

GIT

De master a main

```
$ git config --global init.defaultBranch main  
$ git branch -m main
```

Documentacion

```
$ git - -help
```

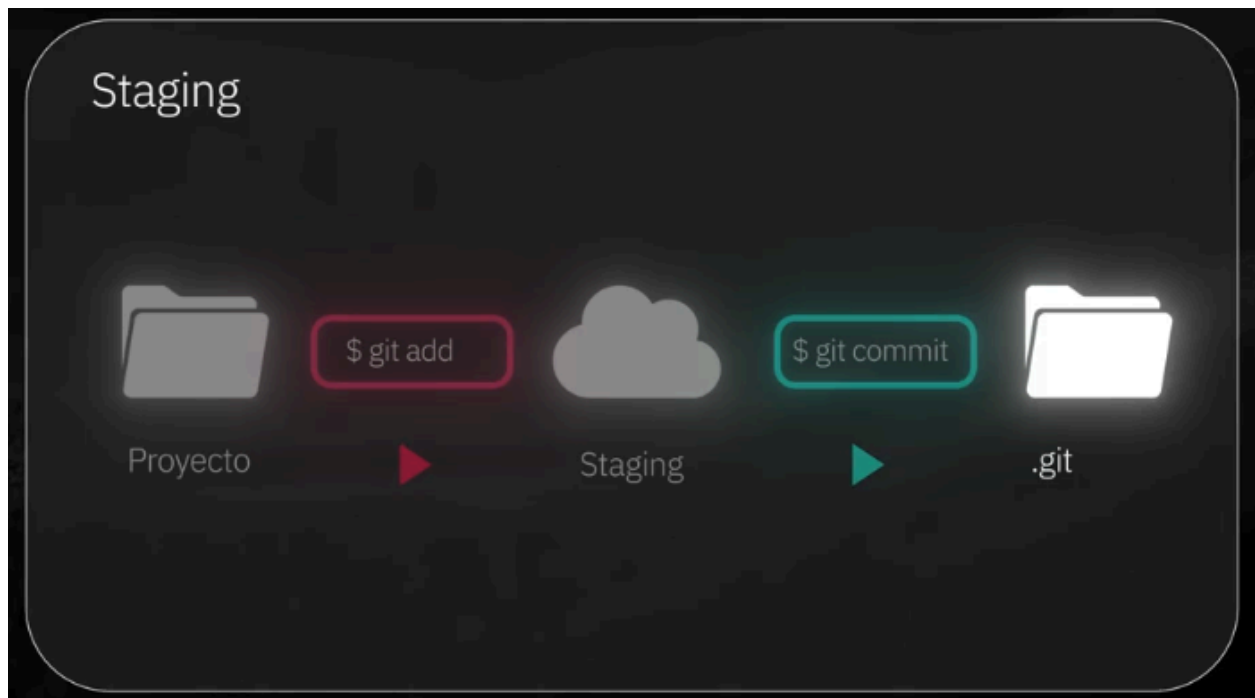
```
$ git config --global user.name "ladydn"  
$ git config --global user.email "lady.quinto123@gmail.com"
```

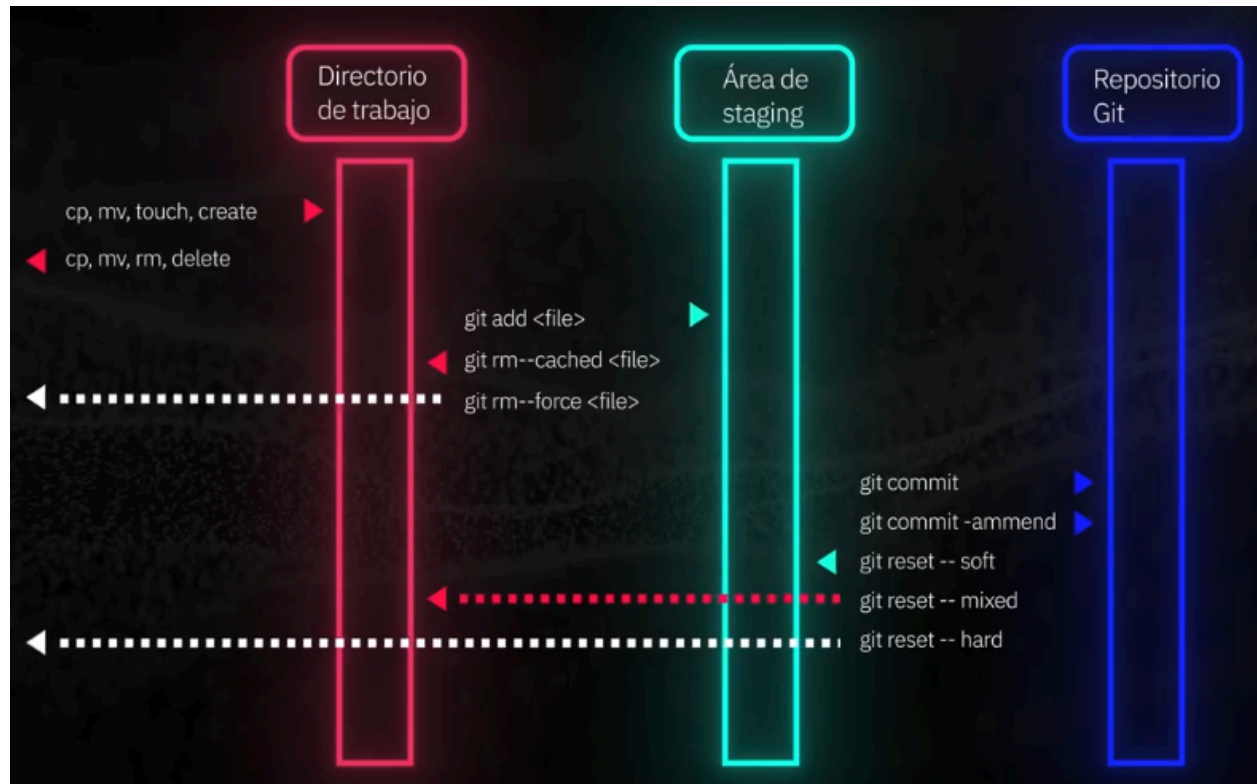
Verificar la configuración

```
$ git config --list
```

