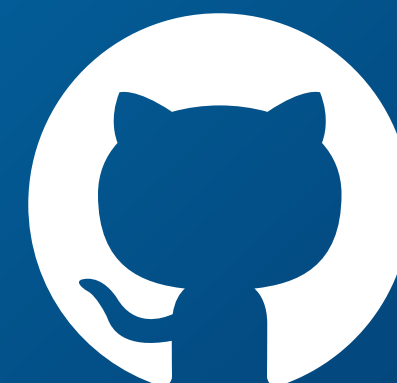




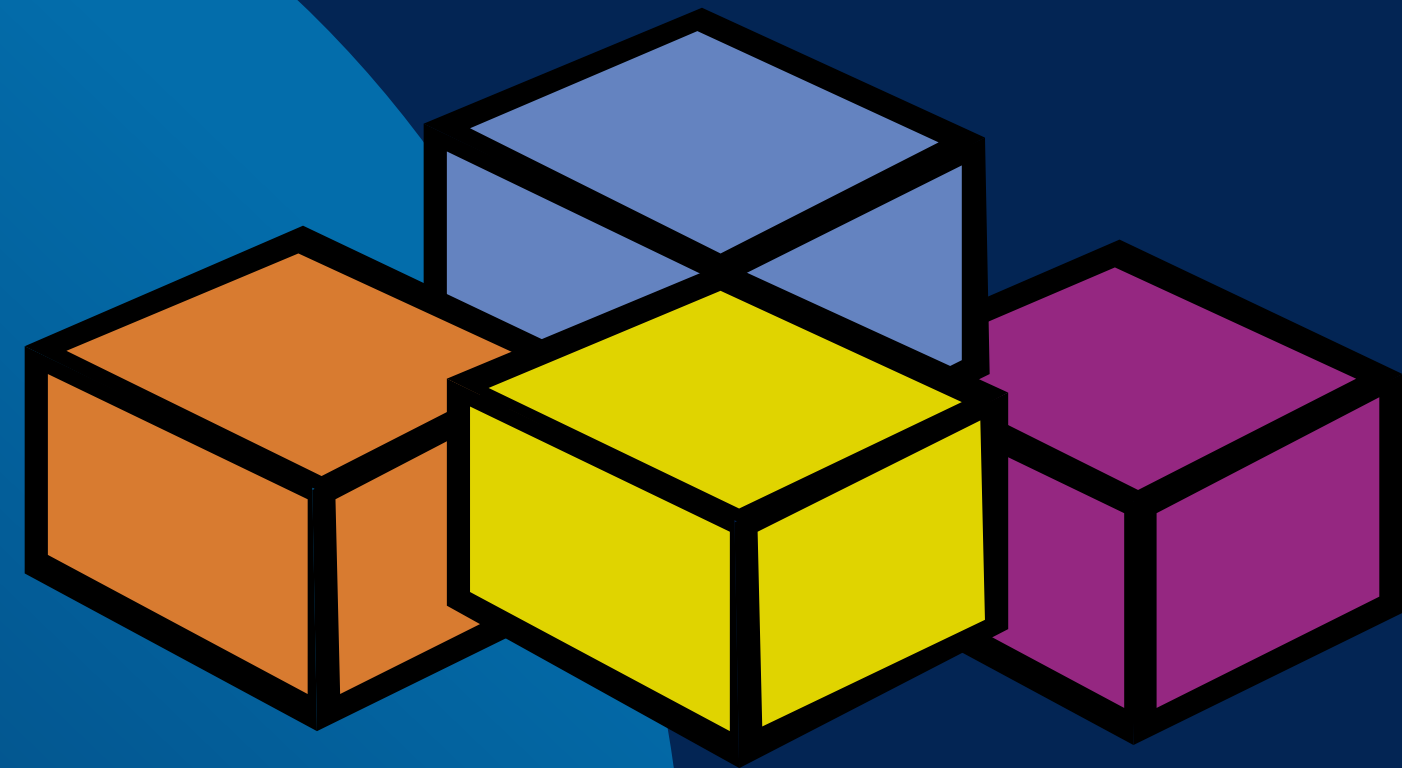
