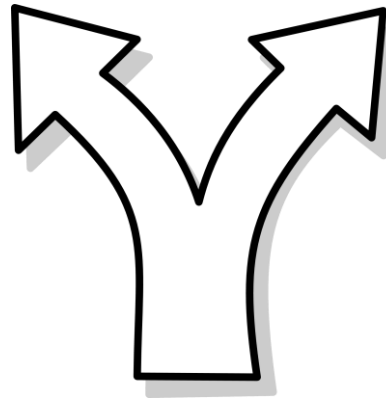
Ciclo de ejemplo 2:

- Partimos de misma situación que partía el ciclo de ejemplo 1.
- En este caso, vamos a realizar un cambio diferente en cada una de las ramas del repositorio, master y testing.
- Teniendo **testing** como rama activa modificamos, por ejemplo, el fichero **index.html**, concretamente el apartado **title**, y realizamos una consolidación.
- Posteriormente, ponemos **master** como rama activa (es interesarse darse cuenta de que, al cambiar la rama activa, no dispondremos en el repositorio del cambio en el apartado title) y modificamos el mismo fichero, concretamente el apartado **body**, y consolidamos.

Git. Ramas. Ciclos de ejemplo: 2

Ciclo de ejemplo 2:

- En este momento, **cada rama** de las existentes en el repositorio **tiene diferente información**, ha seguido una **evolución divergente**.
- Una fusión puede consolidar ambos conjuntos de cambios en el repositorio.



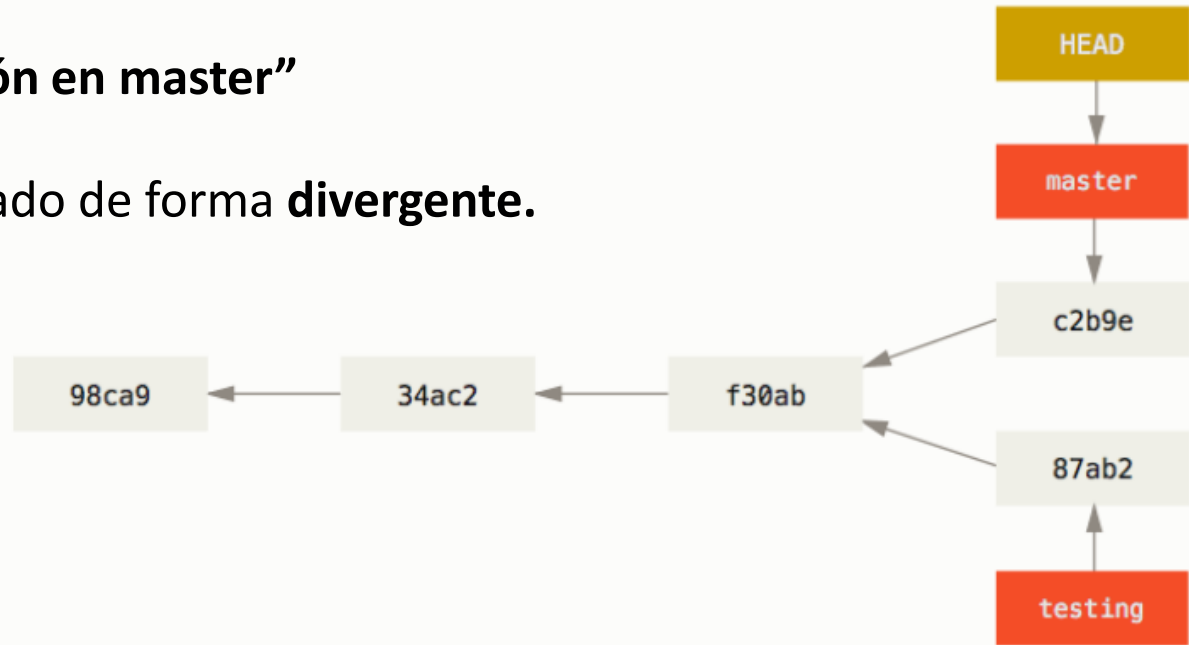
Git. Ramas. Ciclos de ejemplo: 2

- Es decir, tras haber realizado una consolidación en la rama testing, cambiamos la **rama activa** para que vuelva a ser **master** y realizamos una consolidación:

`$ git checkout master`

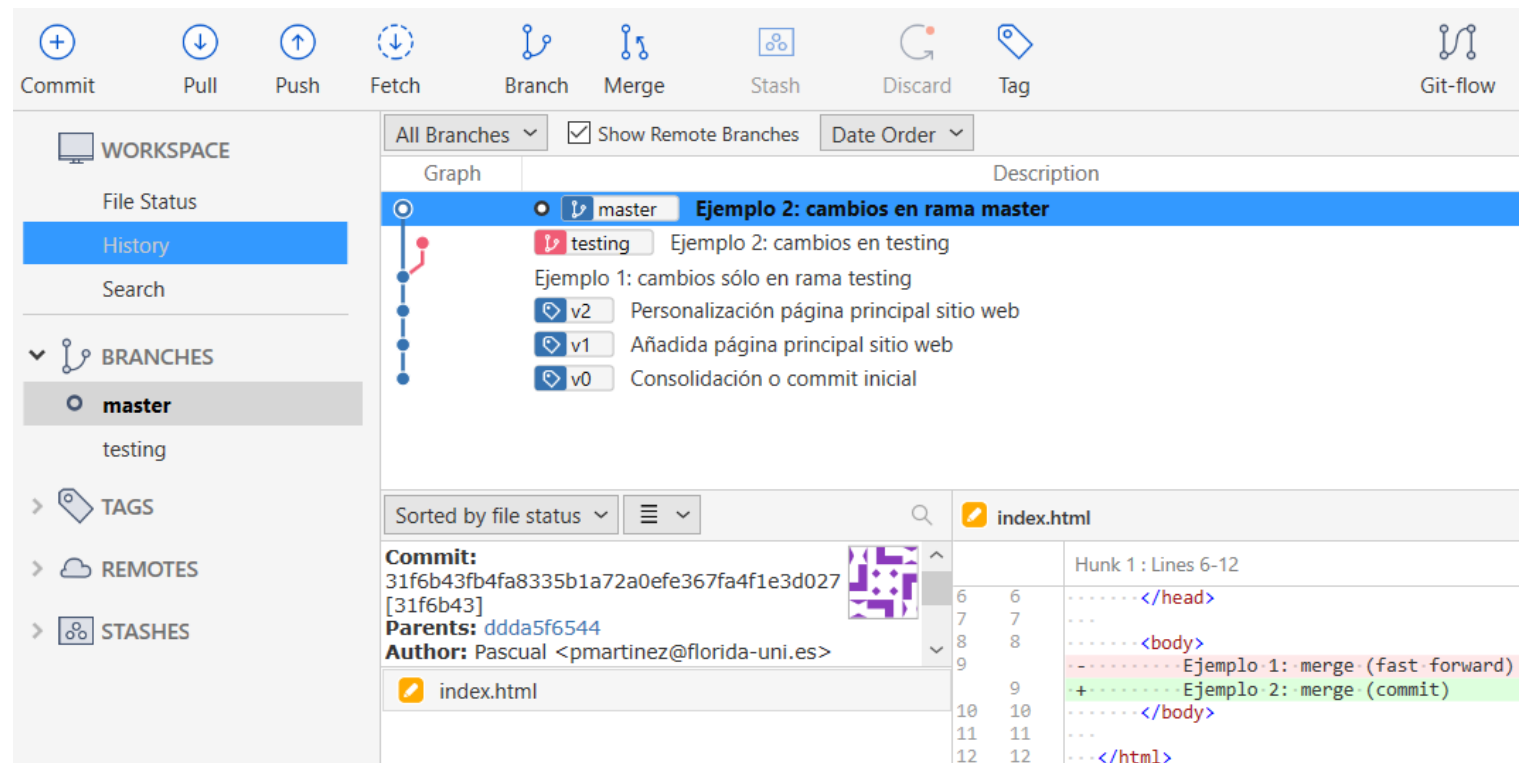
`$ git commit -m "Consolidación en master"`

- Ahora, ambas ramas han evolucionado de forma **divergente**.



