

Taller de iniciación: Git y Github

AISC x GUL UC3M



QR Slides



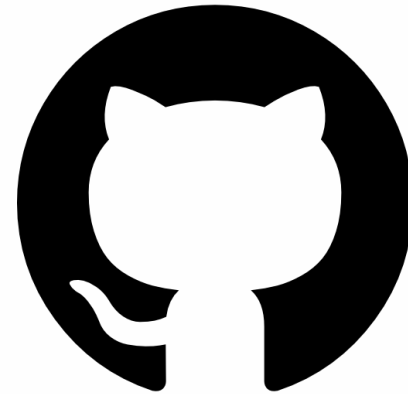
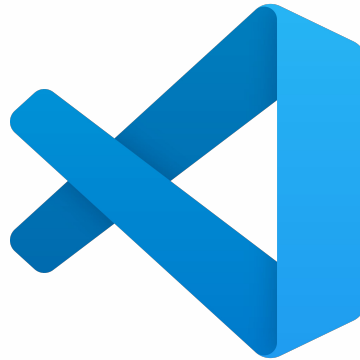
https://aiscmadrid.com/events/event_o.php?id=19



Formato taller



Formato taller



<https://code.visualstudio.com/download>

<https://github.com/>

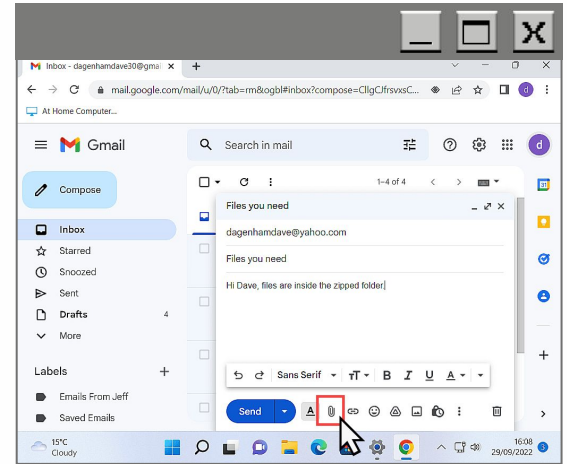
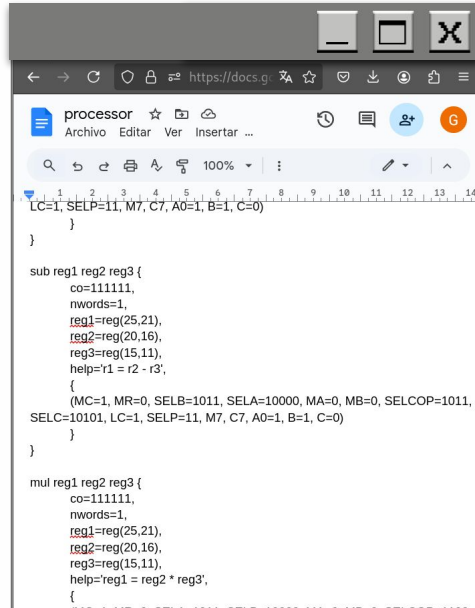
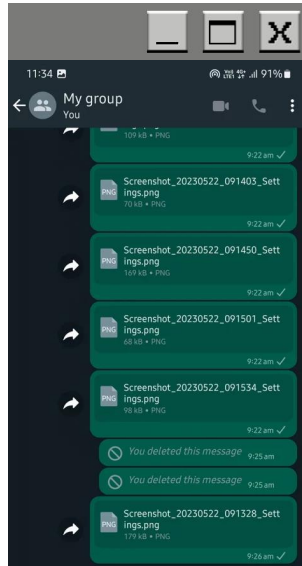


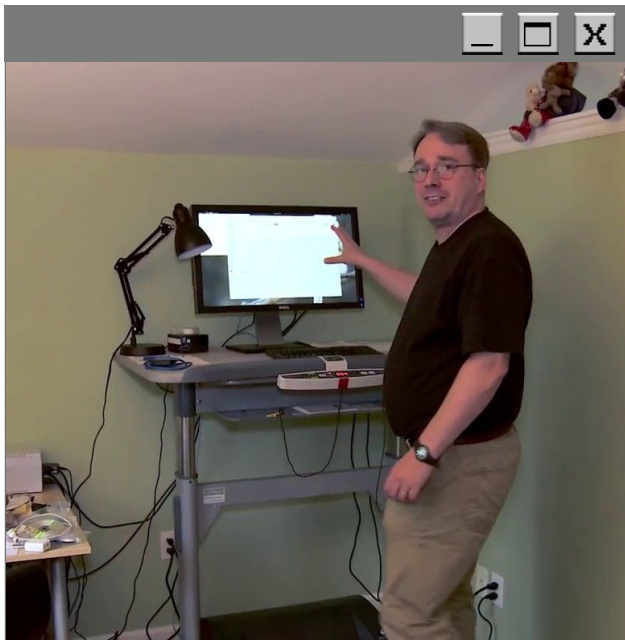
¿Por qué
usar Git?





