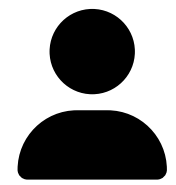


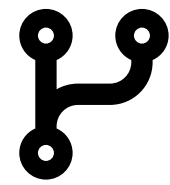
# Flujo de trabajo



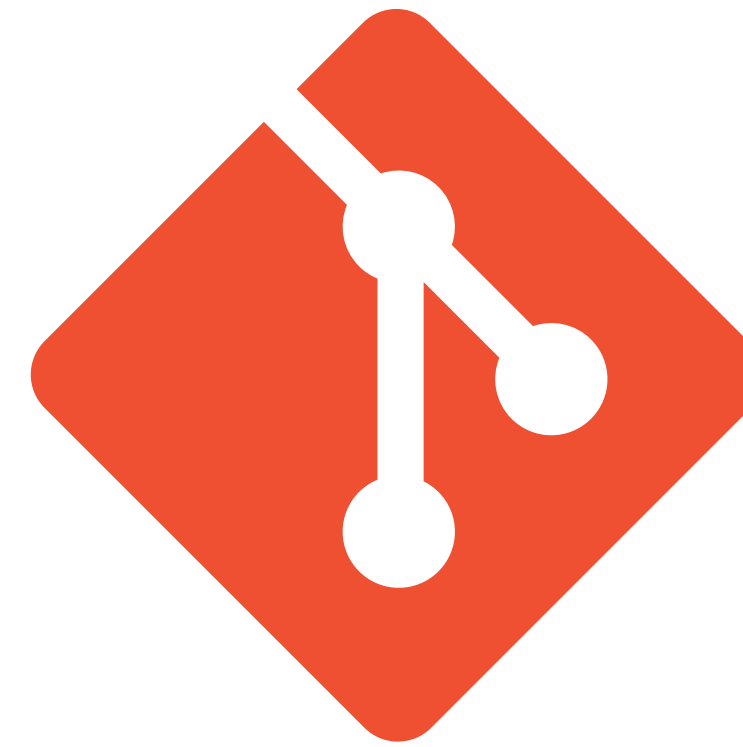
Individual



Distribuido



Ramificado



# git

# Características del flujo de trabajo

<b>Individual</b>	Sólo nosotros hacemos cambios en el repositorio
<b>Distribuido</b>	Hay una copia local del repositorio y una o más copias remotas
<b>Ramificado</b>	Trabajamos con varias líneas de tiempo en paralelo

# Repositorio local

```
git init cuarto-proyecto
cd cuarto-proyecto
echo "cuarto-proyecto" > README.md
git add --all
git commit -m "Commit inicial"
```

```
Inicializado repositorio Git vacío en /Users/.../cuarto-proyecto/.git/
```

```
[master (commit-raíz) e7cca41] Commit inicial
1 file changed, 1 insertion(+)
create mode 100644 README.md
```

# Crear una rama

```
git branch test
```

(si todo va bien, no sale nada en pantalla)

# Comprobar la rama actual

```
git status
```

```
En la rama master  
nada para hacer commit, el árbol de trabajo está limpio
```

# Seleccionar otra rama

```
git switch test
```

```
Cambiado a rama 'test'
```

# Crear y confirmar cambios

```
echo "Contenido del fichero C" > c.txt  
git add c.txt  
git commit -m "Creado el fichero C"
```

```
[test 593e5f3] Creado el fichero C  
1 file changed, 1 insertion(+)  
create mode 100644 c.txt
```

# Mostrar la línea de tiempo

```
git log
```

```
commit 593e5f34114b1ec0eb4695985db692291726a06e (HEAD -> test)
Author: Ion Jaureguialzo Sarasola <ion@jaureguialzo.com>
Date: Thu Sep 12 11:08:50 2024 +0200
```

```
    Creado el fichero C
```

```
commit e7cca41c47494ba65cee4ef6e51d0825f858ce19 (master)
Author: Ion Jaureguialzo Sarasola <ion@jaureguialzo.com>
Date: Thu Sep 12 11:03:46 2024 +0200
```

```
    Commit inicial
```

# Fusionar las ramas

```
git switch master
```

```
git merge test
```

```
Cambiado a rama 'master'
```

```
Actualizando e7cca41..593e5f3
```

```
Fast-forward
```

```
c.txt | 1 +
```

```
1 file changed, 1 insertion(+)
```

```
create mode 100644 c.txt
```

