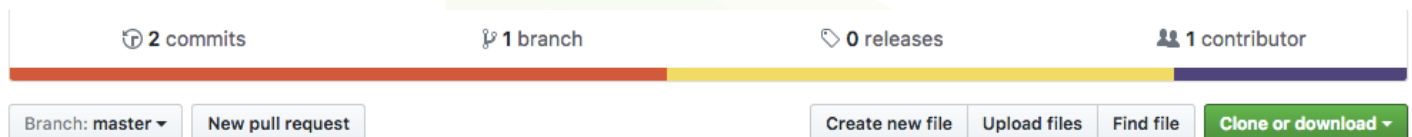


## Ejercicios del tema 8

**Ejercicio 1** - Vamos a crear una aplicación con React y subirla a GitHub. Primero de todo deberás crear el repositorio en GitHub tal y como has visto en las transparencias del tema. Una vez creado, escribiremos en la consola la orden siguiente:

```
git clone https://github.com/user/github.git
```

La url la puedes encontrar en la página principal del repositorio, al pulsar sobre el botón “Clone or download”. En la captura de pantalla siguiente se muestra dicho botón.

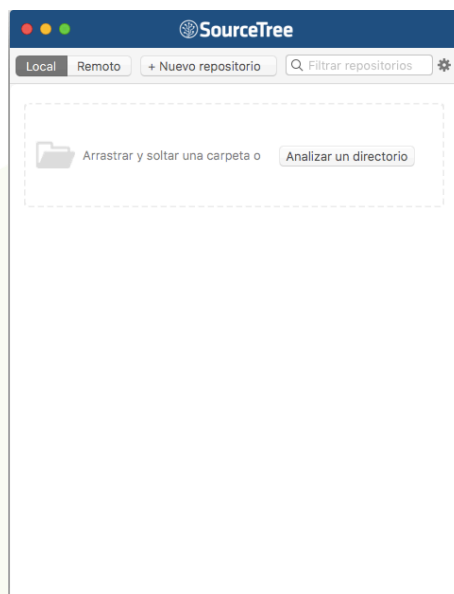


Una vez clonado el proyecto, ejecutaremos la orden **\$ create-react-app nombre** donde “nombre” será el nombre del repositorio. A continuación navegaremos hacia la carpeta creada y realizaremos la serie de órdenes correspondientes para subir nuestra aplicación a la nube. A partir de aquí, ejecuta las órdenes vistas en las transparencias para poder actualizar el repositorio en GitHub (commit, push, etc).

Como entregable del tema, sube un archivo .zip que contenga una o más capturas de la consola que muestren cómo has ejecutado los comandos y un archivo de texto con el enlace al repositorio.

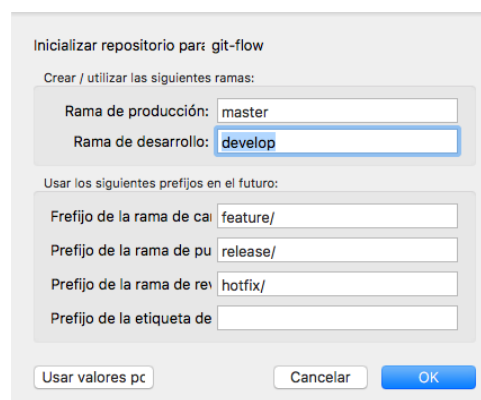
**Ejercicio 2** - Una manera más cómoda de trabajar con git es usando la herramienta **SourceTree**. Esta herramienta nos proporciona una visión gráfica de lo que es el funcionamiento de los repositorios git. Permite tener un mejor control de las diferentes ramas del proyecto y aplicar rápidamente los cambios que se realicen.

Podemos descargarla de <https://www.sourcetreeapp.com/>. Una vez descargada e instalada (es posible que tengas que crearte una cuenta), vamos a empezar a utilizar la herramienta.

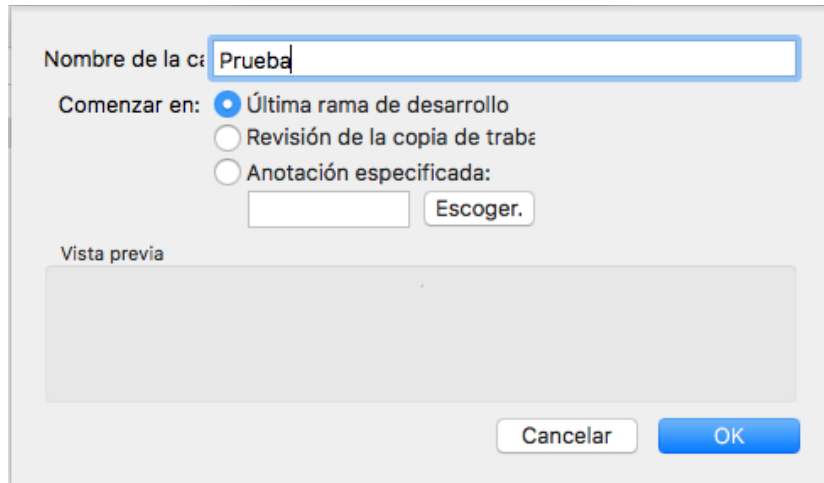


Esta es la pantalla inicial de SourceTree. Continuando con el repositorio del ejercicio 1, vamos a añadirlo como nuevo directorio para la aplicación. Una vez añadido, hacemos doble clic sobre él y se nos abrirá la ventana principal para trabajar con el repositorio.

