



# Git

## Sistema de Control de Versiones

---

Ivana Rojas Córscico

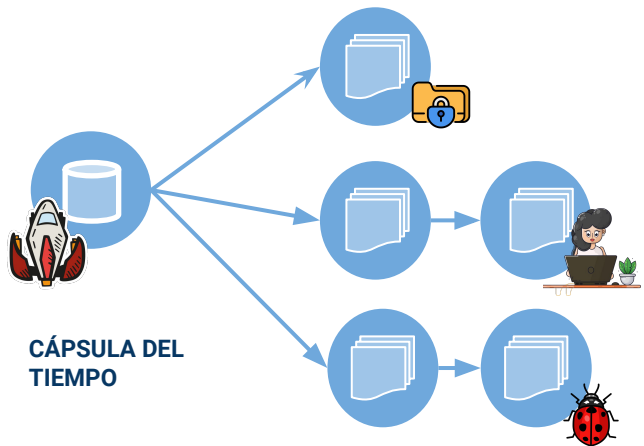
¿Qué es el **código fuente**?



# ¿Qué es el **control de versiones**?

## CONTROL DE CÓDIGO FUENTE

Práctica que permite registrar y gestionar los cambios en el código fuente de un proyecto a lo largo del tiempo.

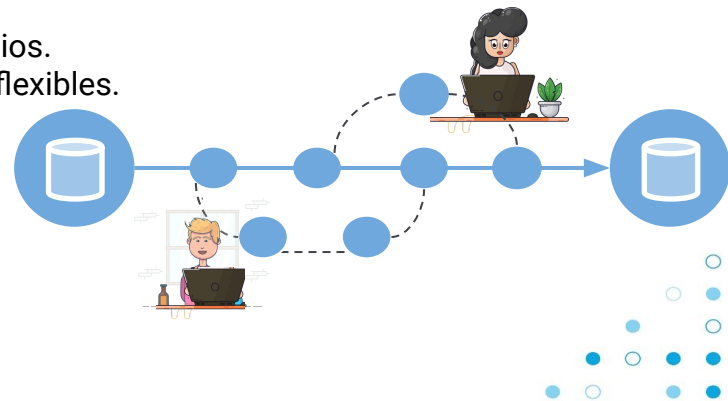


## SISTEMAS DE CONTROL DE VERSIONES

Son **herramientas** de software que ayudan a los equipos de desarrollo a gestionar los cambios en el fuente a lo largo del tiempo.

Proveen:

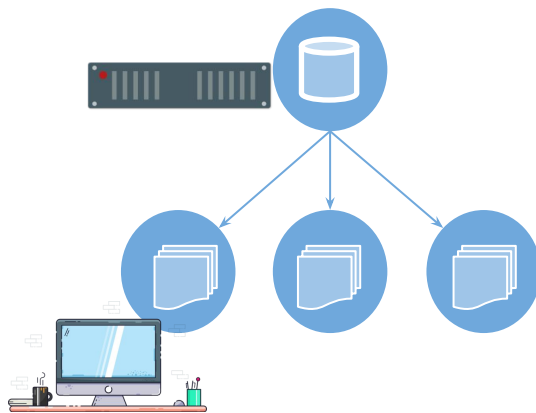
- ▶ Historial de cambios.
- ▶ Flujos de trabajo flexibles.
- ▶ Trazabilidad.



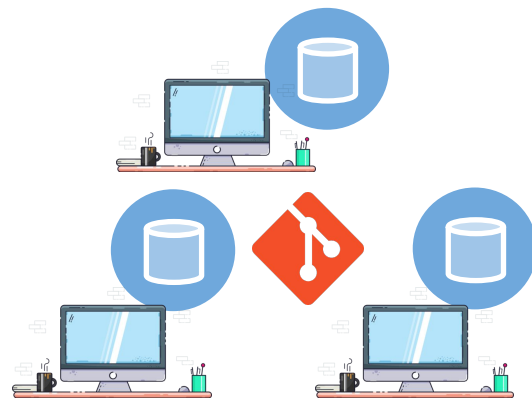
# ¿Qué es **Git**?