CHOOSE YOUR STARTER





Imagina que eres este tío

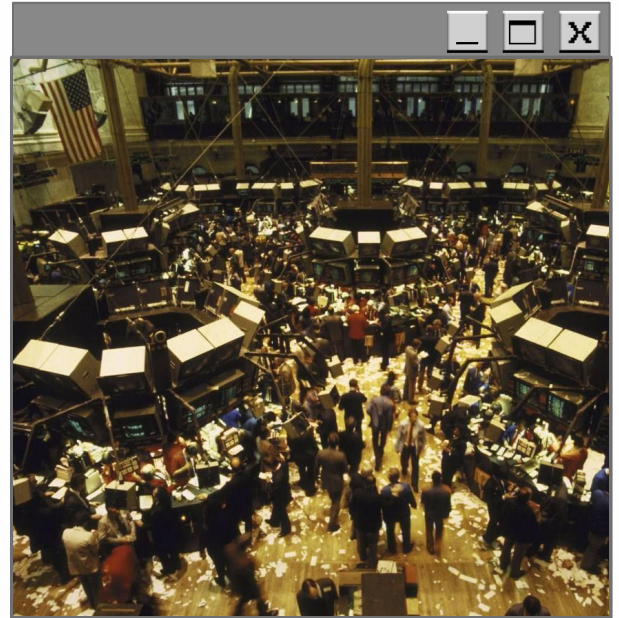
< - - - - -

Y estas creando
algo como esto

- - - - - >



According to The Linux Foundation, since 2005,
“some 14,000 individual developers from over 1,300 different companies have contributed to the kernel.”

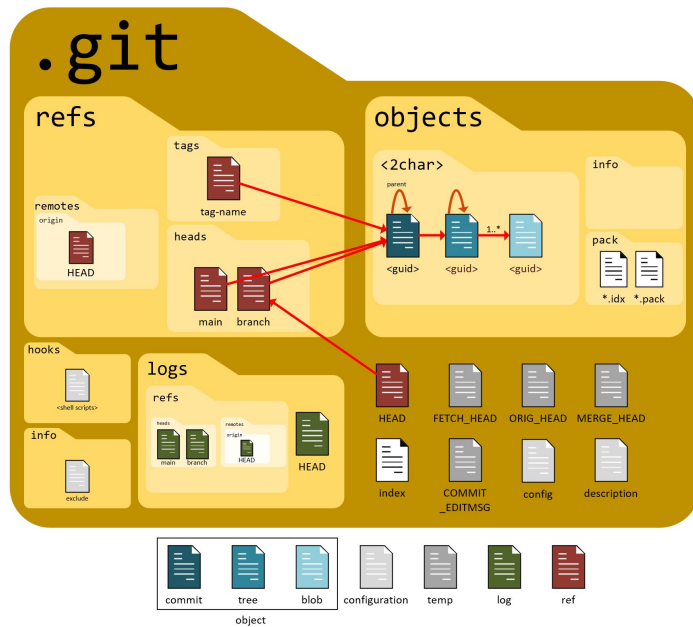




¿Como se
usa Git?