Ver bitacora

```
$ ls -a
```

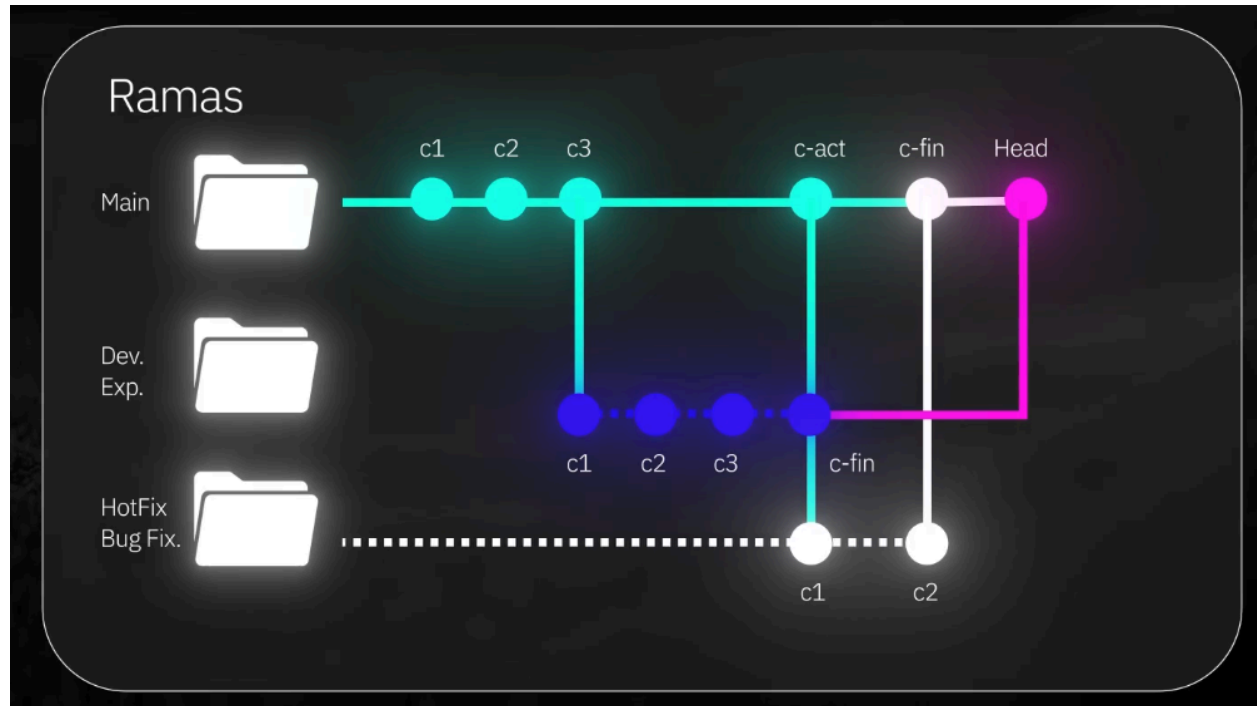




Verificar la bitácora de cambios

\$ `git log`

RAMAS

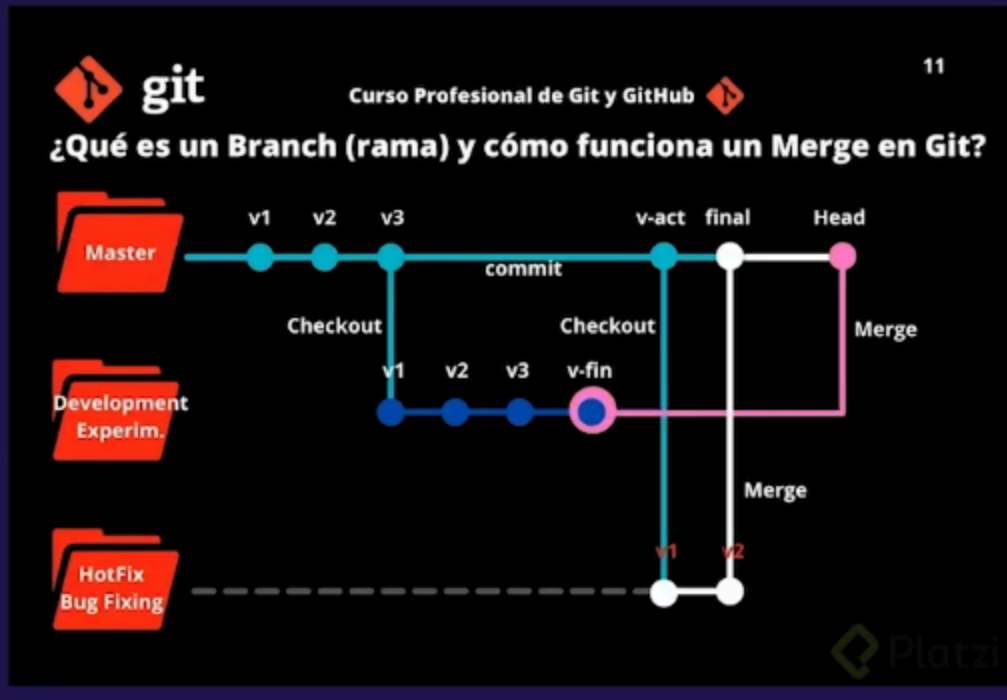


MERGE

Actualiza los cambios individuales a la rama principal.

`$ git merge nombre rama`

Ramas y merge



USOS DE CHECKOUT Y SWITCH

```
$ git checkout main
```

```
$ git switch main
```

Puedes usar cualquiera de los dos comandos para cambiar de rama.

ELIMINAR UNA RAMA

```
$ git branch -D nombre-rama
```

RESET Y REVERT

```
commit 0e4e4806d74acfe321b3b55cb3234bdc47ff1855 (HEAD
Author: Amin Espinoza <amin.espinoza@outlook.com>
Date: Mon Aug 26 10:51:58 2024 -0600

    nuevo archivo especial creado

commit 70b00436ccd84c8873862a1e50d2016836b8e7d8
Author: Amin Espinoza <amin.espinoza@outlook.com>
Date: Mon Aug 26 09:48:45 2024 -0600

    nuevo archivo creado

commit 824e1c0a8609308973f788dbcc4fc06b24773c86
Author: Amin Espinoza <amin.espinoza@outlook.com>
Date: Mon Aug 26 09:43:28 2024 -0600

    archivos actualizados

commit cbbb5002dd52ff90f97edb88f1bb3808c88a999d
Author: Amin Espinoza <amin.espinoza@outlook.com>
Date: Mon Aug 26 08:59:23 2024 -0600

    segundo commit
```

RESET: Este comando te devuelve a un commit anterior, eliminando los cambios en el historial como si nunca hubiera ocurrido.
Luego de hacer add y commit

Podemos reiniciar varios commit

```
$ git reset --hard id_commit
```

Parametros:

- Soft: permite eliminar los archivos
- Mix: regresar los commit
- Hard: permite hacer deshacer todos los cambios

REVERT: Crear un nuevo commit que "revierte" los cambios realizados por un commit específico

Luego de hacer add y commit

```
$ git revert id_commit
```

Tag y checkout

Un **tag** es como una etiqueta que puedes aplicar a un commit para identificarlo

```
$ git tag -a v1.0 -m "Mi primera version"
```

Buscar el tag, en vez de usar log

```
$ git tag
```

```
aminespinoza@CompuAmin:/mnt/c/Users/amine/Documents/Platzi/git_testing$ git log
commit 70b00436ccd84c8873862a1e50d2016836b8e7d8 (HEAD -> main, tag: v1.0)
Author: Amin Espinoza <amin.espinoza@outlook.com>
Date:   Mon Aug 26 09:48:45 2024 -0600

    nuevo archivo creado

commit 824e1c0a8609308973f788dbcc4fc06b24773c86
Author: Amin Espinoza <amin.espinoza@outlook.com>
Date:   Mon Aug 26 09:43:28 2024 -0600

    archivos actualizados
```

Mostrar el historial de tag creado

```
$ git show v1.0
```

```
aminespinoza@CompuAmin:/mnt/c/Users/amine/Documents/Platzi/git_testing$ git tag
v1.0
aminespinoza@CompuAmin:/mnt/c/Users/amine/Documents/Platzi/git_testing$ git show v1.0
tag v1.0
Tagger: Amin Espinoza <amin.espinoza@outlook.com>
Date:   Mon Aug 26 11:24:58 2024 -0600

    Mi primera version

commit 70b00436ccd84c8873862a1e50d2016836b8e7d8 (HEAD -> main, tag: v1.0)
Author: Amin Espinoza <amin.espinoza@outlook.com>
Date:   Mon Aug 26 09:48:45 2024 -0600
```