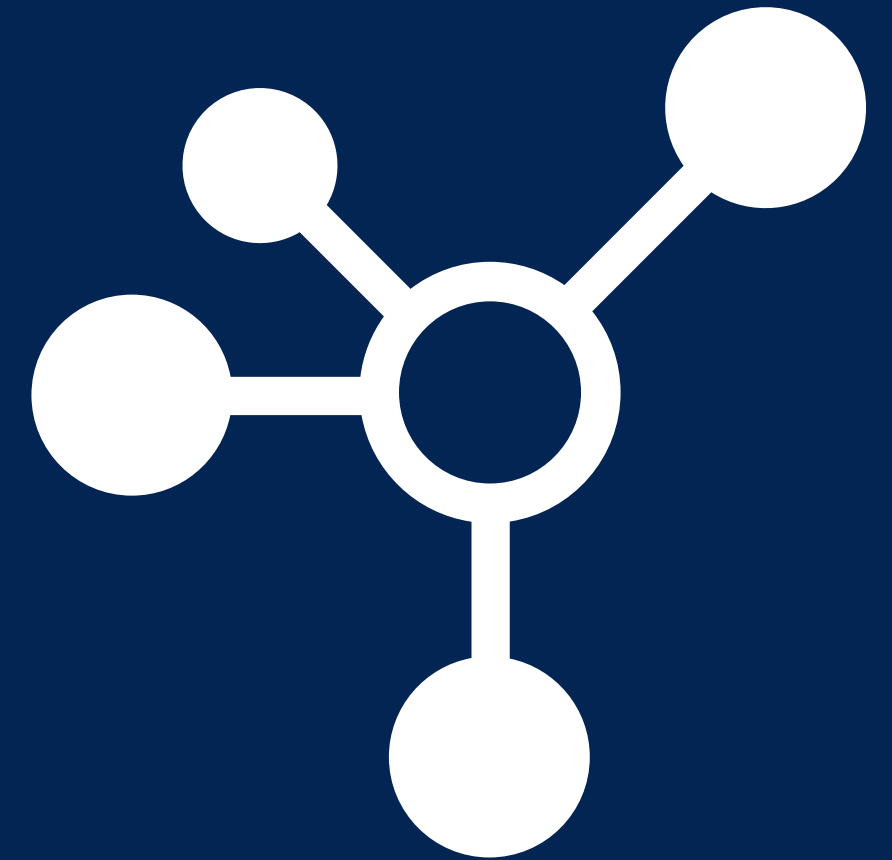
TALLER DE GIT Y GITHUB