# Mostrar la línea de tiempo

```
git log
```

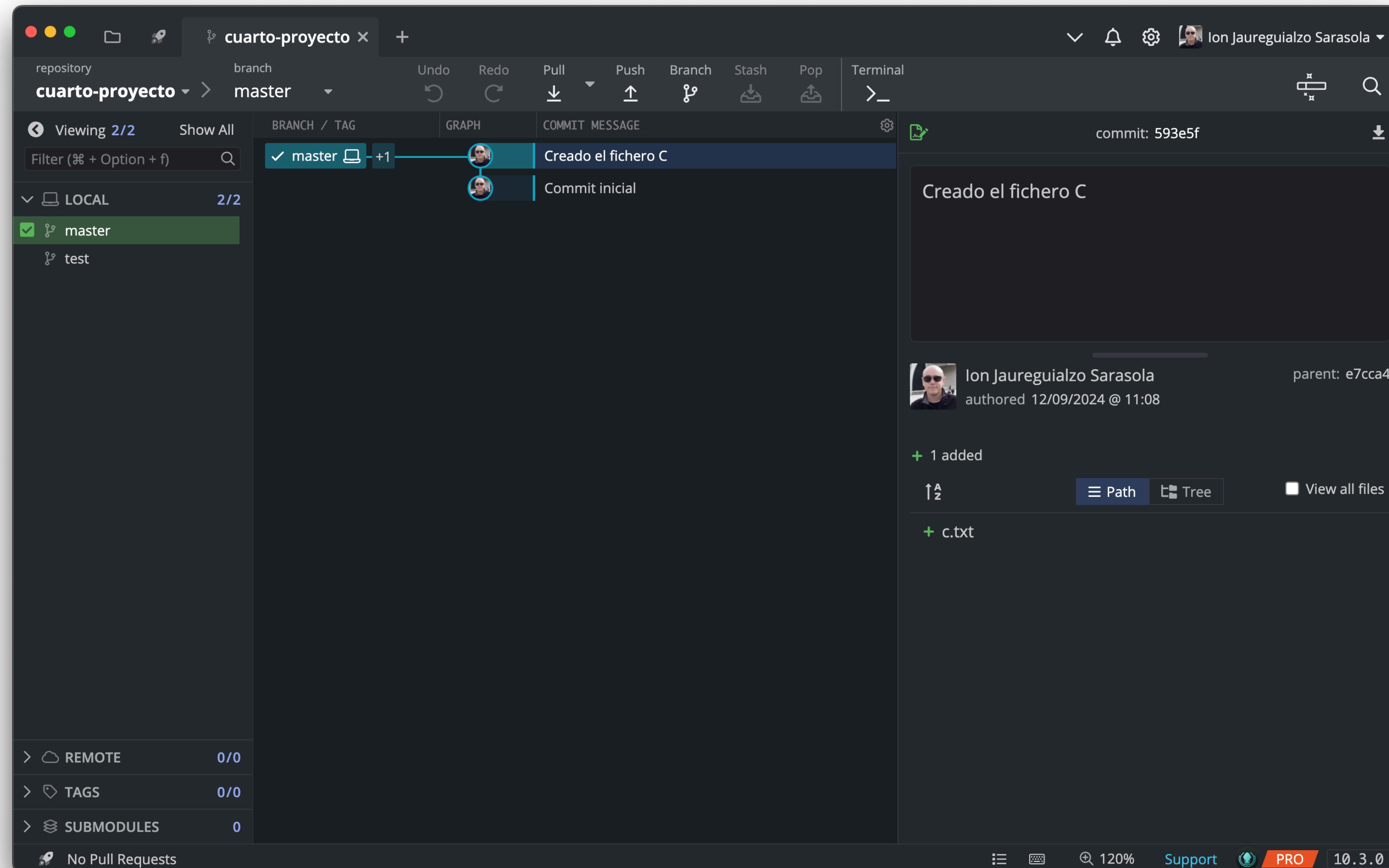
```
commit 593e5f34114b1ec0eb4695985db692291726a06e (HEAD -> master, test)
Author: Ion Jaureguialzo Sarasola <ion@jaureguialzo.com>
Date: Thu Sep 12 11:08:50 2024 +0200
```

```
    Creado el fichero C
```

```
commit e7cca41c47494ba65cee4ef6e51d0825f858ce19
Author: Ion Jaureguialzo Sarasola <ion@jaureguialzo.com>
Date: Thu Sep 12 11:03:46 2024 +0200
```

```
    Commit inicial
```

# Mostrar la línea de tiempo

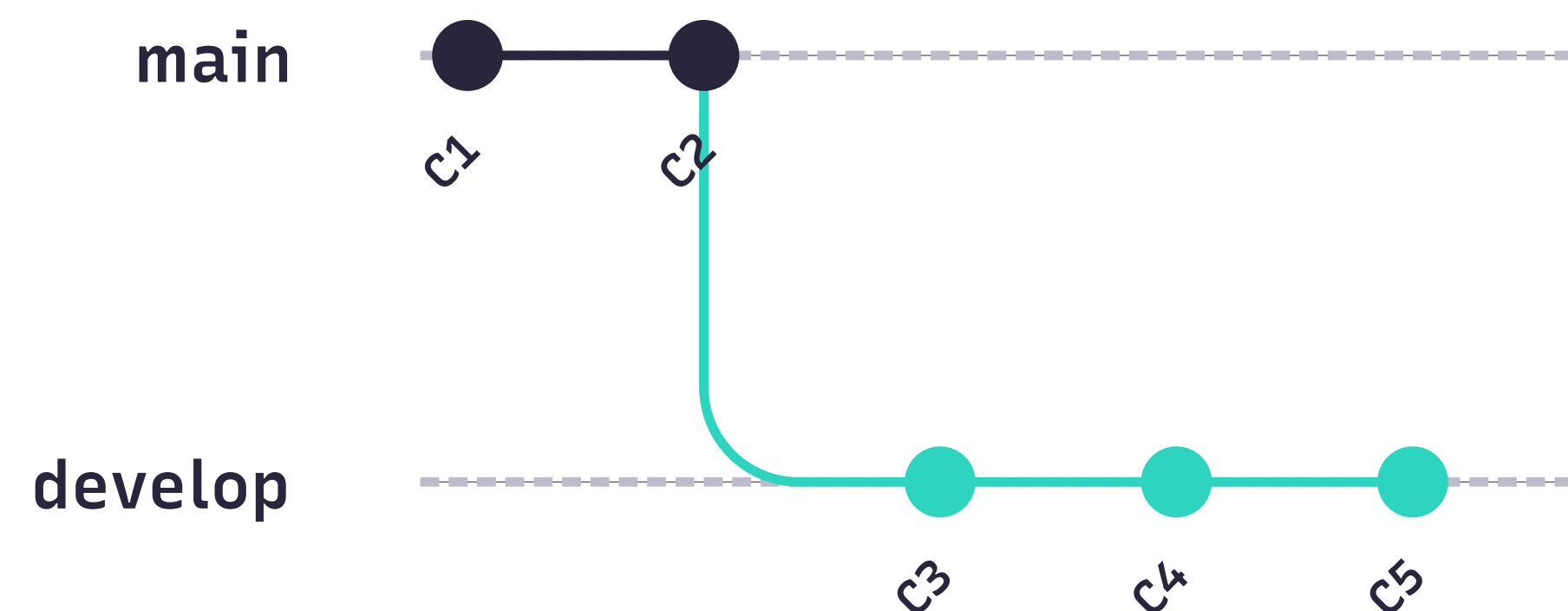


# Posibles situaciones al fusionar

- Avance rápido: `fast-forward`
- Fusión automática: `merge`
- Conflicto: Intervención manual + `merge`