Eliminar un tag

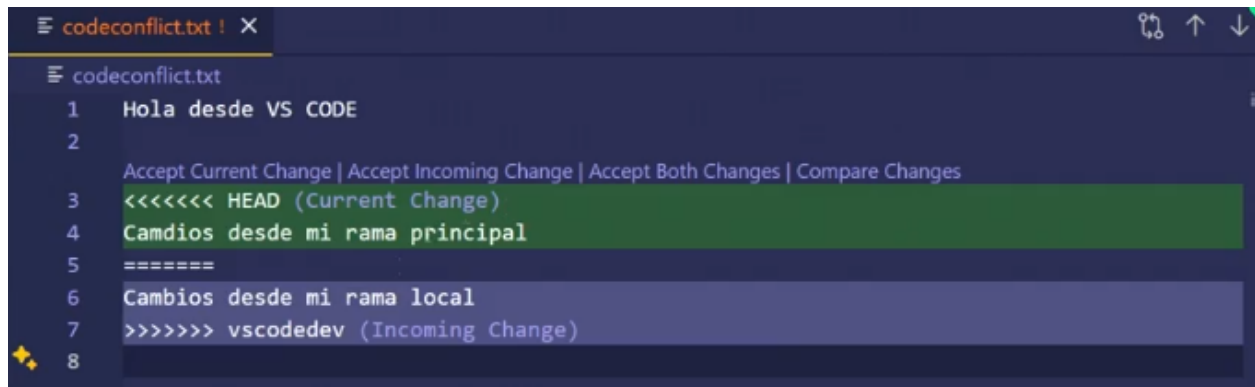
```
$ git tag -d v1.0
```

Un **checkout** permite cambiar de una rama o un commit específico a otro

CONFLICTOS

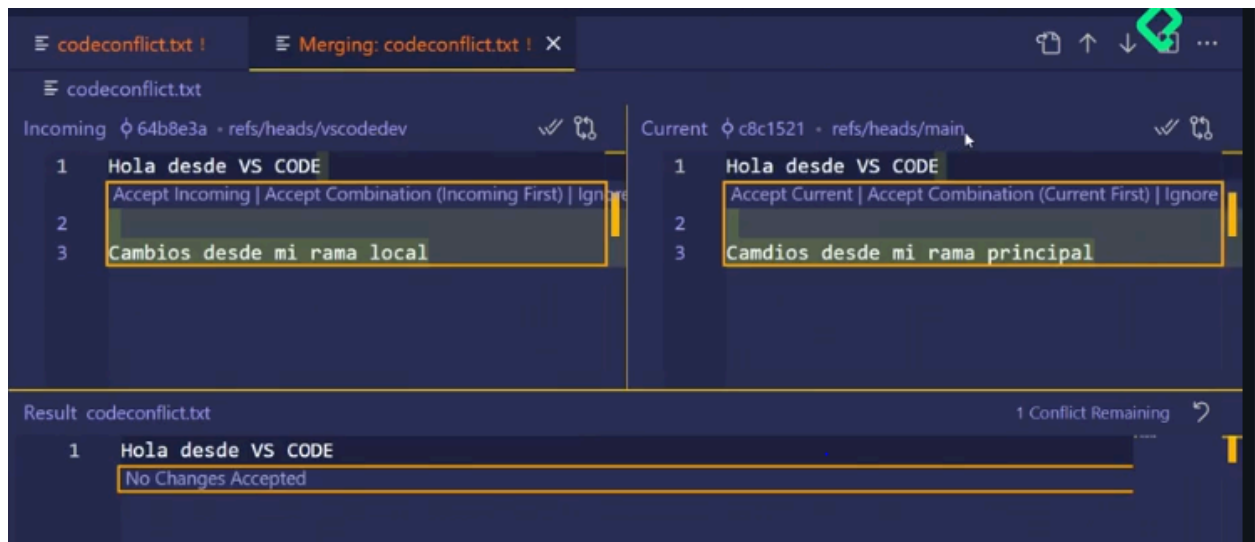
Los conflictos se generan al modificar un archivo por mas de un desarrollador. Estos conflictos se reflejan al realizar merge en el main.

La forma de solucionar estos conflictos es aceptar los cambios de otro desarrollador e incluir nuestros cambios



```
codeconflict.txt ! X
codeconflict.txt
1  Hola desde VS CODE
2
3  Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
4  <<<<<<< HEAD (Current Change)
5  Cambios desde mi rama principal
6  =====
7  Cambios desde mi rama local
8  >>>>>> vscode-dev (Incoming Change)
```