Presentado por:
Wilter Díaz
Gustavo Gutiérrez



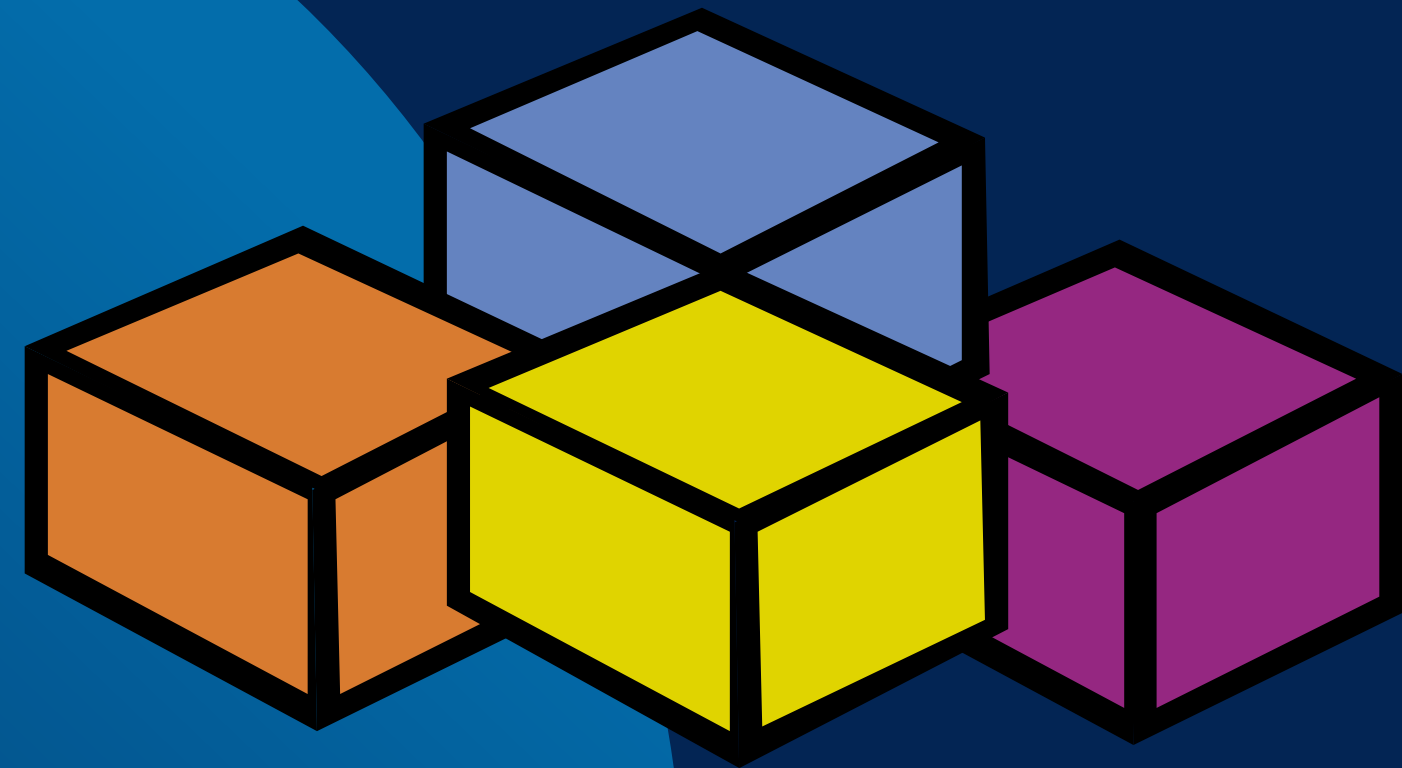
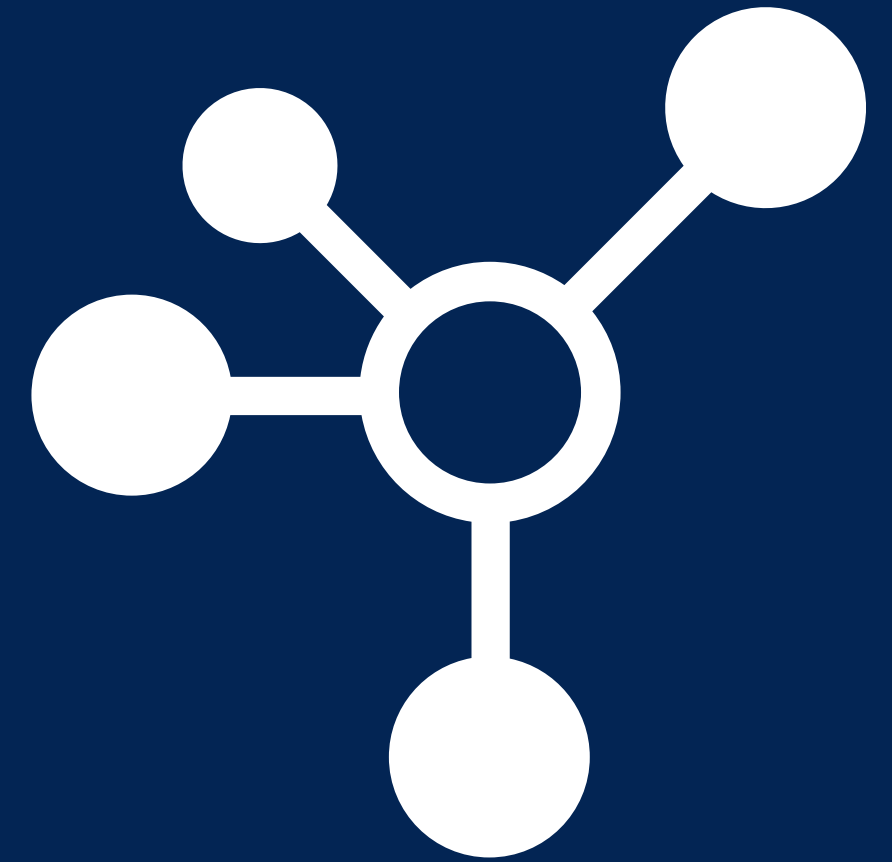
¿QUÉ ES GIT?