The .git folder

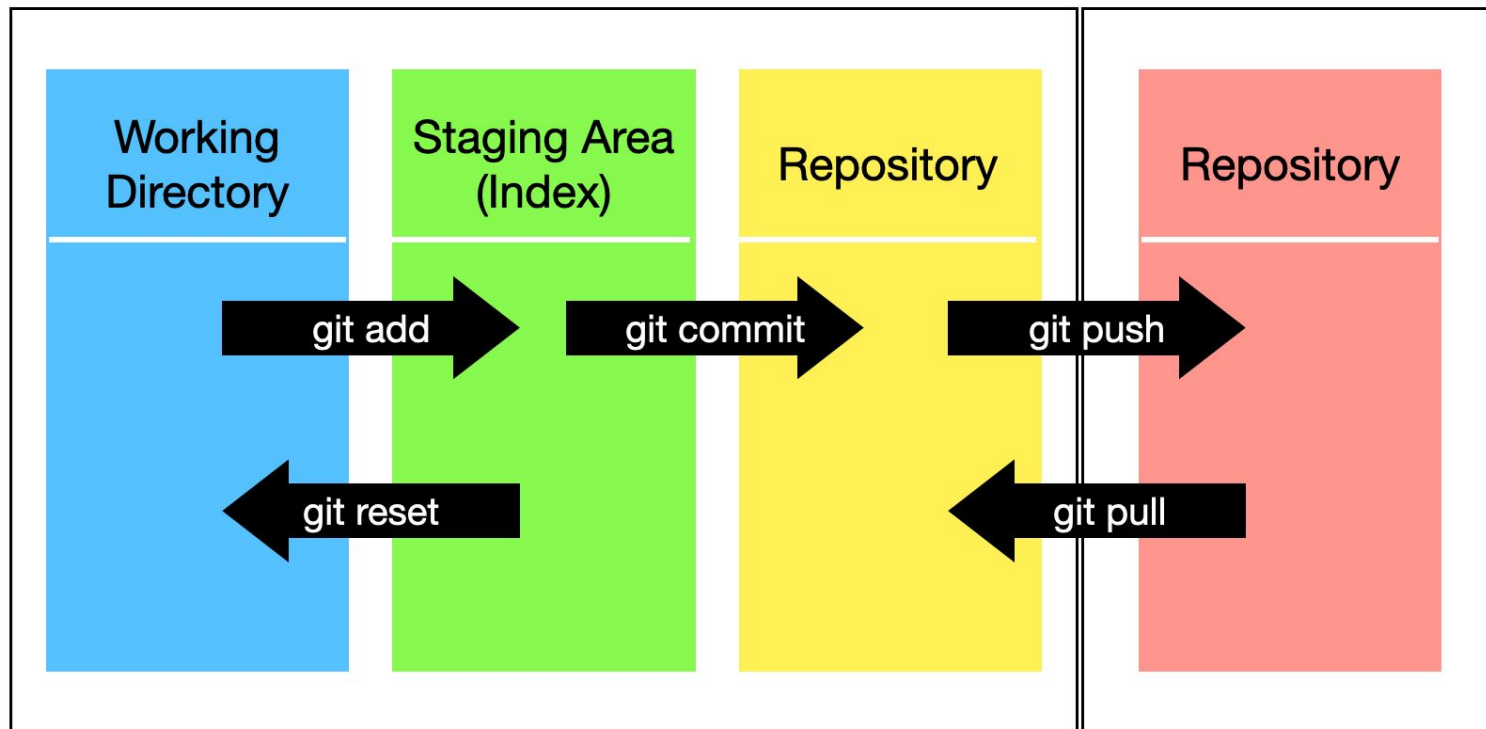


git init



LOCAL

REMOTE













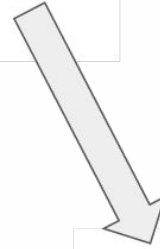
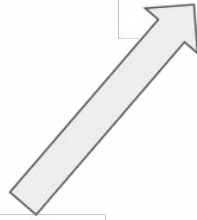




fichero.txt



```
git add  
git commit
```

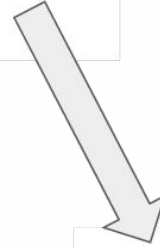
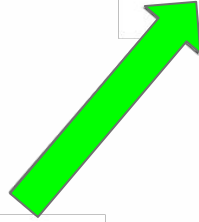




fichero.txt



fichero.txt



git push

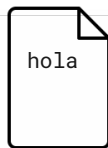




fichero.txt



fichero.txt



fichero.txt



git pull



fichero.txt



Se cambia
fichero.txt



new version



Showing 1 **changed file** with 2 additions and 1 deletion.

```
3  src/content/posts/my-git-workflow-for-refactoring/_components.tsx

@@ -36,9 +36,10 @@ export function ExampleButton({
36 36  }
37 37
38 38  /** WE CHANGED THIS ONE BELOW IF YOU'RE LOOKING AT THIS ON GITHUB */
39 - type Variant = 'primary' | 'secondary'
+ type Variant = 'danger' | 'primary' | 'secondary'
40
41 41  const VARIANT_TO_STYLES: Record<Variant, React.CSSProperties> = {
42 +   danger: { backgroundColor: 'red', color: 'white' },
43    primary: PRIMARY_STYLES,
44    secondary: SECONDARY_STYLES,
45  }
```



new version

fichero.txt



```
git add  
git commit
```



GitHub

fichero.txt



old version

old version

fichero.txt





new version

fichero.txt



git push



GitHub

fichero.txt



new version

old version

fichero.txt





new version

fichero.txt



GitHub

fichero.txt



new version

new version

fichero.txt



`git pull`



Ejercicio práctico 1.1:

Primer repositorio



Ejercicio práctico 1.1

- | | |
|-------------------------------------|-------------------------|
| 1. Crear repositorio en GitHub | <code>git init</code> |
| 2. Clonar el repositorio en local | <code>git clone</code> |
| 3. Añadir un nuevo archivo en local | <code>git add</code> |
| 4. Hacer commit de los cambios | <code>git commit</code> |
| 5. Publicar los cambios | <code>git push</code> |



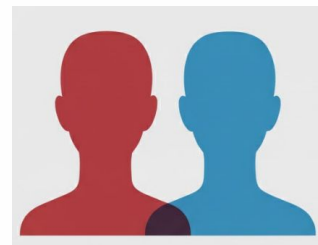
Ejercicio práctico 1.2:

Repositorio Compartido



Ejercicio práctico 1.2

1. Invitar al compañero a participar en el repositorio
2. Clonar el repositorio en local
3. Añadir un nuevo archivo en local
4. Hacer commit de los cambios
5. Publicar los cambios
6. Hacer pull de la versión remota del repositorio

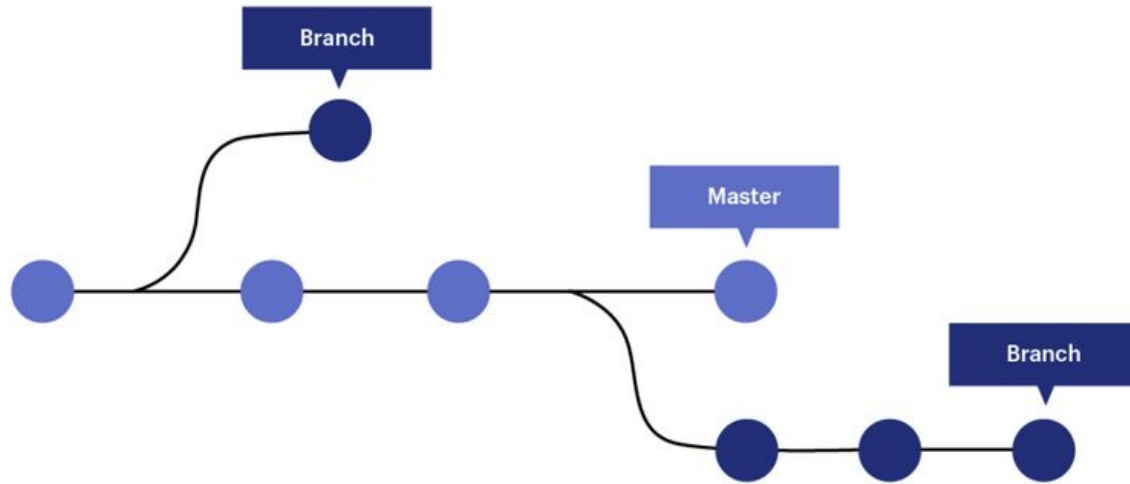


Ramas

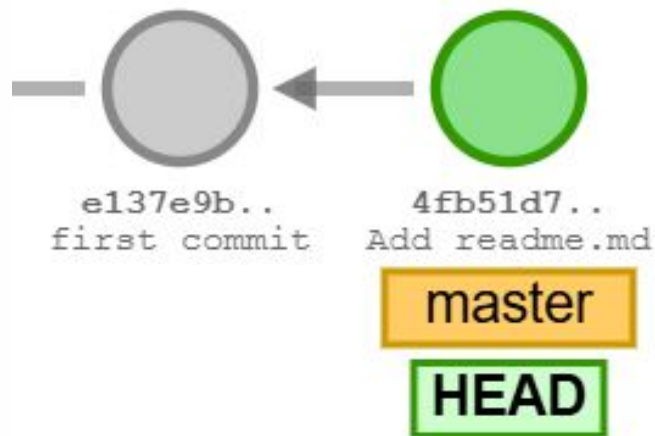
- Trabajo en paralelo
- Rama main/master siempre funciona