Una forma de resolver conflictos dentro de visual studio code es en el Resolve in Merge Editor.



```
codeconflict.txt !   Merging: codeconflict.txt ! X
codeconflict.txt
Incoming  64b8e3a - refs/heads/vscode-dev  ✓  🔄
1  Hola desde VS CODE
2  Accept Incoming | Accept Combination (Incoming First) | Ignore
3  Cambios desde mi rama local

Current  c8c1521 - refs/heads/main  ✓  🔄
1  Hola desde VS CODE
2  Accept Current | Accept Combination (Current First) | Ignore
3  Cambios desde mi rama principal

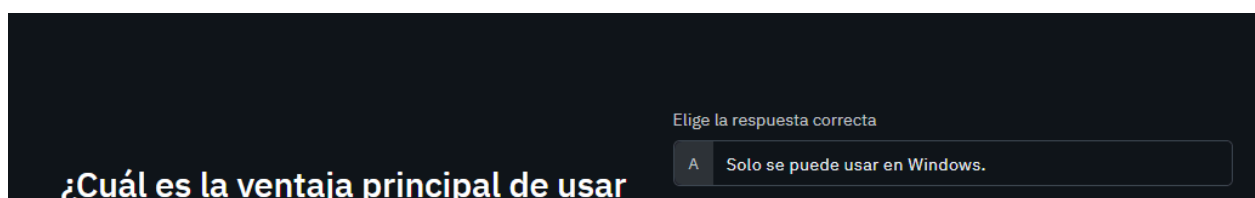
Result codeconflict.txt  1 Conflict Remaining  🔄
1  Hola desde VS CODE
   No Changes Accepted
```

GITHUB

Iniciar creando una cuenta en:

<https://github.com/>

Ojo. Añadir la autenticación en dos pasos





Agregar colaboradores en Settings, colaboradores y buscar el usuario.

Crear un nuevo repositorio

Luego de crear un nuevo repositorio podemos clonarlo con
`$ git clone https://github.com/username/Git.git`

Llaves SSH



Generar una clave SSH.

<https://docs.github.com/es/authentication/connecting-to-github-with-ssh>

Generar SSH.

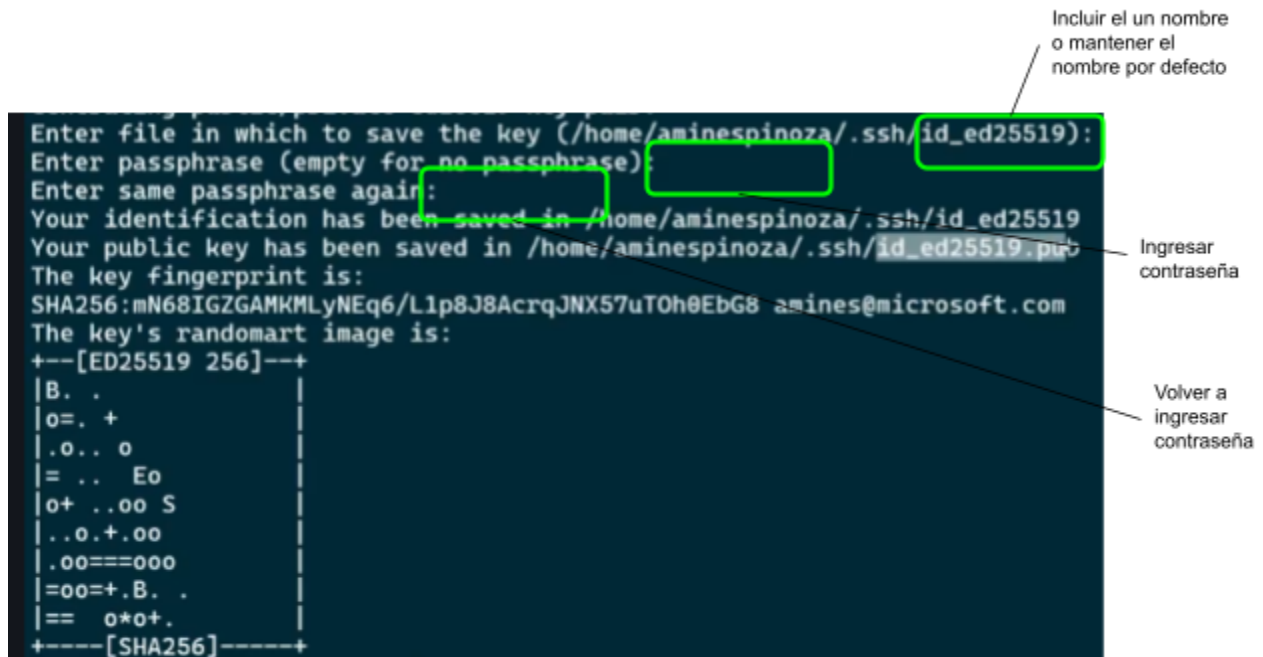
```
mpuAmin:~$ ssh-keygen -t ed25519 -C "amines@microsoft.com"
```

Correo electronico

Generar llave

Nivel de encriptacion

- Luego incluir el nombre de la llave
- Incluir la contraseña



Ahora ingresar:

```
$ eval "$(ssh-agent -s)"
```

Donde dará como respuesta Agent pid valor_agent

Siguiente paso, guardar la llave dentro del agente

```
$ ssh-add ~/.ssh/id_ed25519
```

Ojo. Incluir la contraseña antes ingresada

Abrir el archivo

```
$ nano id_ed25519.pub
```

Copiar todo el contenido de id_ed25519.pub

- Luego volver a tu perfil de github
- Ve a settings
- Buscar **SSH and GPG keys** en la izquierda de tu pantalla
- New SSH key
 - Agregar título y pegar lo copiado en key
 - Click en add SSH key