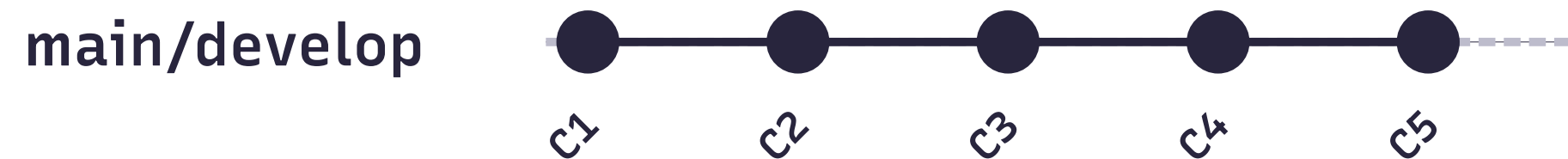
# Avance rápido

- Se va a hacer la fusión de una rama secundaria `develop` con la rama actual `main`
- No ha habido nuevos commits en la rama actual



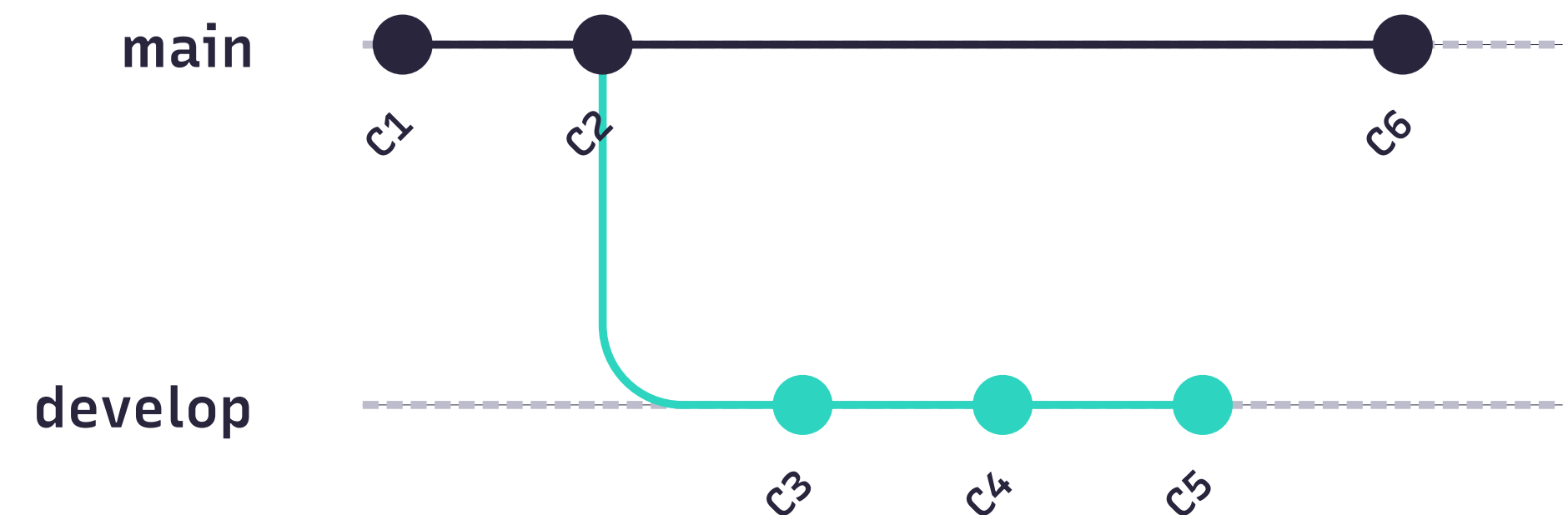
# Avance rápido

- En el `fast-forward` todos los commits de la rama secundaria se añadirán tal cual en la rama actual y esta acabará apuntando al último commit publicado en la rama secundaria



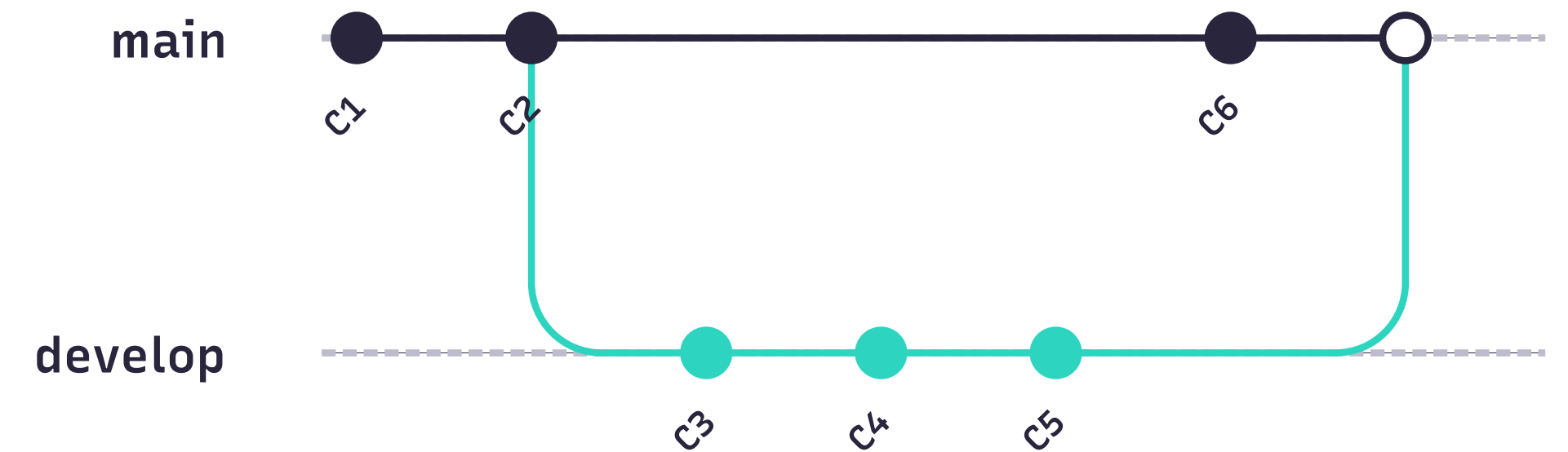
# Fusión automática

- Se va a hacer la fusión de una rama secundaria `develop` con la rama actual `main`
- Hay commits publicados en la rama actual desde el momento de la bifurcación
- No hay cambios que entren en conflicto entre ambas ramas