## **GIT**

Git es un **sistema de control de versiones distribuido** gratuito y de código abierto diseñado para manejar desde proyectos pequeños a muy grandes, con velocidad y eficiencia.



**REPOSITORIO CENTRALIZADO**



**REPOSITORIO DISTRIBUIDO**



# ¿Cómo instalar **Git**?

## 1. Instalar Git en tu PC.

## 2. Configurar tu identidad en Git (opcional)

```
git config --global user.name "Mi Usuario"
```

```
git config --global user.email mi\_correo\_electronico@example.com
```

## 3. Comprobar la configuración

```
git config --list
```

```
git --version
```



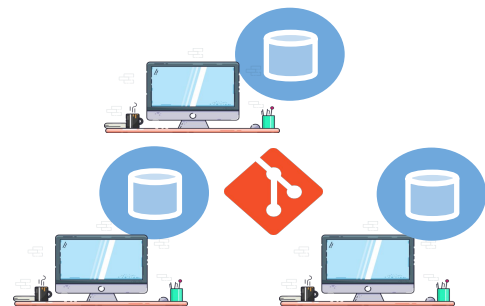
## Repositorios en Git

Un repositorio es un directorio dónde tienes alojado tu proyecto permitiéndote además, guardar **versiones del código** a las que puedes acceder cuando lo necesites.



Algunas consideraciones de Git en cuanto a repositorios:

- ▶ Cada repositorio pertenece a una cuenta de usuario o equipo.
- ▶ El propietario del repositorio es la única persona que puede eliminarlo. Si el repositorio pertenece a un equipo, lo puede eliminar un administrador.
- ▶ Un proyecto puede estar formado por uno o varios repositorios.



## Crear un nuevo repositorios en Git

Es tan simple como moverte al repositorio y ejecutar el comando:

```
git init
```



- ▶ Al ejecutar este comando, se creará un nuevo subdirectorio **.git** en tu directorio y, una nueva rama principal (Ej. **main**, **master**, etc).
- ▶ Este comando se utiliza una sola vez.
- ▶ Puede utilizarse para convertir un proyecto existente en un repositorio de Git, o para inicializar un nuevo repositorio vacío.



## Clonar un repositorios en Git

Si deseas una copia de un repositorio existente, puedes ejecutar el comando:



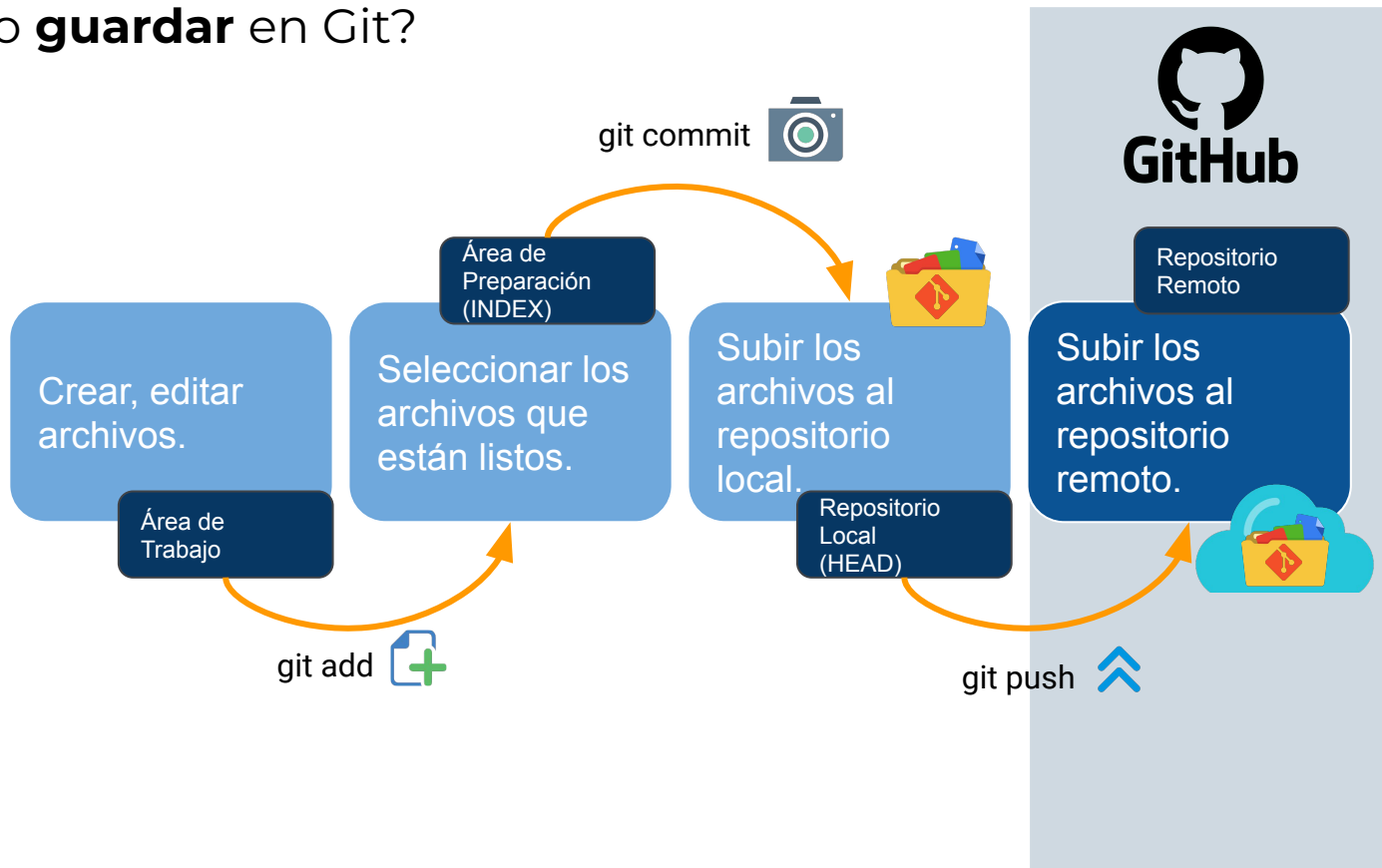
```
git clone <repo url>
```

- ▶ Al ejecutar este comando, éste ejecutará de manera interna primero git init para crear el repositorio, luego copiará los datos del repositorio existente y por último los archivos.