Ramas



Ramas



```
git commit -m "Add readme.md"
```

<https://git-school.github.io/visualizing-git/>



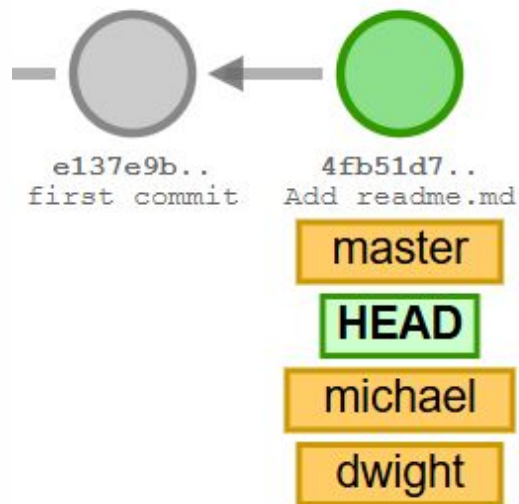


Ramas

```
git branch "name-branch"  
git switch "name-branch"
```



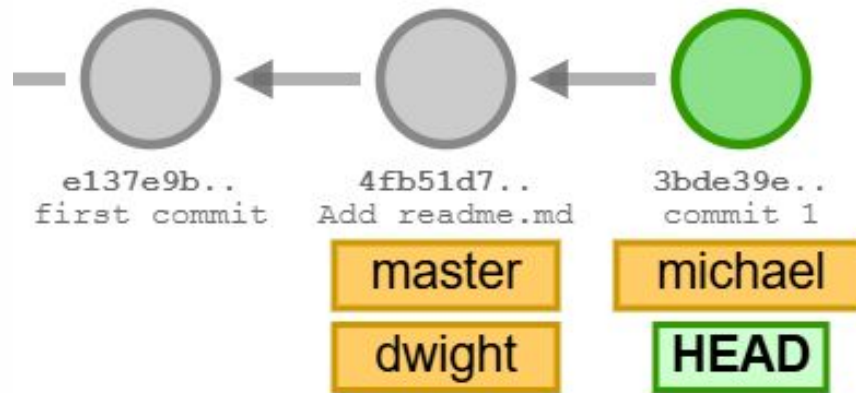
Ramas



```
git branch michael
git branch dwight
```



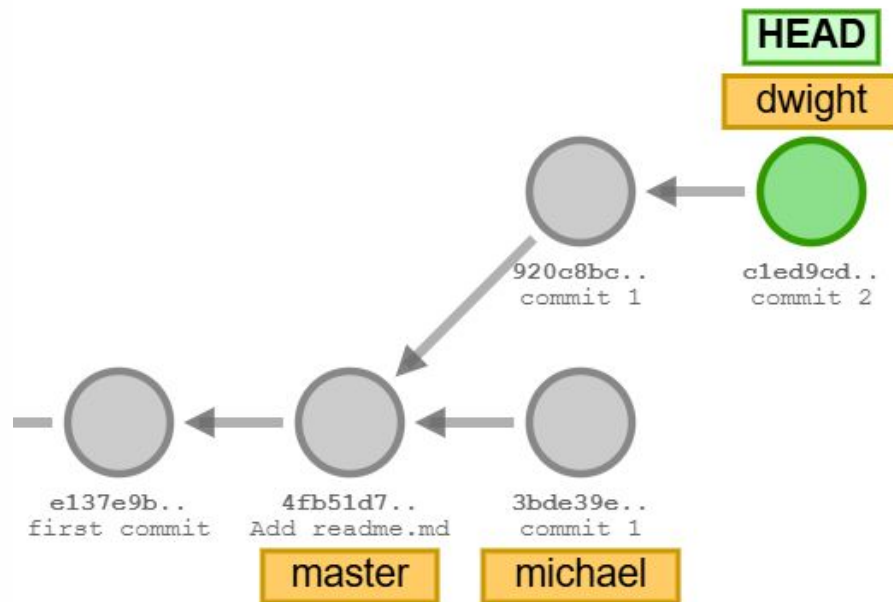
Ramas



```
git switch michael  
git commit -m "commit 1"
```



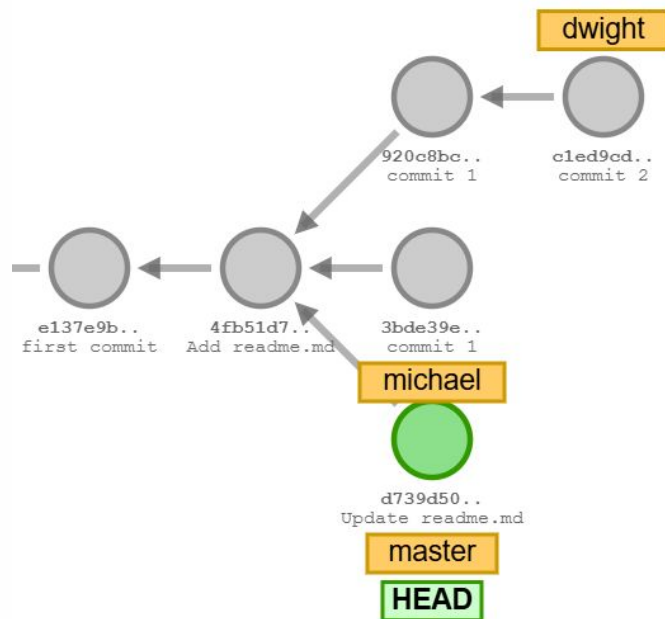
Ramas



```
git switch dwight
git commit -m "commit 1"
git commit -m "commit 2"
```