Luego agregar el comando, para establecer la conexión a github

```
$ ssh -T git@github.com
```

Al final debe mostrar esto

```
aminespinoza@CompuAmin:~/ssh$ ssh -T git@github.com
The authenticity of host 'github.com (140.82.113.3)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvV6TuJJhbpZisF/zLDA0zPMSvHdkr4UvC0qU.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
Hi aminespinoza! You've successfully authenticated, but GitHub does not provide shell access.
```

Fork y estrellas

fork



Estrella

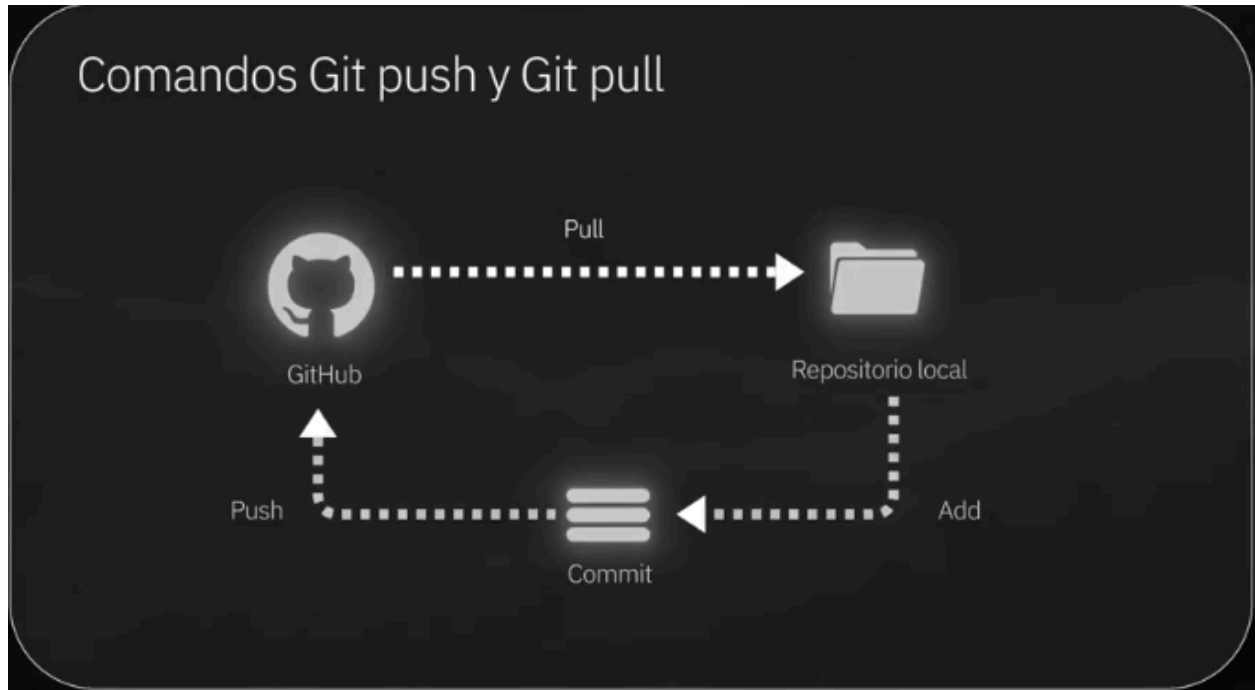
Segmenta una mayor referencia en un repositorio

Al darle un estrella a un repositorio, podrás ver de manera inmediata el repositorio al que se dio la estrella, los pasos a seguir son

- Click en perfil
- All star

Te mostrará los repositorios al se le dio las estrellas

Git pull, push y fetch



Pull: Actualiza tu repositorio local con el repositorio principal

Push: permite subir los cambios realizados del entorno local al entorno principal