Git es un software de control de versiones, pensando en la eficiencia, la confiabilidad y compatibilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente.



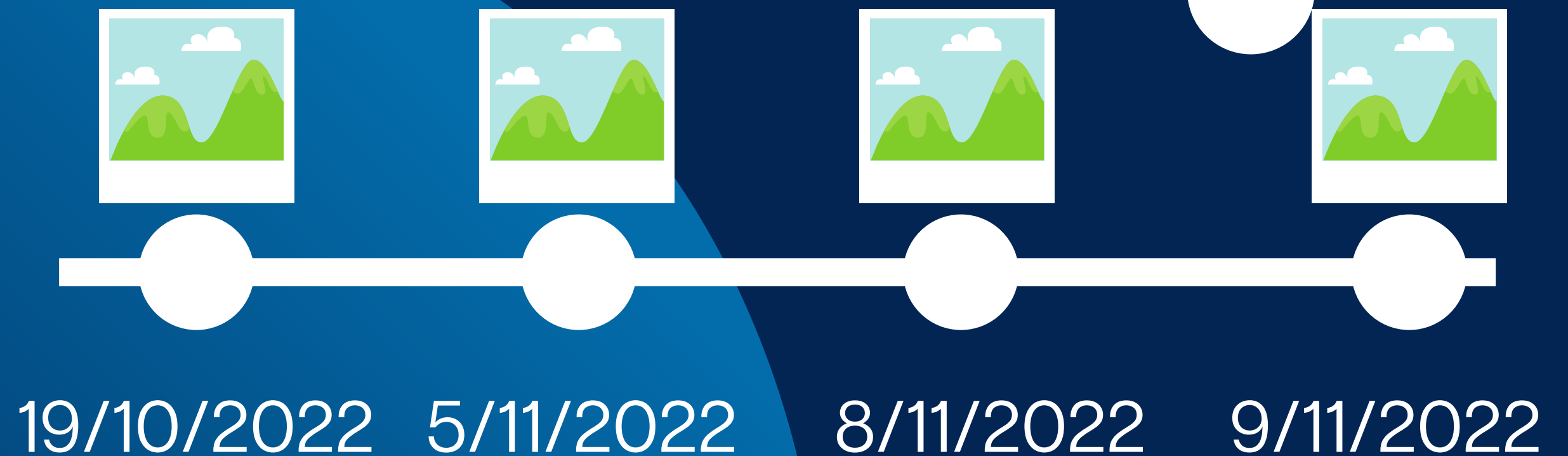
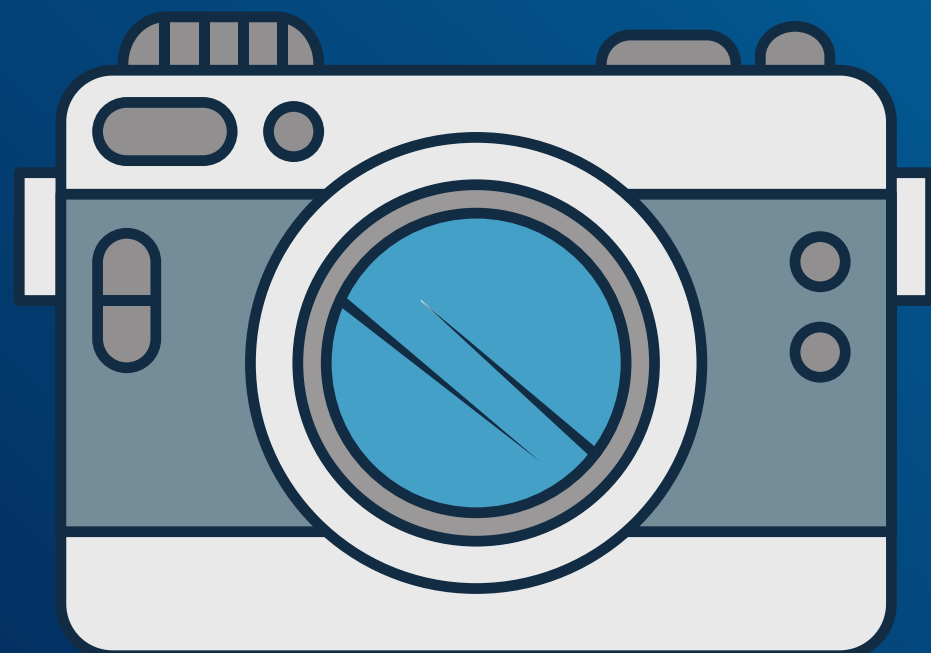
¿QUÉ ES UN REPOSITORIO?