Ramas



```
git switch master
```

```
git commit -m "Update readme.md"
```





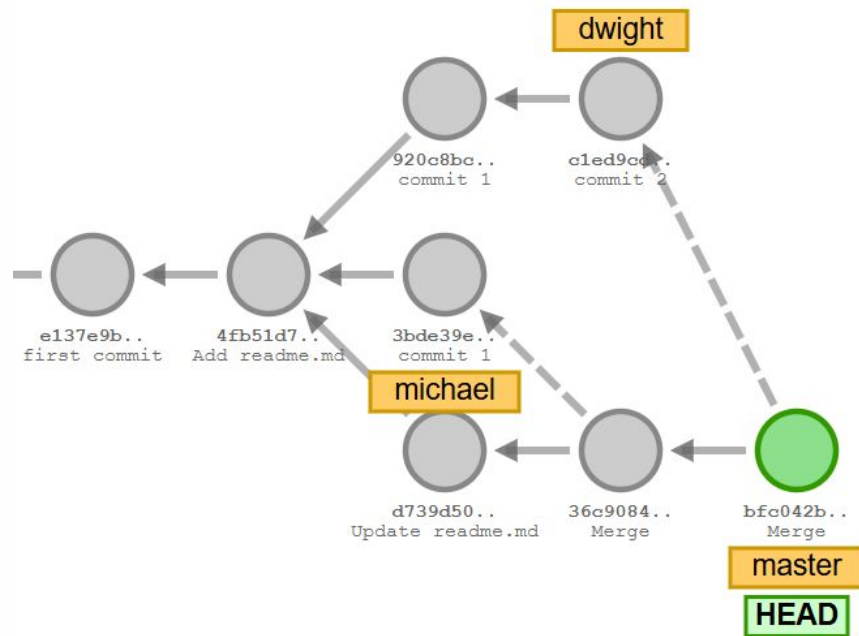
Ramas

Unir los cambios de dos ramas distintas en
una sola

```
git merge
```



Ramas



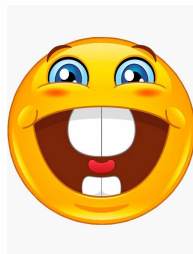
```
git merge michael  
git merge dwight
```



Ramas

Posibilidades al unir dos ramas:

-No conflicto



-Conflicto







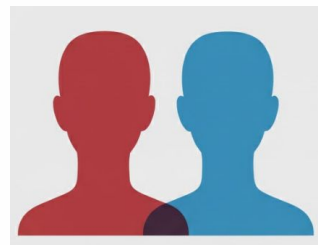
Ejercicio práctico 1.3

Repositorio Compartido con buen
Workflow



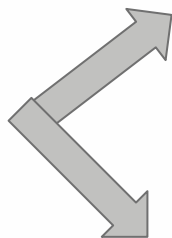
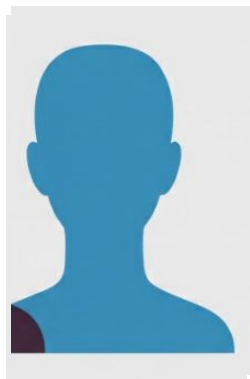
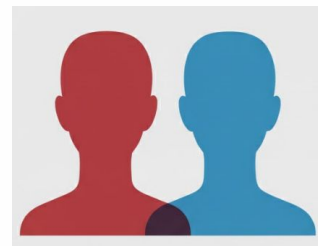
Ejercicio práctico 1.3

1. Crear un nuevo repositorio
2. Añadir 1 commit al repositorio
3. Invitar al compañero a participar en el repositorio
4. Clonar el repositorio en local
5. Crear una nueva rama
6. Añadir dos nuevos archivos en local (en su rama)
7. `archivo_compartido.txt` `archivo_azul.txt`
`archivo_rojo.txt`



Ejercicio práctico 1.3

6. `archivo_compartido.txt` `archivo_azul.txt`
`archivo_rojo.txt`



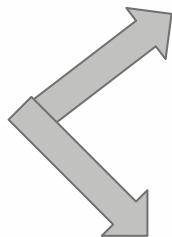
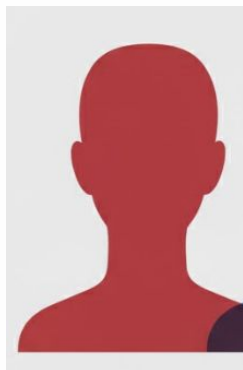
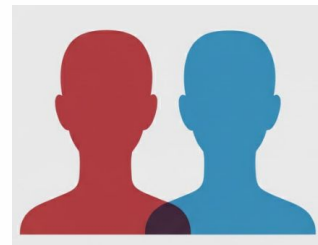
`archivo_compartido.txt`: **distinto contenido que rojo!**