Desde mi rama actual enviar

```
$ git push -u origin nombre_rama_actual
```

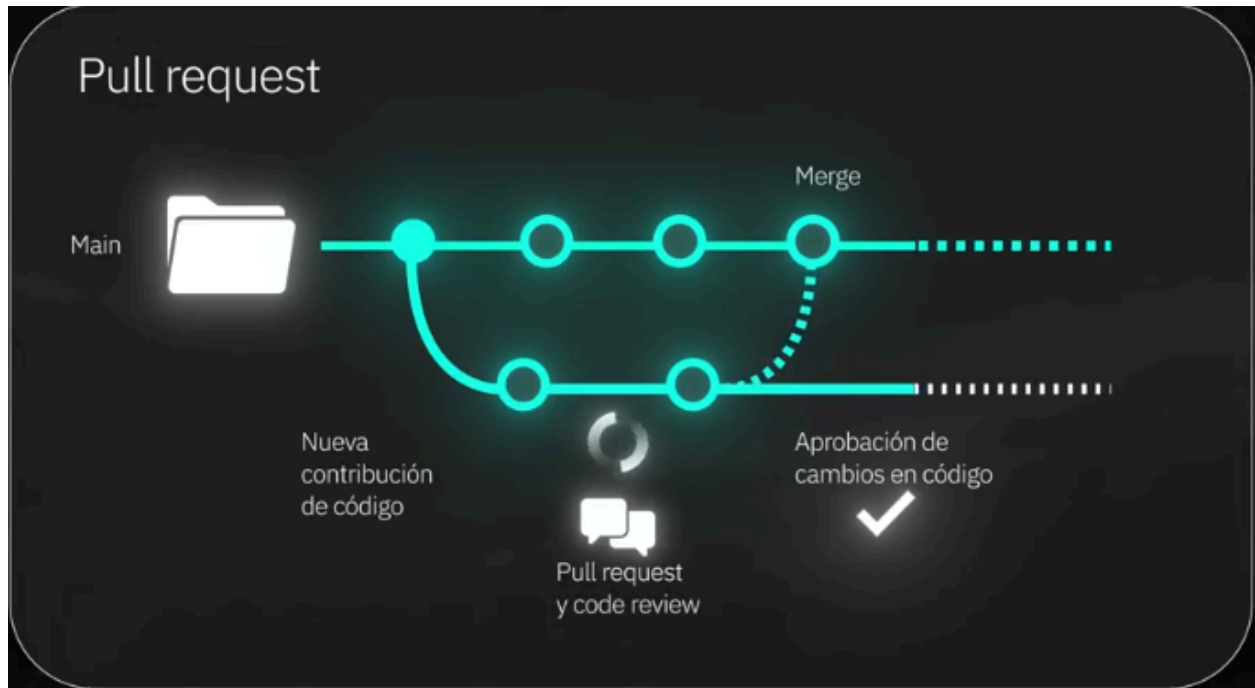
Desde la rama main bajar los cambios de la rama main

```
git pull origin main
```

Fetch:

Pull request

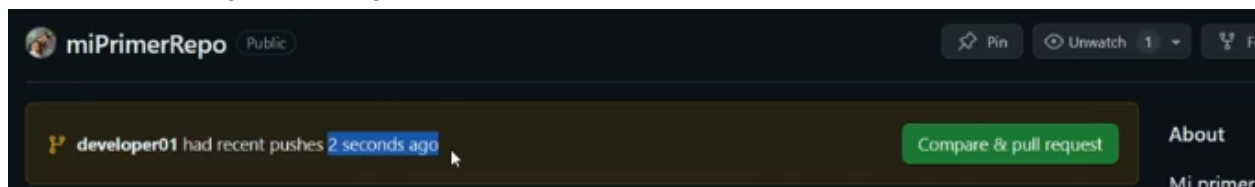
Permite fusionar ramas locales con la rama principal.



Luego de hacer

```
$ git push -u origin nombre_rama
```

Se mostrará lo siguiente en github



En esta parte un otro desarrollador debe revisar tus cambios para luego aprobarlo, es el paso del code review. El otro developer podra fusionar el pull request.

WORKFLOWS

Permite automatizar tus tareas

MARKDOWN

<https://www.markdownguide.org/>

<https://www.markdownguide.org/cheat-sheet/>

EXTENSION

<https://marketplace.visualstudio.com/items?itemName=DavidAnson.vscode-markdownlint>

Mermaid

Generar diagramas de codigo

https://mermaid.live/edit#pako:eNpVjkFrwzAMhf-K0GmD5g_kMFiTrZfCButpcQ8iUWKz2DKOTSIJ_vucdoNNJ-m97z00YysdY4n9KJdWU4hwqpWDPM9NpYOZoqXpDEXxtBw4ghXH1wX2DweBSYv3xg2Pd36_QVDNxb1jiNq4r_VuVbf8m-MF6uZIPoo_3VOF1ngpTHvOtf_d3TgnHpteip7KloKUFG4IbhDy8GS6fL786YojJotKyzz2nFPaYwKIVszSinKx9W1WMaQeldB0qAxd45TvpLvKHJtaAhkfxFP7IPE_kDrN7nAYR4

Por ejemplo al insertar

```
gitGraph
```

```
  commit
```

```
  commit
```

```
  branch develop
```

```
  checkout develop
```

```
  commit
```

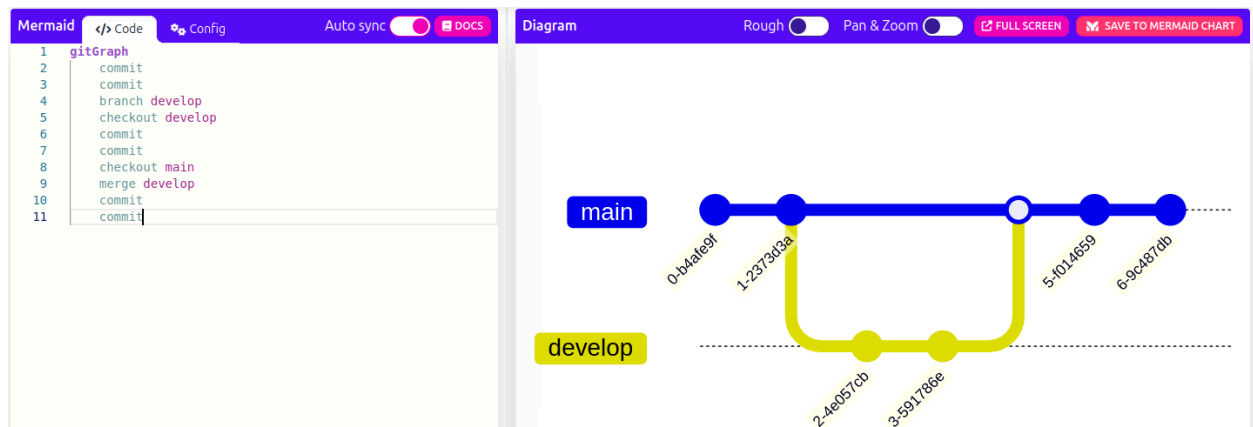
```
  commit
```

```
  checkout main
```

```
  merge develop
```

```
  commit
```

```
  commit
```