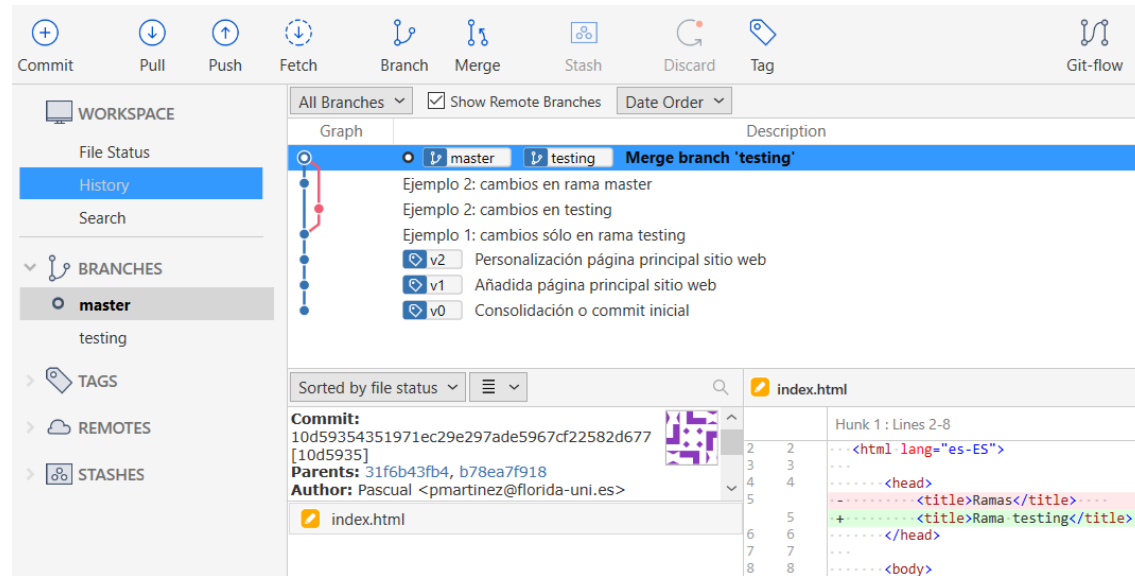
Git. Ramas. Ciclos de ejemplo: 2

- Este tipo de fusión de ramas se denominaría **merge de tipo commit**.



Git. Ramas. Ciclos de ejemplo: 2

- Para fusionar y consolidar las ramas, con **master** como rama activa, pulsaremos **botón derecho** sobre la última consolidación de la rama **testing** y ejecutaremos “**merge**”.
- Las ramas quedan fusionadas en una **nueva consolidación** con ambos cambios integrados.



Git. Ramas. Ciclos de ejemplo: 3

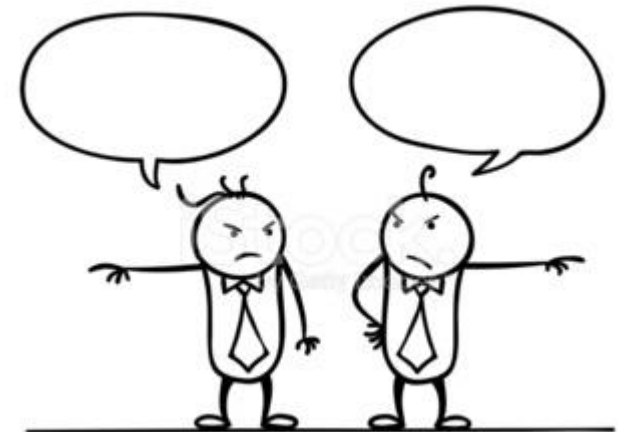
Ciclo de ejemplo 3:

- Partimos de la situación final del ciclo de ejemplo 2.
- Vamos a explicar un caso parecido al ciclo de ejemplo 2, pero con un importante matiz adicional.
- Vamos a crear una nueva rama, llamada “**portada**”.
- Vamos a realizar un cambio en cada una de las dos ramas del repositorio, pero en este caso vamos a modificar la **misma línea del mismo fichero del proyecto**, en ambas ramas. **En concreto, en el fichero index.html, modificamos el apartado title.**

Git. Ramas. Ciclos de ejemplo: 3

Ciclo de ejemplo 3:

- Cuando intentemos fusionar (merge) las ramas master y portada, siendo master la rama activa, Git nos indicará que se ha producido un **conflicto**, porque se ha modificado justo lo mismo en ambas ramas.
- ¿Cuál de los dos cambios debe consolidarse en el repositorio?...



Git. Ramas. Ciclos de ejemplo: 3

The screenshot shows the Git GUI interface. The top toolbar includes icons for Commit, Pull, Push, Fetch, Branch, Merge, Stash, Discard, Tag, and Git-flow. The left sidebar shows the Workspace (File Status, History, Search) and Branches (master, portada, testing). The main area displays a commit history graph and a list of commits. The 'portada' branch is selected, and a context menu is open over the 'Ejemplo 3: cambios en rama portada' commit.

Commit History:

- master: Ejemplo 3: cambios en rama master
- portada: Ejemplo 3: cambios en rama portada
- testing: Merge branch 'testing'
- Ejemplo 2: cambios en rama master
- Ejemplo 2: cambios en testing
- Ejemplo 1: cambios sólo en rama testing
- v2: Personalización página principal sitio web
- v1: Añadida página principal sitio web
- v0: Consolidación o commit inicial

Commit Details:

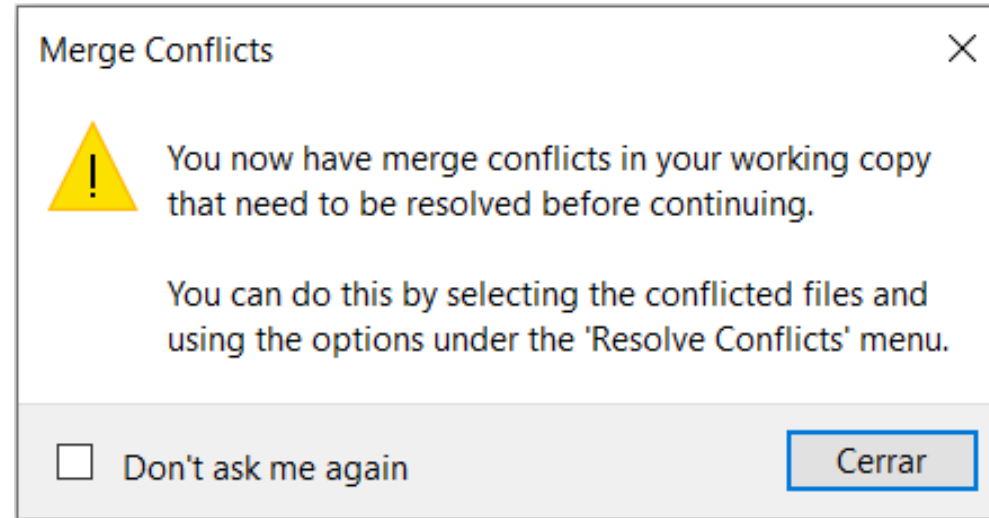
Commit: 4c3495832a4f58e553a89ec5d7da7319a9b5534b [4c34958]
Parents: 10d5935435
Author: Pascual <pmartinez@florida-uni.es>

Files: index.html

Context Menu Options:

- Checkout...
- Merge...
- Rebase...
- Tag...
- Archive...
- Branch...
- Rebase children of 4c34958 interactively...
- Reset current branch to this commit
- Reverse commit...
- Create Patch...
- Cherry Pick
- Copy SHA to Clipboard
- Custom Actions

Git. Ramas. Ciclos de ejemplo: 3



- Será necesario **resolver manualmente este conflicto** en algún momento, para que el contenido del fichero sea coherente.