`archivo_azul.txt`: libre



Ejercicio práctico 1.3

6. `archivo_compartido.txt` `archivo_azul.txt`
`archivo_rojo.txt`



`archivo_compartido.txt`: **distinto contenido que azul!**

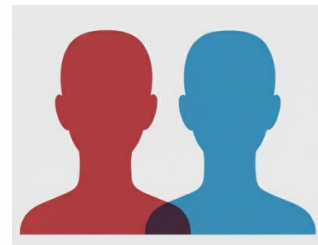
`archivo_rojo.txt`: libre



Ejercicio práctico 1.3

7. Hacer commit de sus cambios (a su rama)

8. Hacer push de su rama al remoto



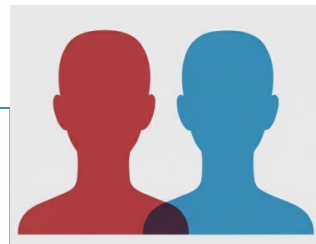
Ejercicio práctico 1.3

Situación:

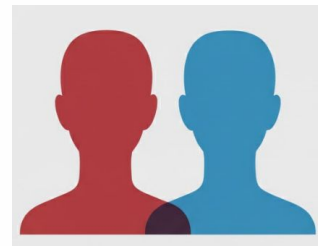
Rama 1: `archivo_compartido.txt` `archivo_rojo.txt`

Rama 2: `archivo_compartido.txt` `archivo_azul.txt`

Rama main/master



Ejercicio práctico 1.3



Default

Branch	Updated	Check status	Behind	Ahead	Pull request
main	8 minutes ago			Default	

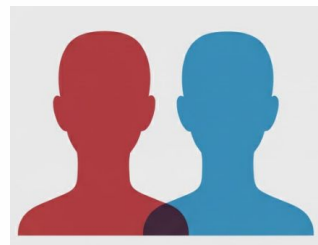
Your branches

Branch	Updated	Check status	Behind	Ahead	Pull request
rojo	1 minute ago		0	1	
azul	3 minutes ago		0	1	



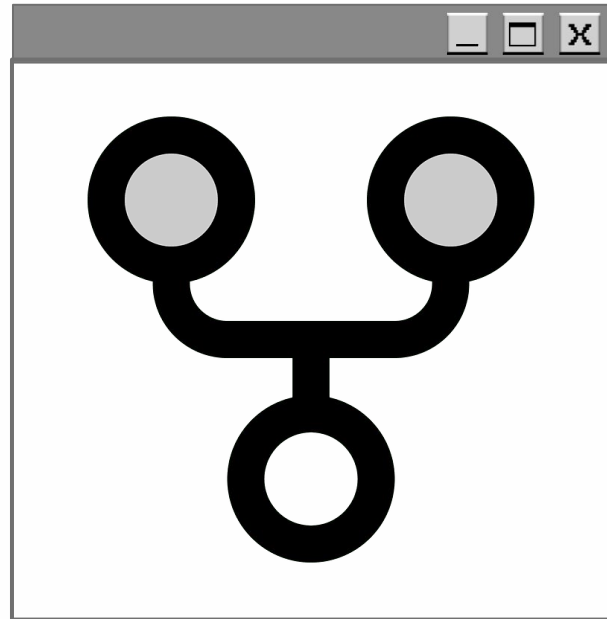
Ejercicio práctico 1.3

9. Incorporar sus cambios a main/master en local
10. Nueva version de main/master al remoto
11. Incorporar sus cambios a main/master en local
12. Nueva versión de main/master al remoto
13. ¡Conflicto!
14. Solucionar conflicto
15. Nueva versión de main/master al remoto



Forks

<https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/working-with-forks/fork-a-repo>



Forks (mediante GitHub)



Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)

Required fields are marked with an asterisk ().*

Owner *

Choose an owner ▾

Repository name *

/

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description

/ 350 characters

☒ Copy the `master` branch only

Contribute back to

by adding your own branch. [Learn more.](#)