Un repositorio es un espacio centralizado donde se almacena, organiza, mantiene y se crean proyectos informáticos. Permitiendo el trabajo en equipo en proyectos de muchos archivos.



¿QUÉ ES UN COMMIT?

Es una instantánea del estado de los archivos de un repositorio en un momento dado. Es como una fotografía de como estaba el proyecto en algún punto del pasado (puntos de control).



ETAPAS DE UN COMMIT



`git commit`

Working
Directory

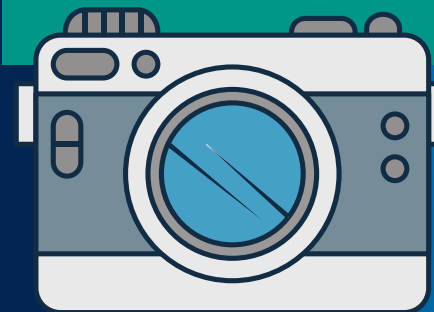
Staging
Area

Local
Repository

Remote
Repository

`git add`

`git push`



GIT

- 1 – Descargar e Instalar GIT
- 2 – Configurar un usuario y correo para el registro de GIT (sólo en la primera instalación)



```
git config --global user.name "tuNombreDeUsuario"
```

```
git config --global user.email "tuCorreoElectronico"
```


PRIMEROS PASOS CON GIT

En Visual Studio Code, abrimos la terminal y colocamos `cd ../rutaProyecto`

Para iniciar un repositorio local utilizamos el siguiente comando.



```
git init
```



PRIMEROS PASOS CON GIT

Para agregar todos los archivos de nuestro proyecto al repositorio, colocamos el siguiente comando.

```
● ● ●  
git add .
```

```
● ● ●  
git add <NombreDelArchivo.txt>
```



Sí sólo quieres agregar un archivo, entonces colocas este comando.

PRIMEROS PASOS CON GIT

Luego de agregar tus archivos al repositorio, colocas el mensaje del cambio realizado.