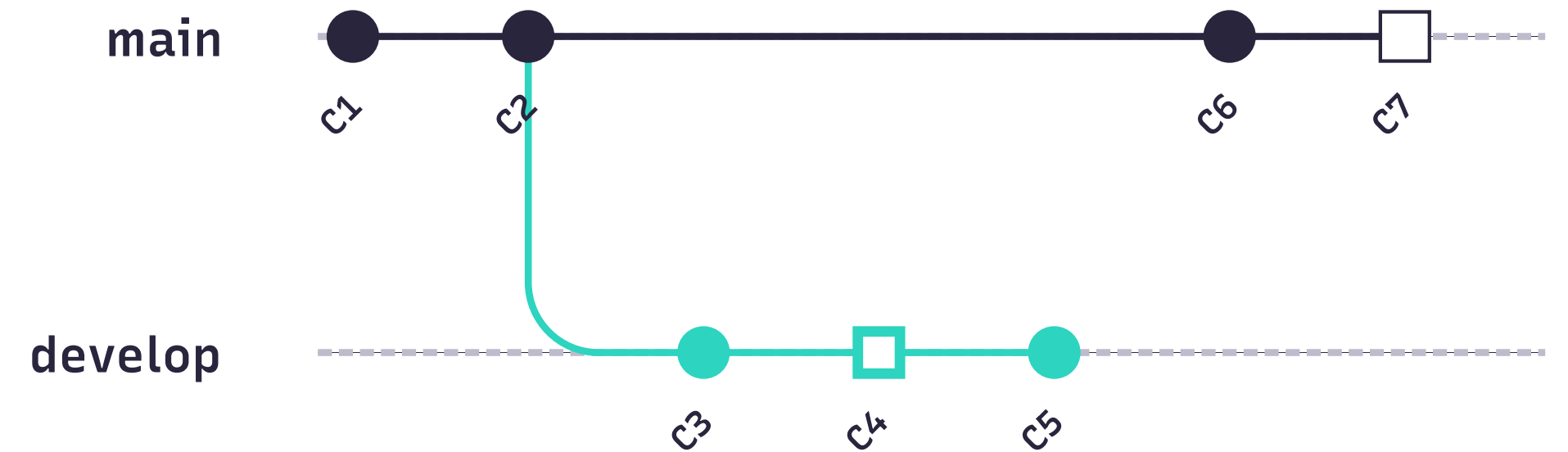
# Fusión automática

- El `merge` consistirá en la publicación de un commit automático que unirá de nuevo las dos ramas y que añadirá los cambios de la rama secundaria a la rama actual



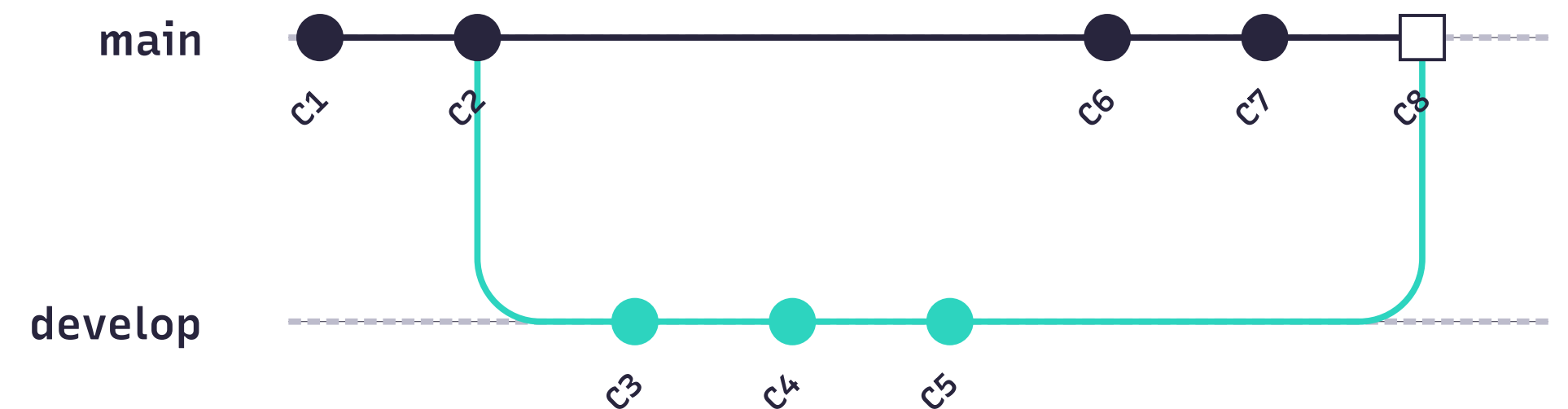
# Conflicto

- Se va a hacer la fusión de una rama secundaria `develop` con la rama actual `main`
- Hay commits publicados en la rama actual desde el momento de la bifurcación
- Hay cambios conflictivos entre las dos ramas (entre `C4` y `C7`)



# Conflicto

- En el momento del `merge` Git alertará de los cambios conflictivos y no lo completará automáticamente
- Tendremos que intervenir manualmente para seleccionar los cambios que queremos incluir en el commit
- Una vez seleccionados los cambios, se realiza el commit correspondiente al `merge` (C8) y se termina el proceso de fusión



# Provocar un conflicto

```
echo "Contenido del fichero D" > d.txt  
git add d.txt  
git commit -m "Creado el fichero D"
```

```
[master 080b7a3] Creado el fichero D  
1 file changed, 1 insertion(+)  
create mode 100644 d.txt
```