Create fork

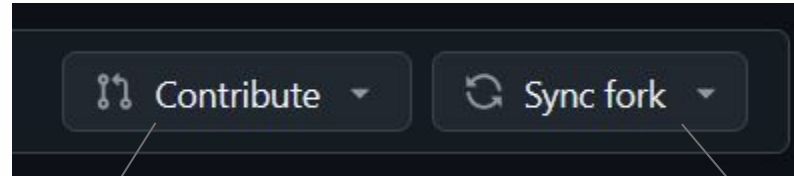


Pull request

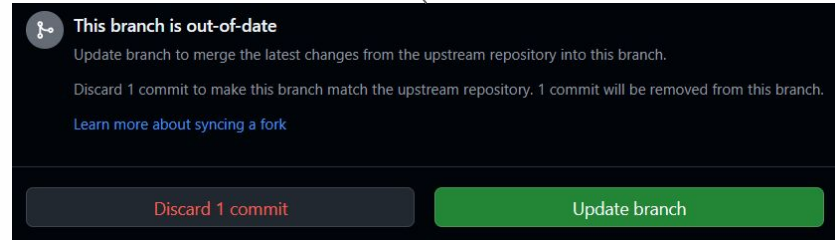
<https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/about-pull-requests>



Pull requests (mediante GitHub)



Open pull request





Ejercicio Práctico 2



1. Fork

Haremos un fork del repositorio desde GitHub.

2. Clone

Clonamos el fork creado a nuestro dispositivo.

3. Commit

Hacemos nuestros cambios (siguiente slide) y le hacemos commit y push.

4. Pull request

Hacemos un pull request y se mergea en la repo.



Ejercicio práctico 2

Cambio:

1. Navega a la carpeta del repositorio clonado
2. Crea un archivo que sea <NIA>.txt, ejemplo:
100550195.txt
3. Dentro del archivo de texto, pon un mensaje, el que quieras



Ejercicio práctico 2

Link:

<https://github.com/guluc3m/actividad-charla-git>

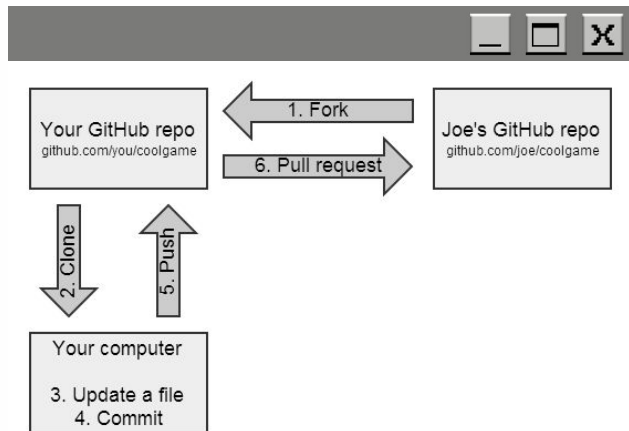
QR del link:



Scan Me



¿Cómo debo contribuir?



- ¿Es útil?
- ¿Aporta sin cambiar drásticamente?
- Sigue las guías del proyecto (CONTRIBUTING.md).
- Incluye ejemplos o pruebas si modificas código.
- Usa mensajes de commit claros.
- Asegúrate de que todo siga funcionando.
- Propón primero un issue si el cambio es grande.



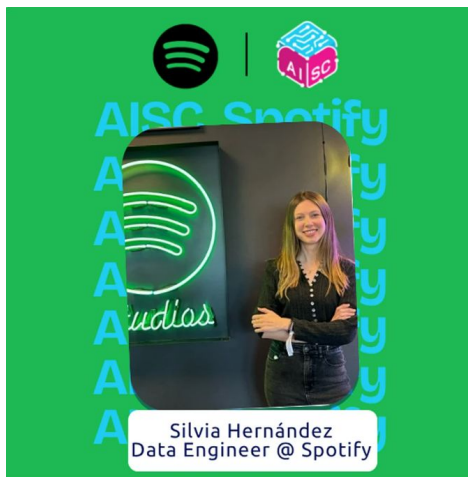
QR Slides



https://aiscmadrid.com/events/event_o.php?id=19



¡Gracias!



Silvia Hernández Data Engineer | AISC x Spotify

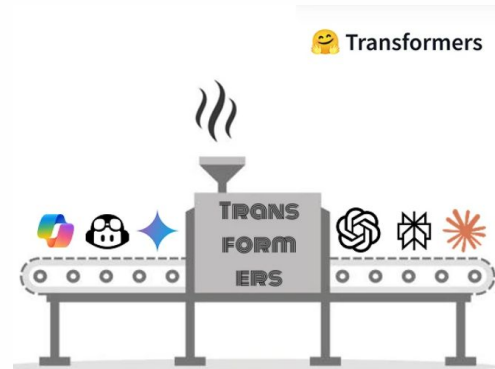
17/11/2025
17:00 - 18:00

Aula 2.3.D03, Edificio Sabatini EPS Universidad Carlos III

Saber más

aiscmadrid.com

gul.uc3m.es



PyTorch



Transformers: El Corazón de la IA Moderna

02/12/2025
13:30 - 14:30

Aula 2.3.A03B, Edificio Sabatini EPS Universidad Carlos III

Saber más