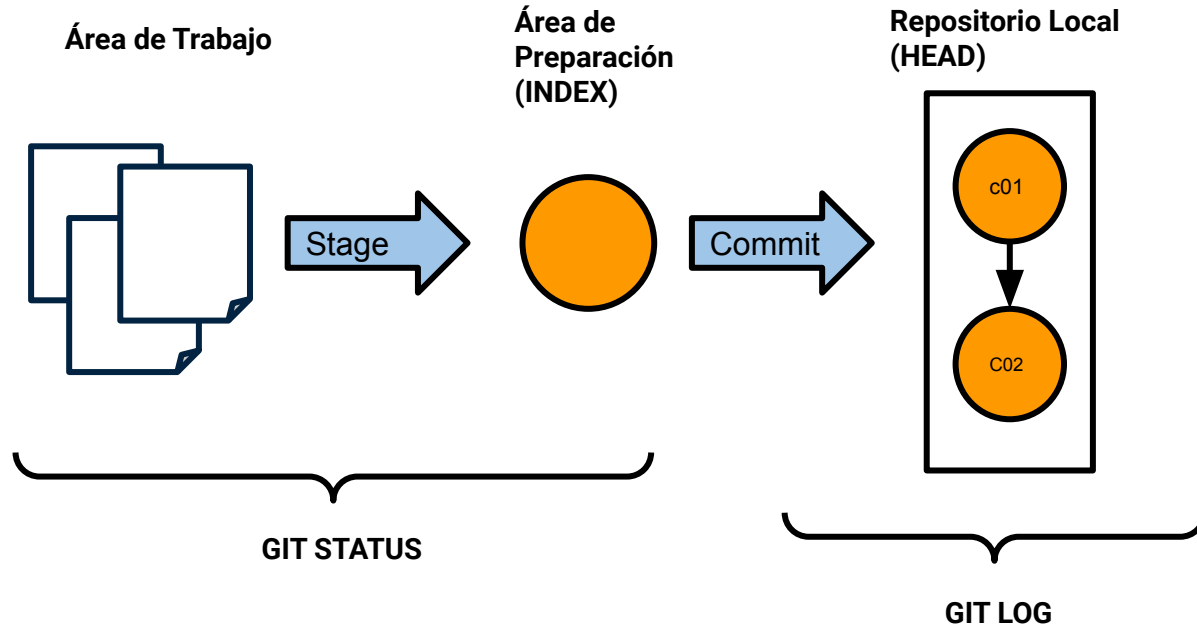




## ¿Cómo **guardar** en Git?



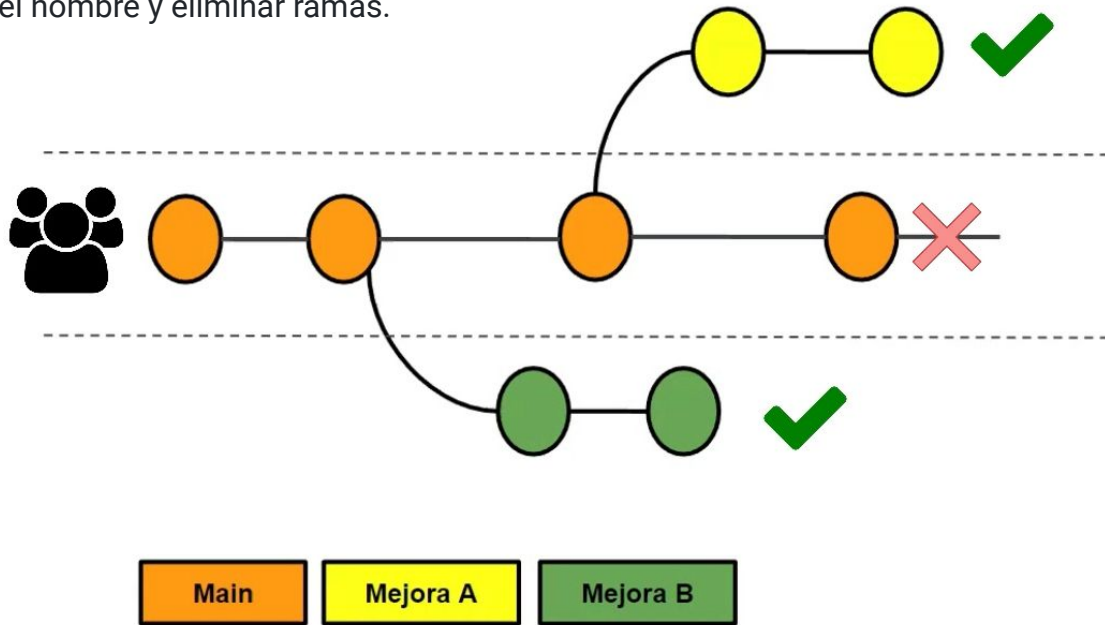
## ¿Cómo **inspeccionar** los cambios en Git?