# Provocar un conflicto

```
git switch test
```

```
echo "Nuevo contenido del fichero D" > d.txt
```

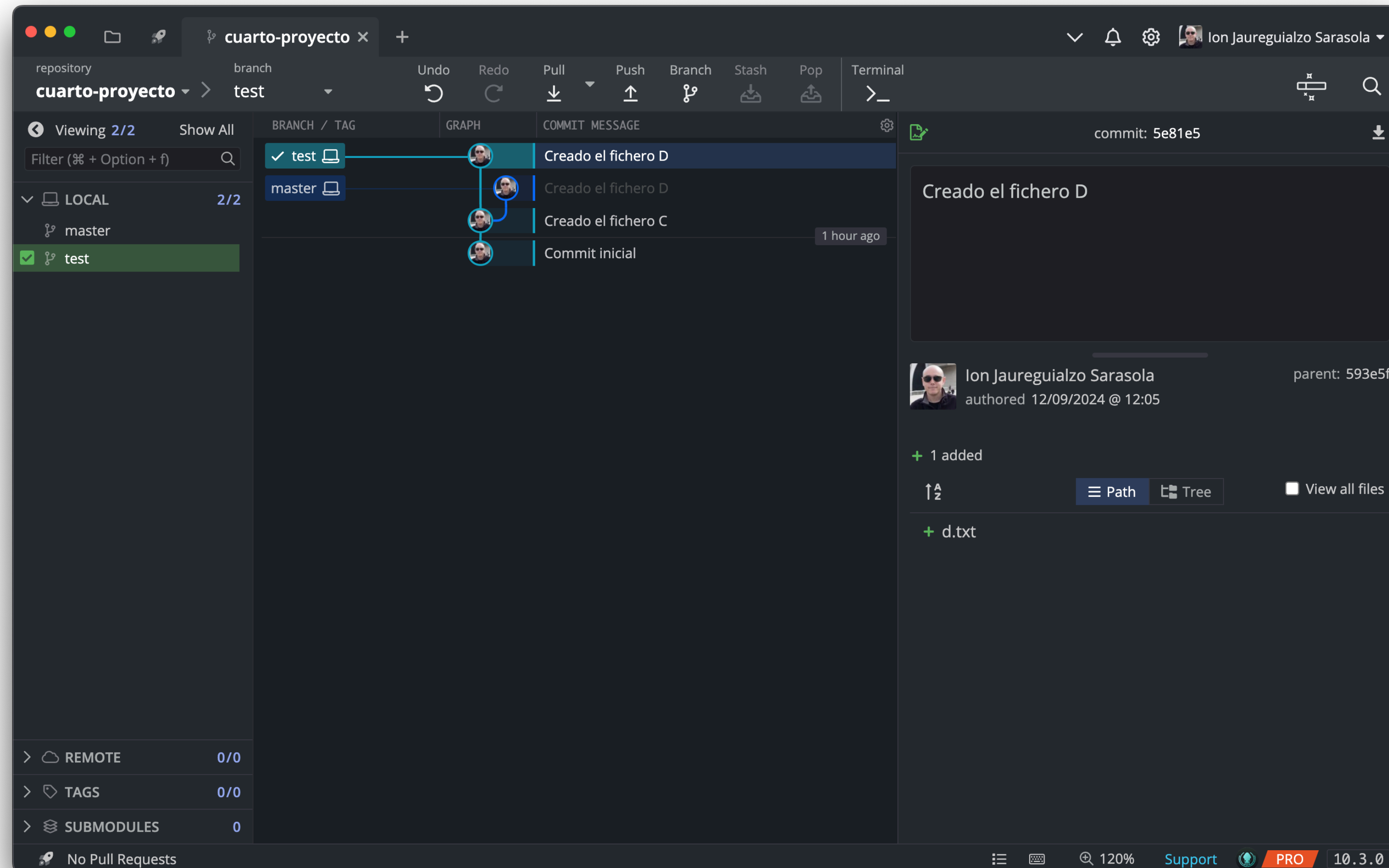
```
git add d.txt
```

```
git commit -m "Creado el fichero D"
```

```
Cambiado a rama 'test'
```

```
[test 5e81e5a] Creado el fichero D  
1 file changed, 1 insertion(+)  
create mode 100644 d.txt
```