Personalizar tu perfil con Markdown

<https://github.com/platzi/git-github>

<https://github.com/tailwindlabs/tailwindcss-forms>

<https://github.com/nvm-sh/nvm>

<https://github.com/Azure-Samples/azure-search-openai-demo-csharp>

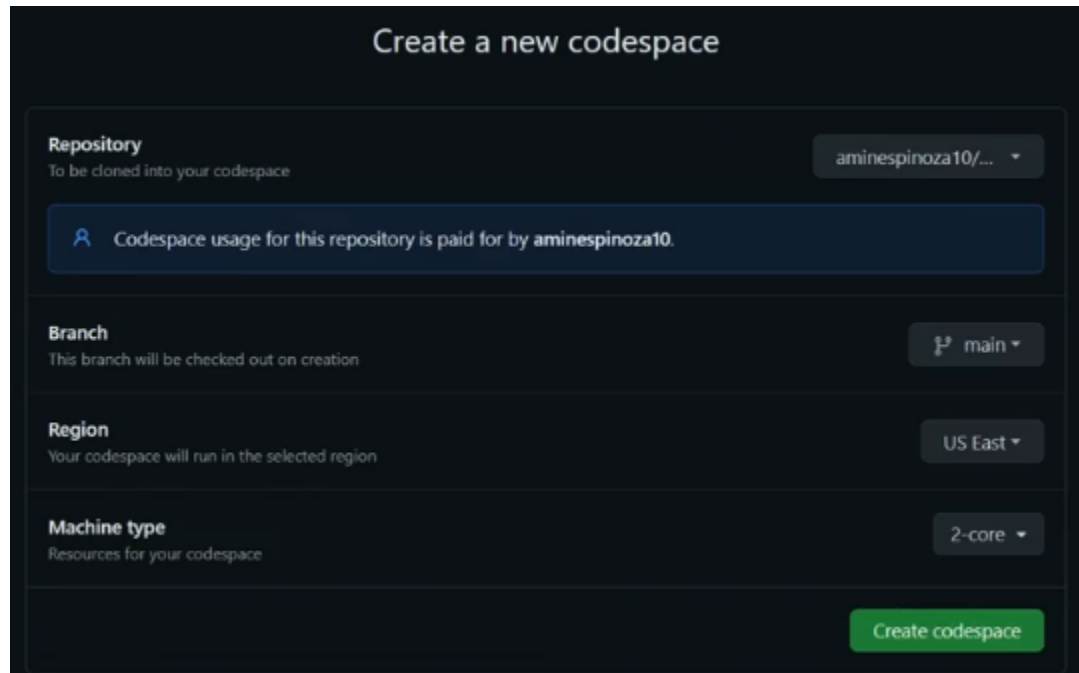
Colocate en el readme y dale click en raw para ver el texto en crudo.

<https://platzi.com/comunidad/listado-completo-de-emojis-en-markdown/>

Github pages

<https://pages.github.com/>

CODESPACES



The screenshot shows the 'Create a new codespace' interface. It has a dark theme. At the top, the title 'Create a new codespace' is centered. Below it, there are four sections: 'Repository', 'Branch', 'Region', and 'Machine type'. Each section has a dropdown menu. The 'Repository' dropdown is set to 'aminespinoza10/...'. The 'Branch' dropdown is set to 'main'. The 'Region' dropdown is set to 'US East'. The 'Machine type' dropdown is set to '2-core'. A green button labeled 'Create codespace' is at the bottom right. A message in the 'Repository' section states: 'Codespace usage for this repository is paid for by aminespinoza10.'

Ante errores

Error

como solucionar sign_and_send_pubkey: signing failed for ED25519
"/home/lady/.ssh/id_ed25519" from agent: agent refused operation
git@github.com: Permission denied (publickey).
fatal: Could not read from remote repository.

Solucion

Check SSH Agent and Key Configuration

1. Ensure SSH Agent is Running:

Start by checking if the SSH agent is running. You can do this by executing:

2. `bash`

```
eval "$(ssh-agent -s)"
```

- 3.

- 4.

5. Add Your SSH Key:

Add your SSH key to the agent using:

6. `bash`

```
ssh-add ~/.ssh/id_ed25519
```

- 7.

8. If you encounter an error stating that the agent refused the operation, it may indicate an issue with how the key is being managed by your system's keyring.