Git. Ramas. Ciclos de ejemplo: 3

```
index.html: Bloc de notas
Archivo Edición Formato Ver Ayuda
<!DOCTYPE html>
<html lang="es-ES">

<head>
<<<<<< HEAD
    <title>Rama master</title>
=====
    <title>Rama portada</title>
>>>>>> portada
</head>

<body>
Este es el cuerpo de la página master
</body>

</html>
```

The screenshot shows the Git GUI interface. The top toolbar includes buttons for Commit, Pull, Push, Fetch, Branch, Merge, Stash, Discard, and Tag. The left sidebar shows the Workspace with File Status, History, and Search, and the BRANCHES section with master, portada, and testing branches. The main area displays the commit history graph and a list of commits. The 'Uncommitted changes' section shows the current state of the index.html file, which is staged for commit. The file content is displayed in a diff view, showing the changes made in the current branch compared to the previous commit.

Commit history (from top to bottom):

- master: Ejemplo 3: cambios en rama master
- portada: Ejemplo 3: cambios en rama portada
- testing: Merge branch 'testing'
- master: Ejemplo 2: cambios en rama master
- testing: Ejemplo 2: cambios en testing
- testing: Ejemplo 1: cambios sólo en rama testing
- v2: Personalización página principal sitio web
- v1: Añadida página principal sitio web
- v0: Consolidación o commit inicial