# Mostrar la línea de tiempo



# Fusionar las ramas

```
git switch master
```

```
git merge test
```

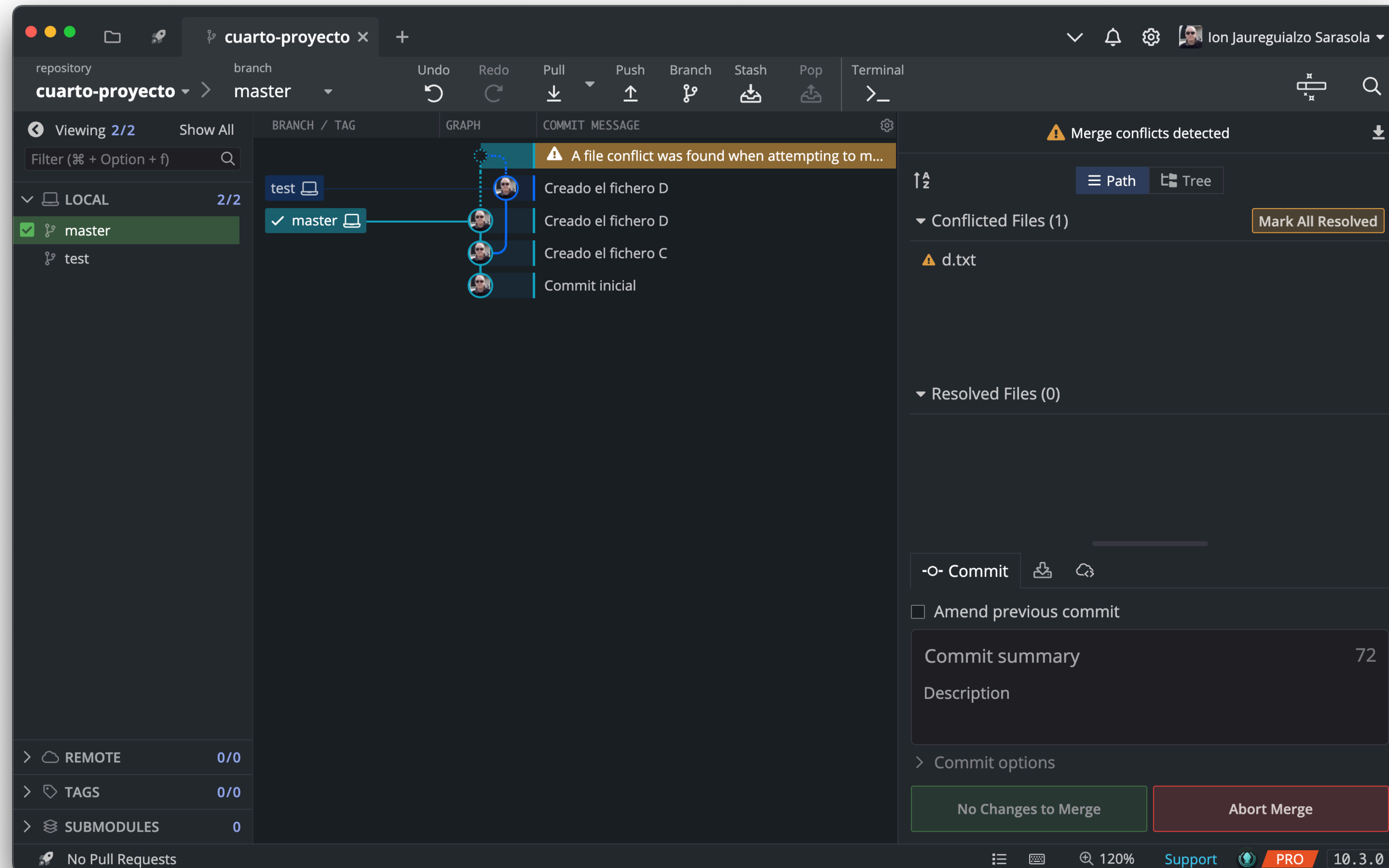
```
Cambiado a rama 'master'
```

```
Auto-fusionando d.txt
```

```
CONFLICTO (agregar/agregar): Conflicto de fusión en d.txt
```

```
Fusión automática falló; arregle los conflictos y luego realice un commit con el resultado.
```

# Mostrar la línea de tiempo



# Solucionar el conflicto

nano d.txt

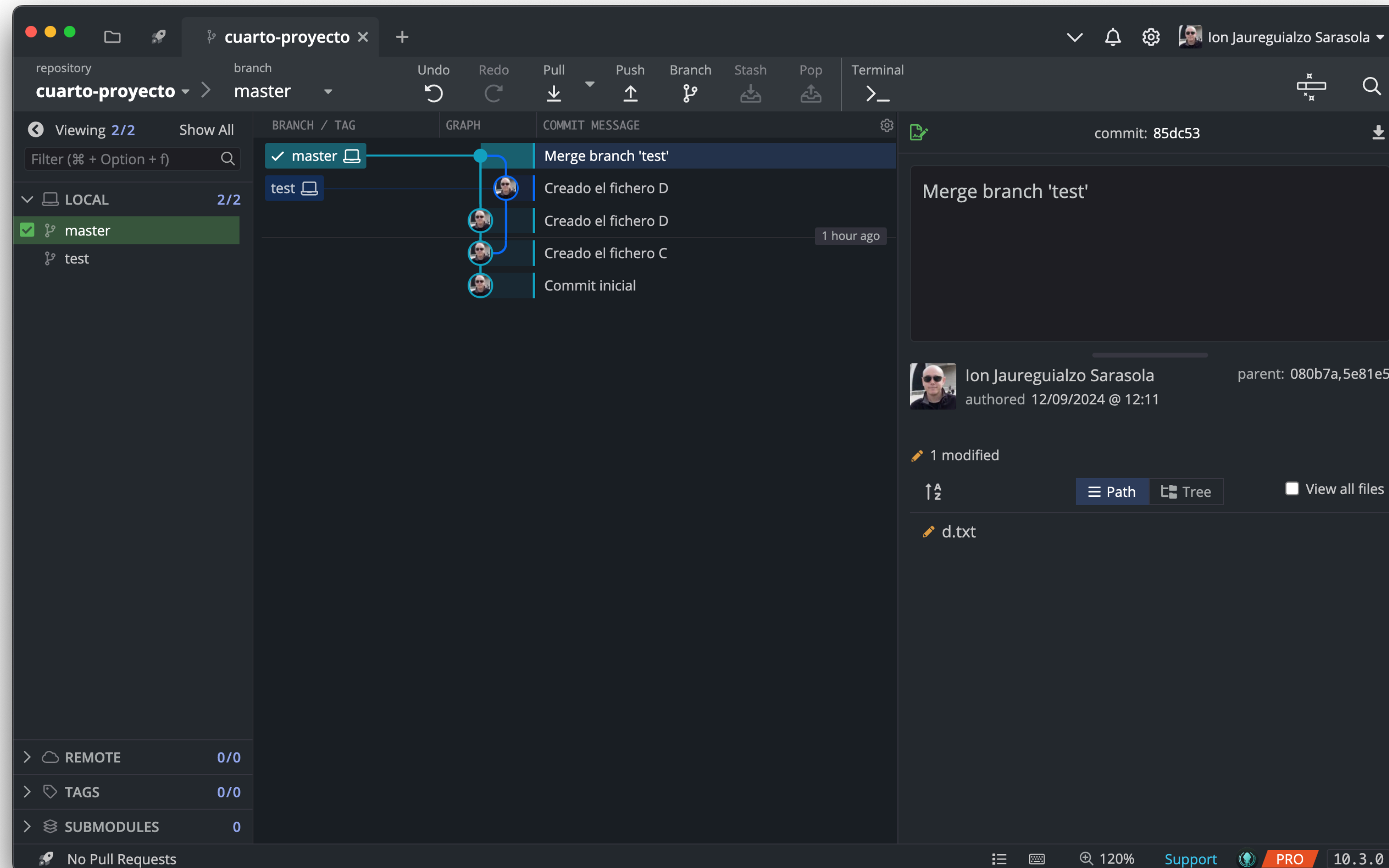
```
<<<<<< HEAD
Contenido del fichero D
=====
Nuevo contenido del fichero D
>>>>>> test
```