Lo primero que debemos hacer es inicializar el repositorio. Para ello vamos a **Repositorio - git flow / hg flow - inicializar repositorio** y aceptamos.

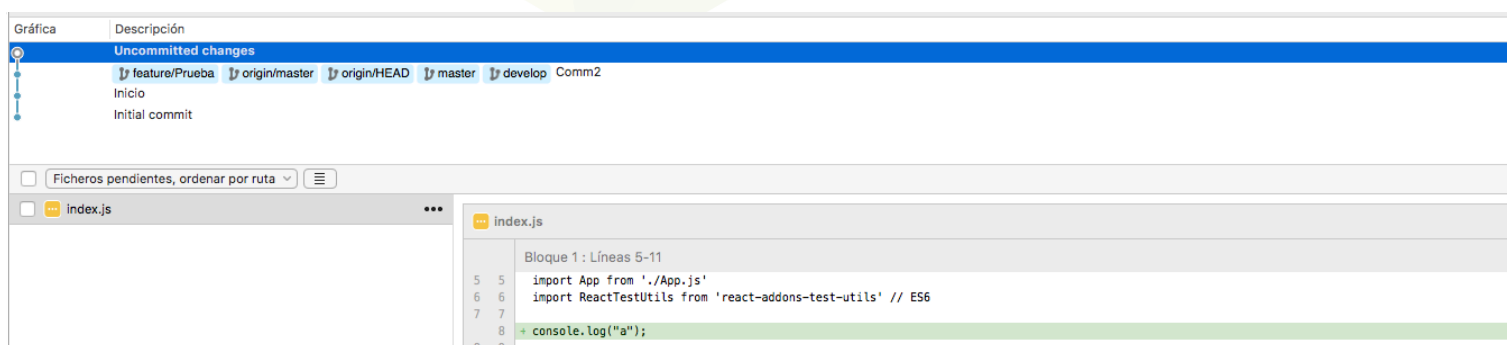


A partir de aquí vamos a ver cómo crear diferentes ramas del proyecto y fusionarlas más adelante. Nos vamos a la pestaña **Repositorio - git flow / hg flow - siguiente acción**. Entre las acciones recomendadas, le daremos a “Iniciar una nueva característica”. A esta nueva característica la llamaremos como queramos.



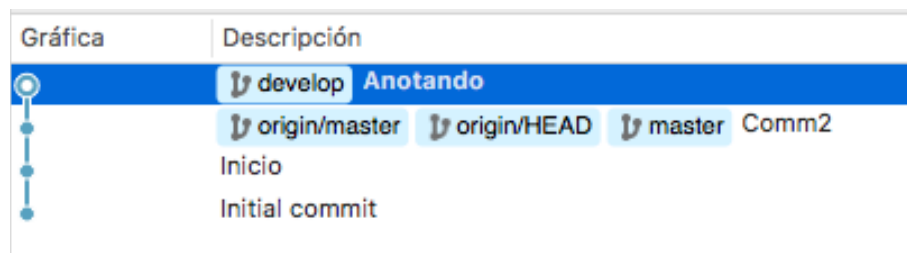
Ahora estaremos trabajando con la característica Prueba. Todos los cambios que hagamos se guardarán en esa característica. Modifica el contenido de cualquier archivo del proyecto para continuar con el siguiente paso.

Una vez realizado el cambio, en **Uncommitted changes** aparecerán los ficheros que hemos modificado:

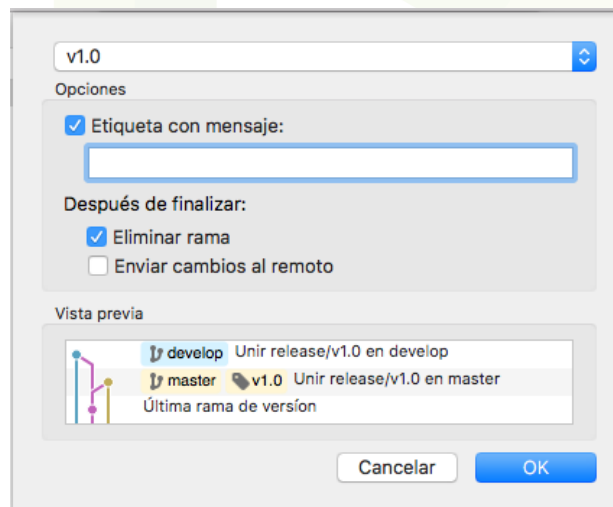


A continuación le damos a **Anotar** para subir los cambios. Elegimos el mensaje que queramos para nuestro commit y aceptamos.

Volvemos a **Repositorio - git flow / hg flow - siguiente acción** y le daremos a Finalizar actual, de manera que eliminaremos la rama de prueba que habíamos creado y la fusionaremos con la rama **develop**. Mediante esta técnica podemos realizar cambios en un proyecto sin sobrescribir las versiones anteriores, para luego más tarde quedarnos con la versión que más nos interese.



Finalmente, para fusionarlo con la rama principal **master**, iremos de nuevo a *siguiente acción* y le daremos a “iniciar una nueva versión”. Esta la llamaremos “v1.0” o con el nombre que queramos darle. Una vez creada, volvemos a *siguiente acción* y le damos a finalizar:



De esta manera, todos los cambios que hemos realizado se han volcado sobre la rama principal. Resulta muy útil poder crear otras ramas para probar funcionalidades de la aplicación y luego descartarlas o fusionarlas según nos interese. Como presentación del ejercicio haz una captura de pantalla final de todo el proceso de las ramas.