Staged files: index.html

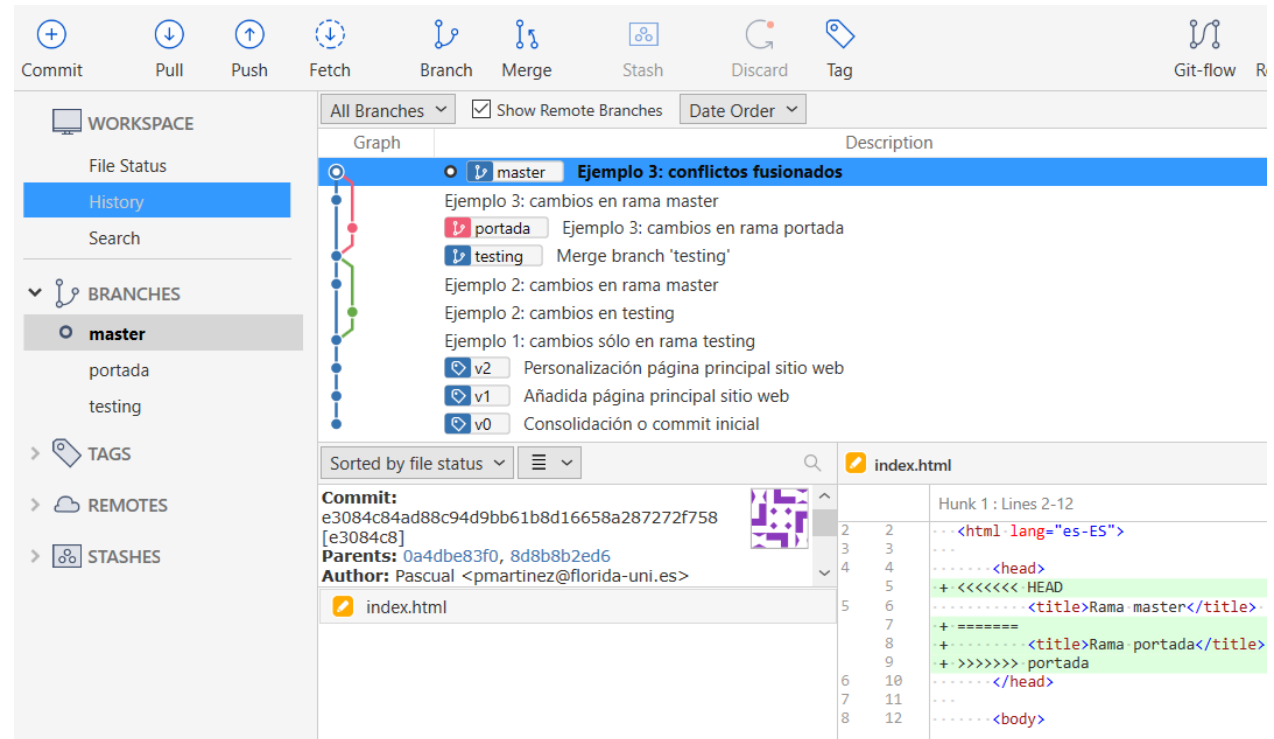
Unstaged files: index.html

Hunk 1: Lines 2-12

```
2 2  ...<html lang="es-ES">
3 3  ...
4 4  ...<head>
5 5  + <<<<<< HEAD
5 6  + ...<title>Rama master</title>...
7 7  + =====
8 8  + ...<title>Rama portada</title>...
9 9  + >>>>>> portada
6 10 ...</head>
7 11 ...
8 12 ...<body>
```

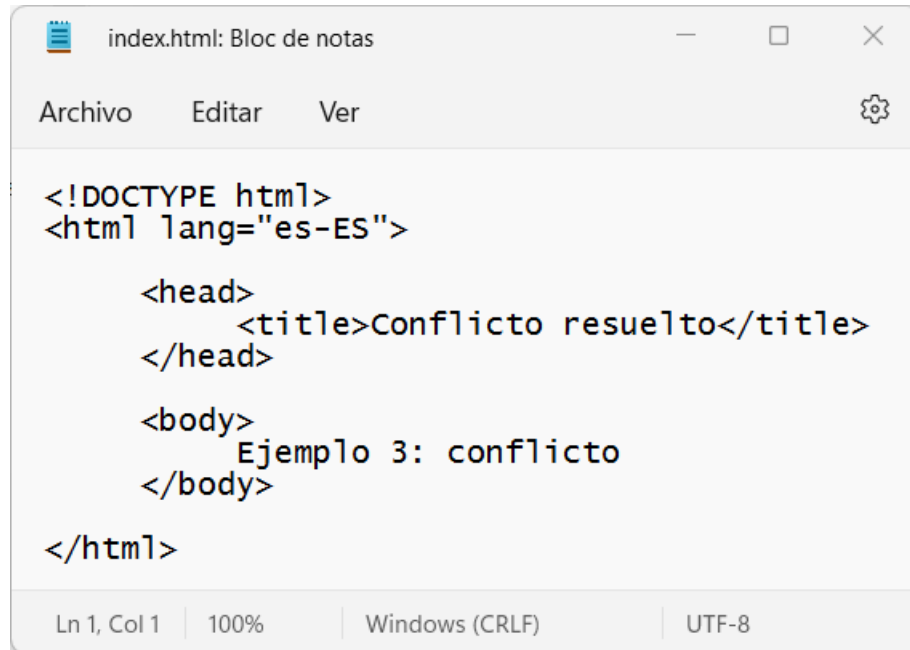
Git. Ramas. Ciclos de ejemplo: 3

- Llegados a este punto, se podría hacer una consolidación con todos los cambios que han generado el conflicto y posteriormente resolver el conflicto en una nueva consolidación.
- O bien se podría resolver el conflicto directamente y después proceder a consolidar.