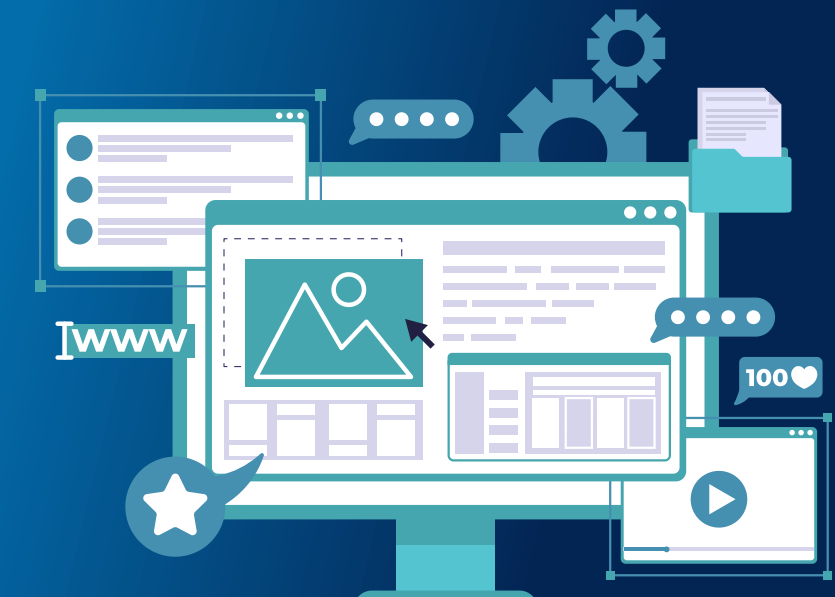


```
git commit -m "mensaje"
```

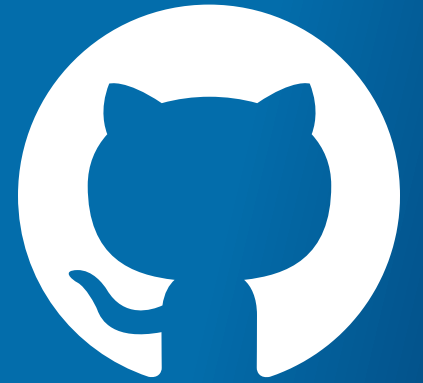
Ejemplos



```
git commit -m "Diseño del menú del proyecto"
```



PRIMEROS PASOS CON GIT

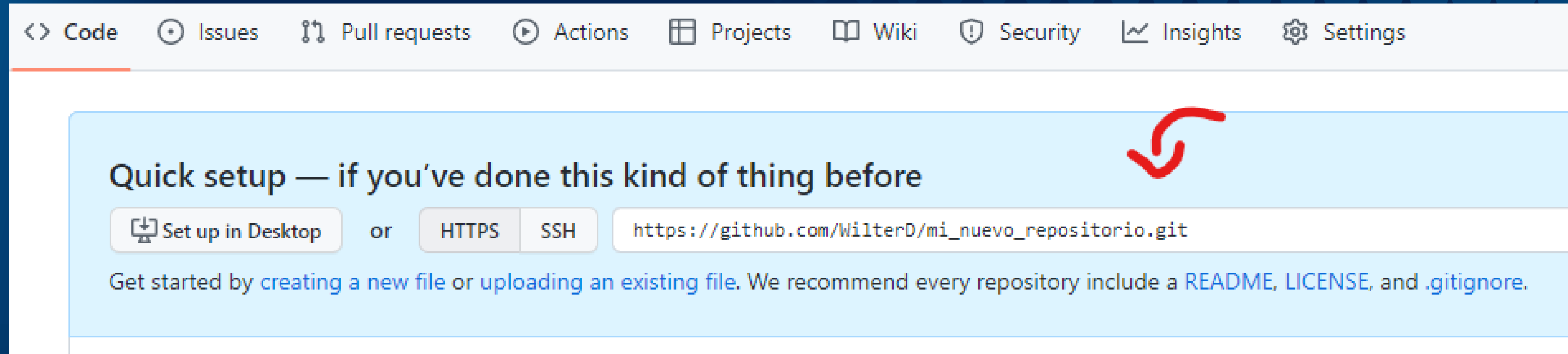


Para ver las modificaciones en nuestro repositorio, colocamos el siguiente comando.



```
git status
```

EN NUESTRA TERMINAL COLOCAMOS LA URL DEL REPOSITORIO



```
git remote add origin <url_del_repositorio>
```



SUBIMOS NUESTRO REPOSITORIO LOCAL A GITHUB

git push , envía nuestro repositorio local hacia nuestro repositorio remoto



```
git push -u origin master
```



CLONAR OTRO REPOSITORIO



```
git clone <url_del_repositorio>
```

Con este comando podemos trabajar con otro proyecto que ya esté en github y descargar los archivos a nuestro ordenador.

HISTORIAL DE VERSIONES - COMMITS

Para visualizar los commits realizados en nuestro proyecto podemos ejecutar.



```
git log
```

```
git log --oneline
```



RAMAS EN GITHUB - BRANCH

Con git branch, creamos una nueva rama en nuestro repositorio local.