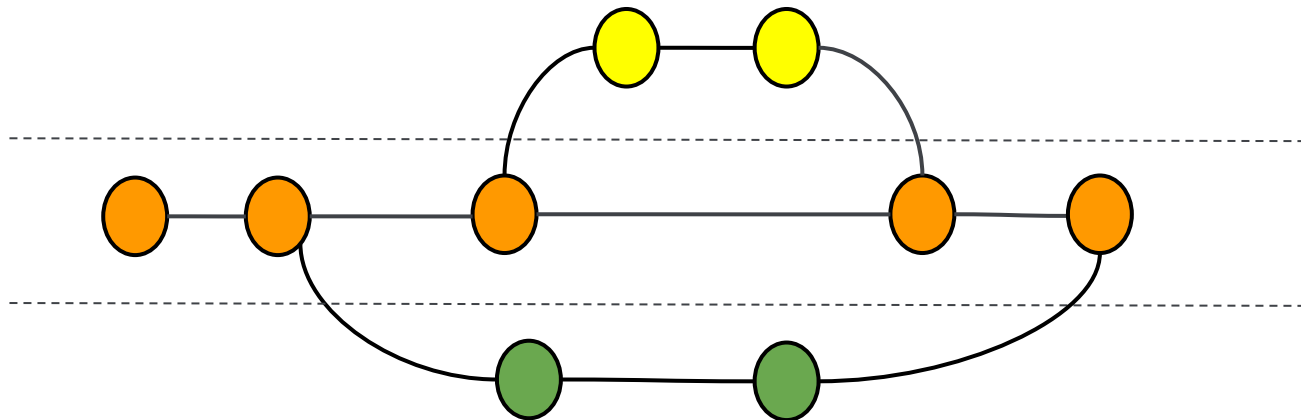
## ¿Cómo trabajar con **ramas** en Git?

Para ello, Git nos provee los siguientes comandos:

- **git branch**, permite crear, enumerar, cambiar el nombre y eliminar ramas.
- **git checkout**, permite cambiar de rama.
- **git merge**, permite fusionar ramas.



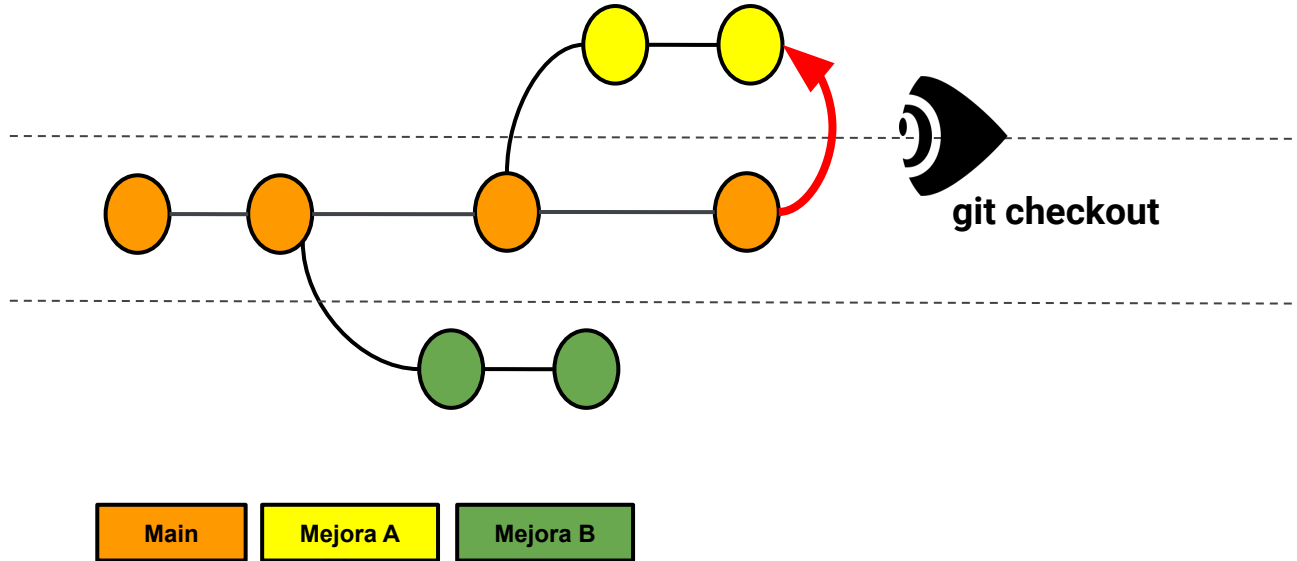
## Manipular ramas en Git



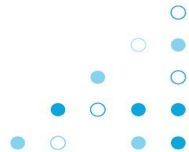
- `git branch -a` //Listar ramas
- `git branch <new branch name>` //Crear nueva rama
- `git branch -m <old branch name> <new branch name>` //Modificar el nombre de la rama
- `git branch -d <delete branch name>` //Eliminar rama