Git. Ramas. Ciclos de ejemplo: 3

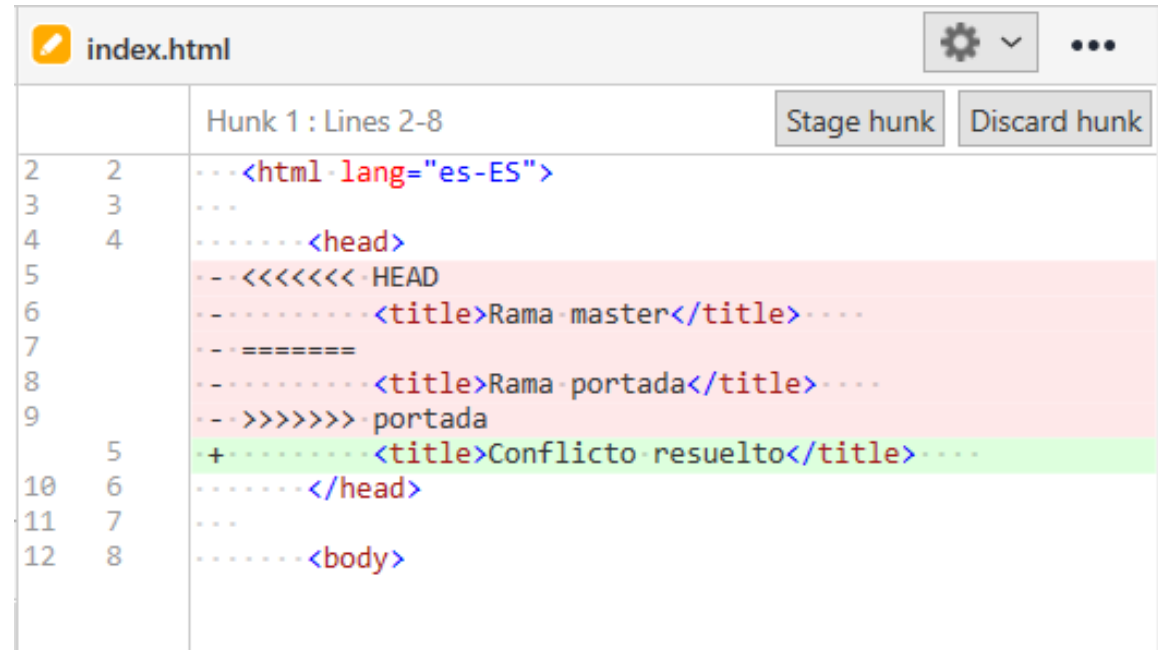
- Modificación del fichero que ha provocado el conflicto para consolidar posteriormente.



A screenshot of a text editor window titled "index.html: Bloc de notas". The editor shows the following HTML code:

```
<!DOCTYPE html>
<html lang="es-ES">
  <head>
    <title>Conflicto resuelto</title>
  </head>
  <body>
    Ejemplo 3: conflicto
  </body>
</html>
```

The status bar at the bottom indicates "Ln 1, Col 1", "100%", "Windows (CRLF)", and "UTF-8".