# Solucionar el conflicto

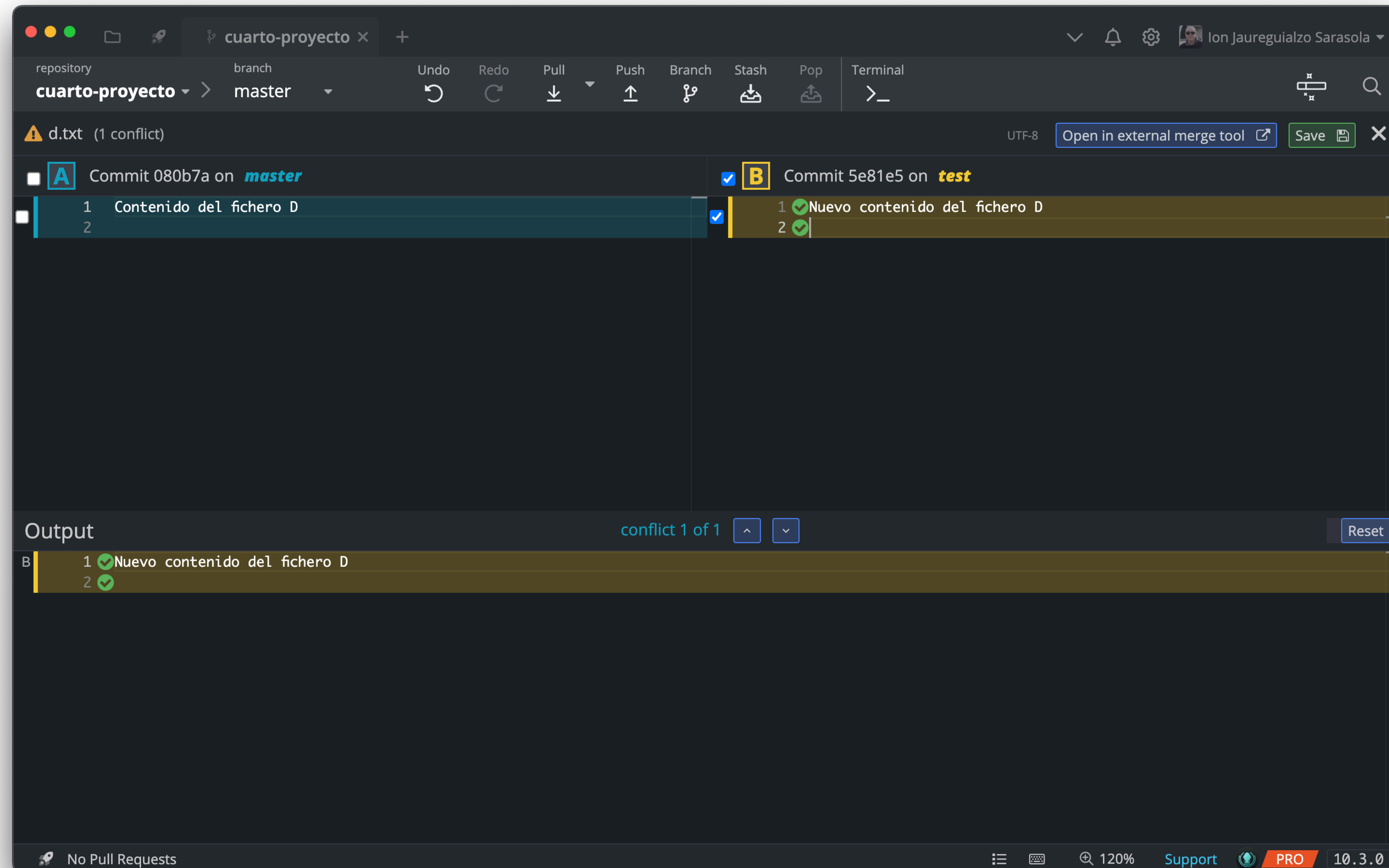
```
git add d.txt  
git commit
```

```
[master 85dc53c] Merge branch 'test'
```