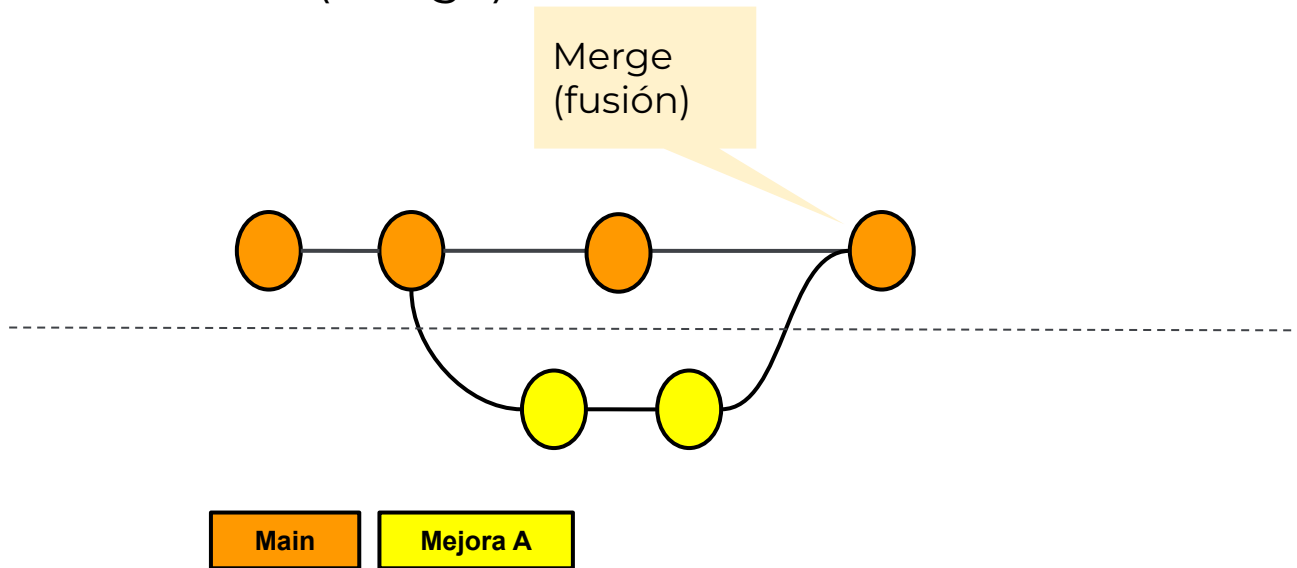
## Cambio de rama (checkout)



- `git checkout <branch name>`
- `git checkout -b <branch name>`



## Fusionar ramas (merge)



- `git merge <branch name>`
- `git merge <branch origin> <branch destination>`



# Git: la guía sencilla



<https://rogerdudler.github.io/git-guide/index.es.ht>



# Resumen de comandos

[https://github.github.com/training-kit/downloads/es\\_ES/github-git-cheat-sheet/](https://github.github.com/training-kit/downloads/es_ES/github-git-cheat-sheet/)



## Instalar Git

GitHub proporciona clientes de escritorio que incluyen una interfaz gráfica de usuario para las acciones más comunes que se pueden realizar en un repositorio y una edición de Git en la línea de comandos actualizada automáticamente para escenarios más avanzados.

### GitHub para Windows

[windows.github.com](https://windows.github.com)

### GitHub para Mac

[mac.github.com](https://mac.github.com)

Distribuciones de Git para sistemas Linux y POSIX se encuentran disponibles en el sitio web oficial Git SCM.