Con git checkout, nos cambiamos a esa rama del proyecto.



```
git branch <NombreDeLaRama>  
git checkout <NombreDeLaRama>
```



RAMAS EN GITHUB - BRANCH



Con este comando, creamos la rama y nos cambiamos a la rama automáticamente.



```
git checkout -b <NombreDeLaRama>
```

RAMAS EN GITHUB - BRANCH



Con este comando, vemos la lista de las ramas de nuestro proyecto.



```
git branch
```

RAMAS EN GITHUB - BRANCH



Con este comando, eliminamos una rama de nuestro proyecto de forma **LOCAL**



```
git branch -D <NombreDeLaRama>
```

RAMAS EN GITHUB - BRANCH



Con este comando, eliminamos una rama de nuestro proyecto de forma **REMOTA**



```
git push origin --delete <NombreDeLaRama>
```

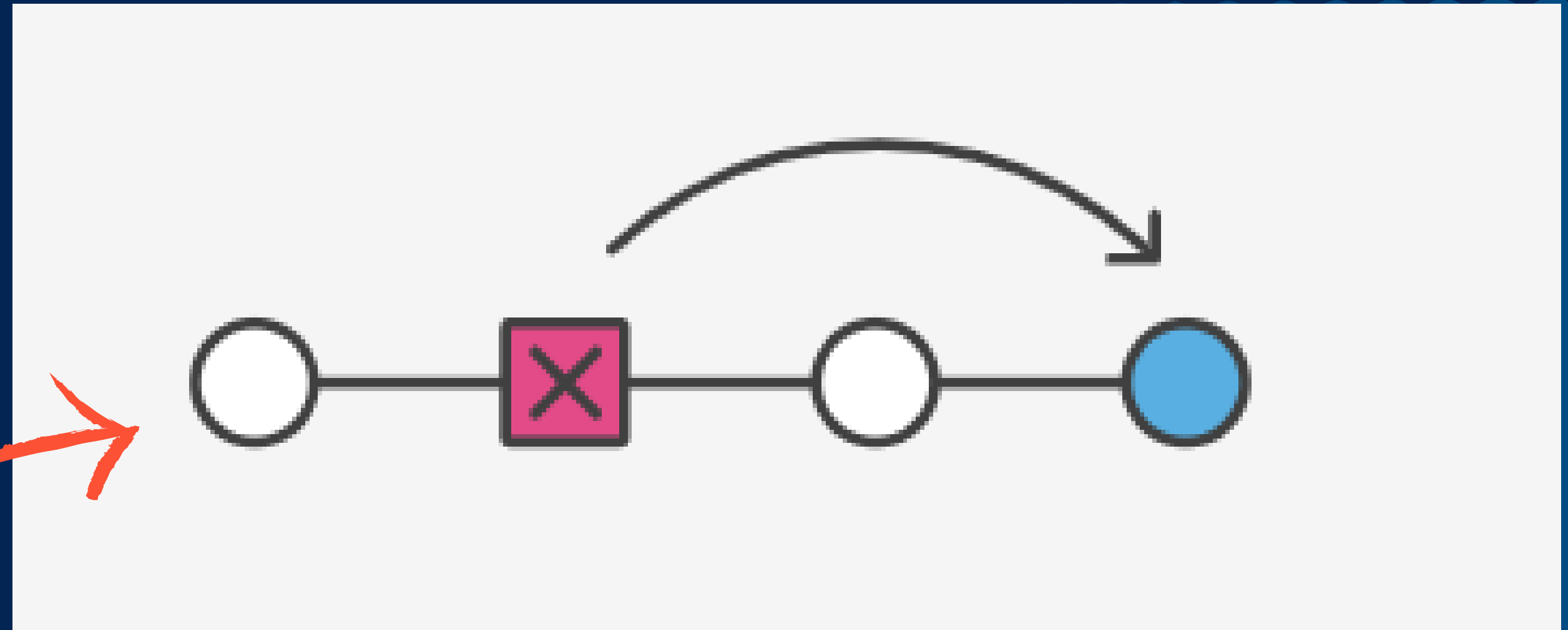
REGRESAR EN EL TIEMPO – COMMITS

Con este comando podemos ir a un commit anterior realizado en nuestro proyecto



```
git checkout <commit_id>
```