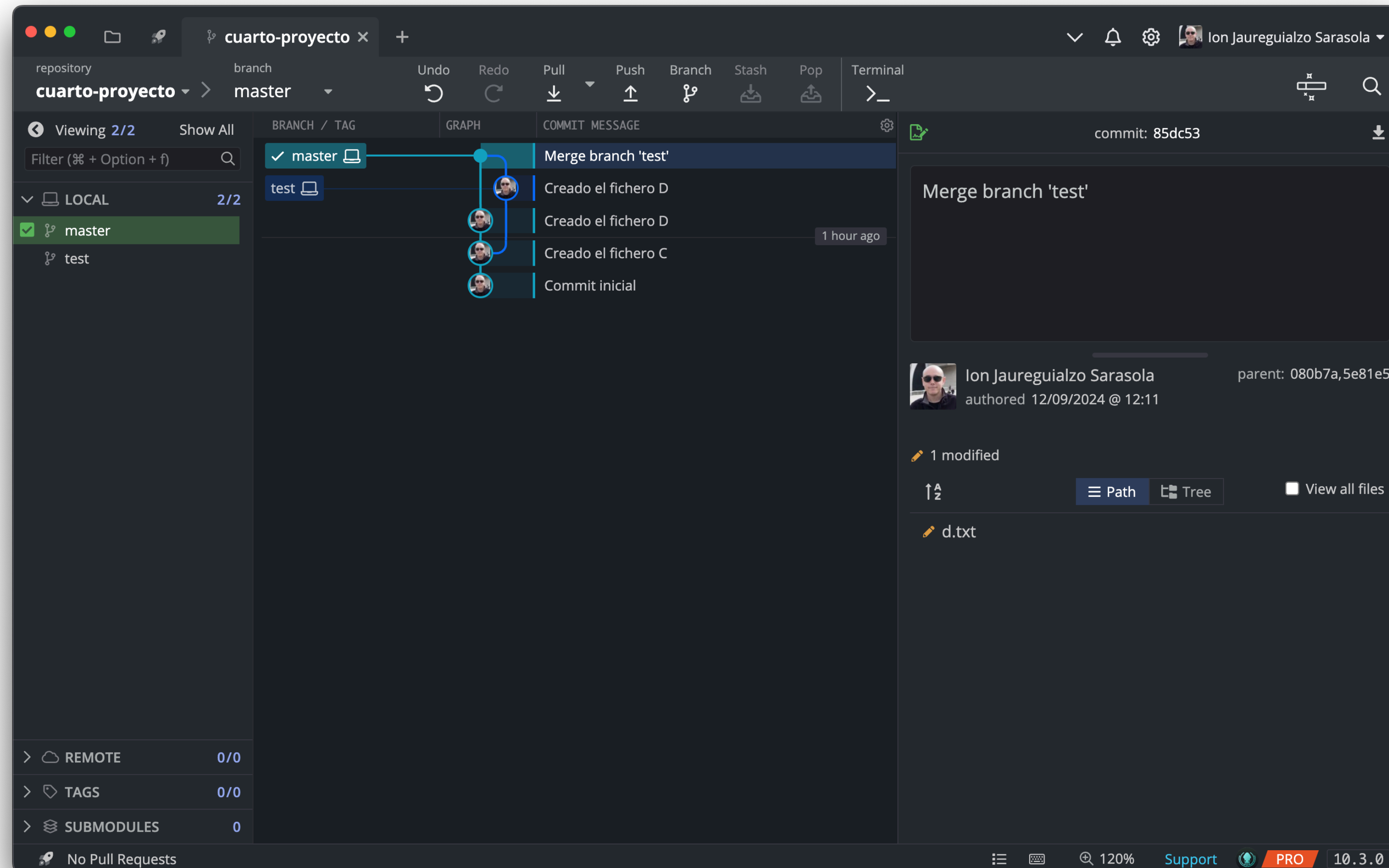
# Mostrar la línea de tiempo



# Solucionar el conflicto

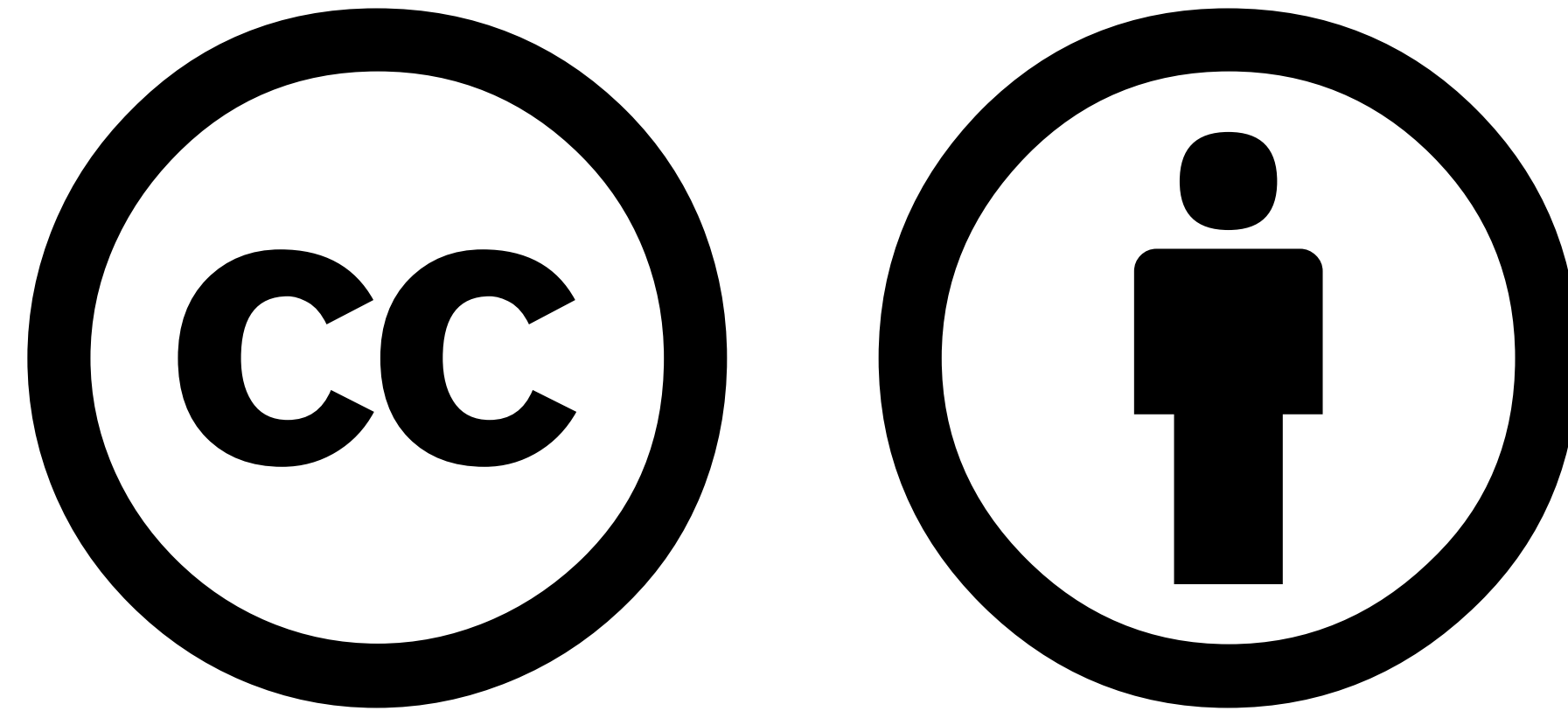


# Mostrar la línea de tiempo



# Seguir practicando con ramas

<https://learngitbranching.js.org>



Excepto si se especifica lo contrario, esta presentación está bajo licencia

**<https://creativecommons.org/licenses/by/4.0/>**

© 2026 Ion Jaureguialzo Sarasola. Algunos derechos reservados.