### Git para todas las plataformas

[git-scm.com](https://git-scm.com)

## Configurar herramientas

Configura la información del usuario para todos los repositorios locales

## Efectuar cambios

Revisa cambios y crea un commit

```
$ git status
```

Enumera todos los archivos nuevos o modificados de los cuales se van a guardar cambios

```
$ git diff
```

Muestra las diferencias entre archivos que no se han enviado aún al área de espera

```
$ git add [file]
```

Guarda el estado del archivo en preparación para realizar un commit

```
$ git diff --staged
```

Muestra las diferencias del archivo entre el área de espera y la última versión del archivo

```
$ git reset [file]
```

Mueve el archivo del área de espera, pero preserva su contenido

```
$ git commit -m"[descriptive message]"
```

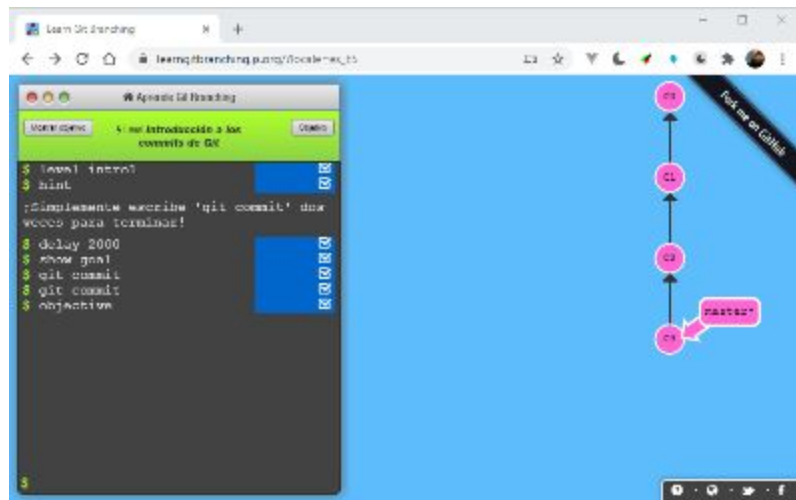
Registra los cambios del archivo permanentemente en el historial de versiones





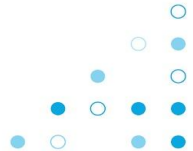
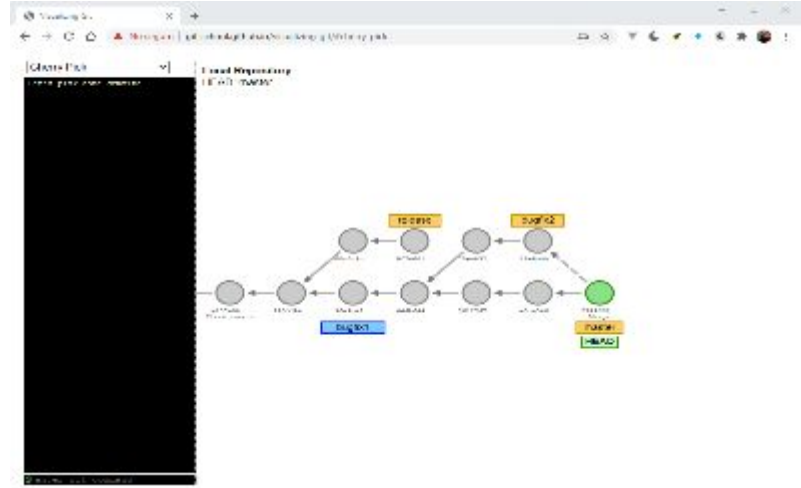
# Aprende ramas en Git de forma interactiva

[https://learngitbranching.js.org/?locale=es\\_ES](https://learngitbranching.js.org/?locale=es_ES)



## También puedes practicar ramas con...

<http://git-school.github.io/visualizing-git/#cherry-pick>



## Referencias

<https://www.atlassian.com/git/tutorials/learn-git-with-bitbucket-cloud>

<https://www.atlassian.com/git/tutorials/setting-up-a-repository>



¡Muchas gracias!