

GIT DESDE CERO

LECCIÓN 2: MANEJO DEL HISTORIAL

DURACIÓN ESTIMADA 50 MIN

Agenda

- Corrección de pratica I
- Repaso
- Introducción manejo de historial
- Comandos
- Archivo .gitignore
- Práctica



Práctica I (corrección)



- Instalar git
- Crear una cuenta en GitLab o GitHub
- Clonar el repositorio:
 - GitLab:** <https://gitlab.com/JSibaja/tallergitdede0.git>
 - GitHub:** <https://github.com/JamesSibaja/TallerGitDede0.git>
- **Llenar formulario y esperar:**
<https://forms.gle/xrDcCodbwAxhdVsS6>
- **Subir un archivo de texto con su nombre al repositorio**



REPASO

LECCIÓN 1: INTRODUCCIÓN

git



Repositorio

- Es un almacenamiento virtual que permite guardar y acceder a versiones de tu proyecto

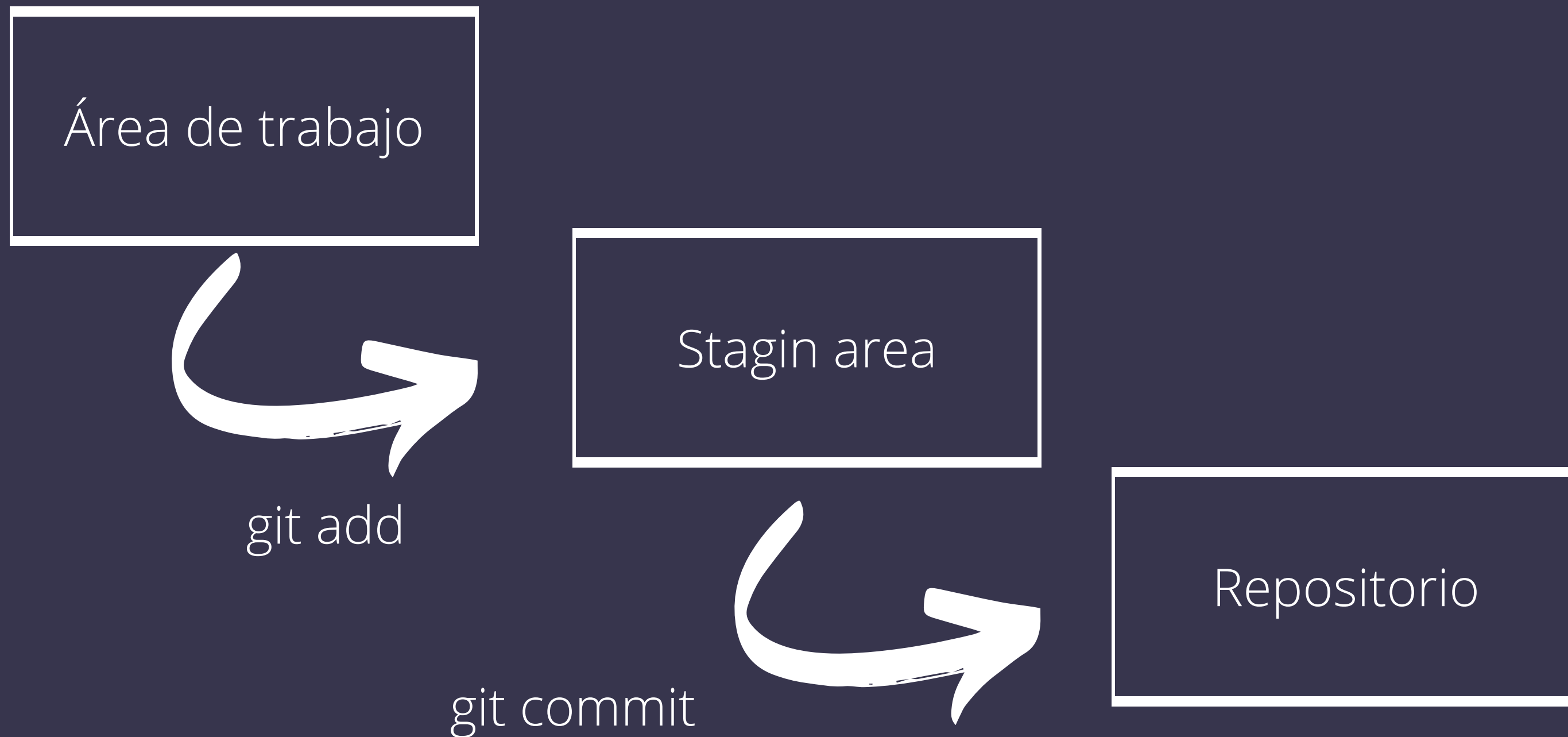
Commit

- Es como una fotografía de nuestro proyecto en un determinado momento

Flujo de trabajo



Creamos o clonamos el repositorio



Pasamos de esto ...



Trabajo historia



Trabajo historia
versión 2



Trabajo historia
versión 2 revisado



Trabajo historia
versión 2 revisado
final



Trabajo historia
versión 2 revisado
final definitivo

a esto ...



Trabajo historia

¿Dónde están mis versiones?

¿Cómo manejar mi historial?



Git

Commit 1:

Trabajo historia

Commit 2:

Trabajo historia versión 2

...

COMANDOS BÁSICOS

LECCIÓN 2: MANEJO DEL HISTORIAL



Lista de comandos

- git config
- git rm
- git log
- git diff
- git reset
- git checkout

git rm



- **git rm <archivo>**
- **git rm --cached <archivo>**

La diferencia es que uno mantiene intacta el área de trabajo (--cached) y el otro no

git config



- **git config --global user.name "<Nombre de usuario>"**