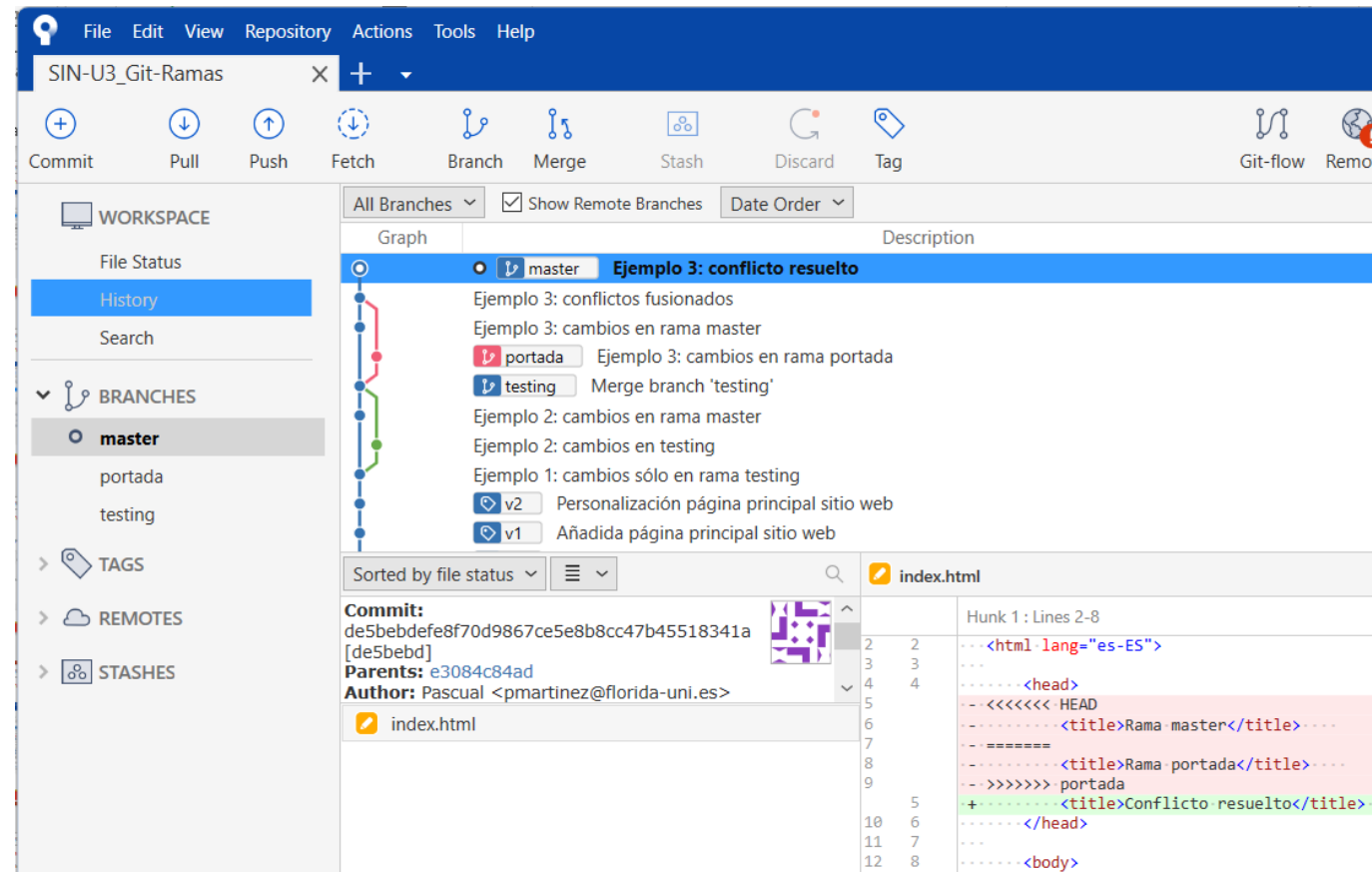


A screenshot of a diff viewer window titled "index.html". It shows a hunk of code with the title "Hunk 1 : Lines 2-8". The diff shows the following changes:

```
2 2 ...<html lang="es-ES">
3 3 ...
4 4 .....<head>
5 5 --<<<<<<< HEAD
6 6 .....<title>Rama master</title>....
7 7 .....
8 8 .....<title>Rama portada</title>....
9 9 -->>>>>>> portada
10 5 + .....<title>Conflicto resuelto</title>....
11 6 .....</head>
12 7 ...
13 8 .....<body>
```

The status bar at the bottom indicates "Stage hunk" and "Discard hunk".

Git. Ramas. Ciclos de ejemplo: 3



Git. Ramas. Ciclos de ejemplo: 3

****Nota:** en este punto, también sería posible usar otras herramientas externas especializadas en la resolución de conflictos, si disponemos de ellas y están configuradas.