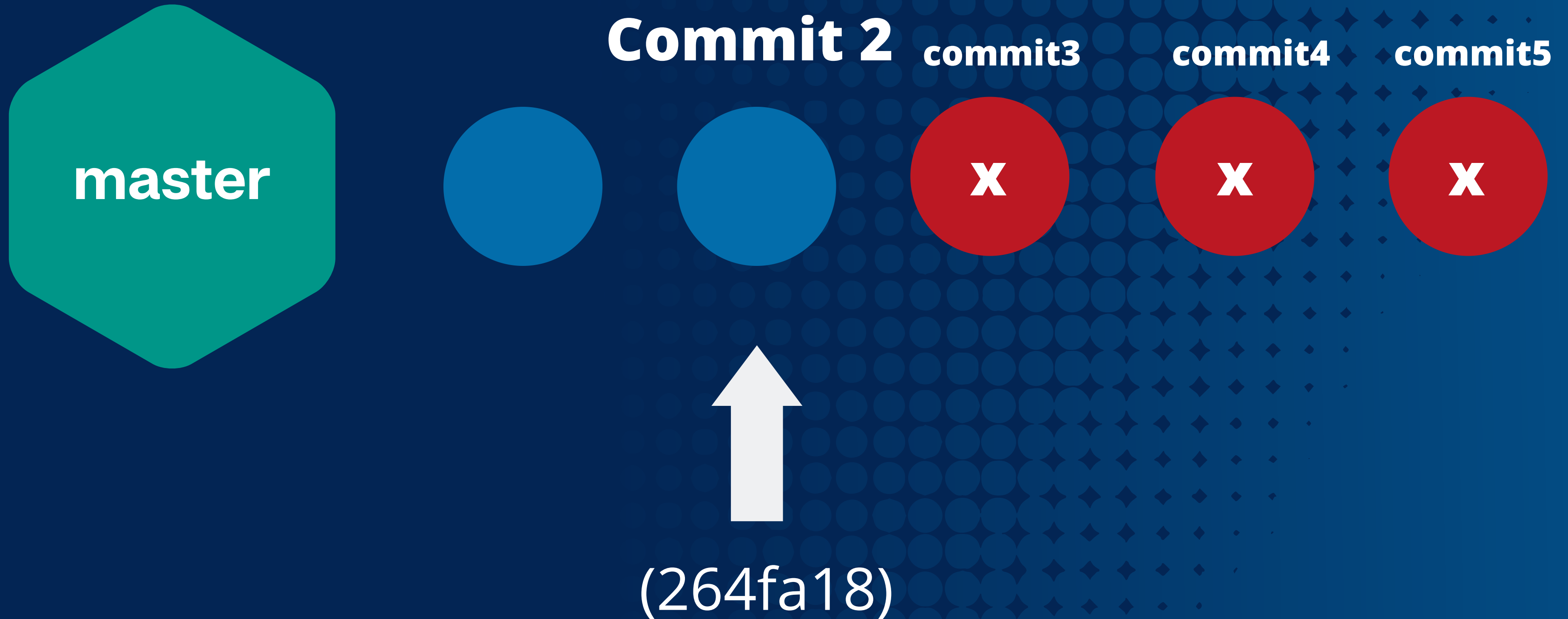

GIT REVERT



```
git revert <commit_hash>
```

Con git revert, añadimos un nuevo commit adelante, revirtiendo los cambios sin mover la cabeza de la lista.

GIT RESET



FUSIONAR RAMAS – MERGE



```
git checkout <rama_principal>  
git merge <rama_a_unir>
```

Con este comando fusionamos dos ramas.

Nota: Debemos evitar conflictos entre fusión de ramas.

(dos ramas modificadas en mismo código)

ACTUALIZAR RAMA LOCAL - PULL



```
git pull origin <rama_principal>
```

Cuando vamos a subir nuestros cambios, debemos actualizar nuestro repositorio local, con los últimos cambios del repositorio remoto.



CEINF



@ceinf_ucabg



@cde_ucabg