- **git config --global user.email "<Correo>"**

git log



```
Peppermint Terminal
james@james-pc ~/Escritorio/ESCRITORIO/nuevogit $ git log
commit 1716971c265d6f2f9b3b9efb6bd52130056fab8d (HEAD -> master)
Author: jfsibajag <jsibajagranados2@gmail.com>
Date:   Wed Jun 16 13:36:51 2021 -0600

    gitignore

commit 503a7eff0a15fb96e991e6f99d005d9d8d87eb81
Author: jfsibajag <jsibajagranados2@gmail.com>
Date:   Wed Jun 16 13:17:11 2021 -0600

    pepe pepe

commit 474da3036a988dd69b04aa16365915362be6c21b
Author: jfsibajag <jsibajagranados2@gmail.com>
Date:   Wed Jun 16 13:13:09 2021 -0600

    terer commit
```

Identificador
único del commit

Datos del commit

git log --oneline



Muestra el historial de git pero en una sola linea

```
Peppermint Terminal
james@james-pc ~/Escritorio/ESCRITORIO/nuevogit $ git log --oneline
1716971 (HEAD -> master) gitignore
503a7ef pepe pepe
474da30 terer commit
9f8f059 otro mensaje
e2fa74e mensaje
```

git log --stat

Muestra más detalles



```
Peppermint Terminal
james@james-pc ~/Escritorio/ESCRITORIO/nuevogit $ git log --stat
commit 1716971c265d6f2f9b3b9efb6bd52130056fab8d (HEAD -> master)
Author: jfsibajag <jsibajagranados2@gmail.com>
Date:   Wed Jun 16 13:36:51 2021 -0600

    gitignore

    .gitignore | 1 +
    1 file changed, 1 insertion(+)

commit 503a7eff0a15fb96e991e6f99d005d9d8d87eb81
Author: jfsibajag <jsibajagranados2@gmail.com>
Date:   Wed Jun 16 13:17:11 2021 -0600

    pepe pepe

    text.txt | 3 ++-
    1 file changed, 2 insertions(+), 1 deletion(-)

commit 474da3036a988dd69b04aa16365915362be6c21b
Author: jfsibajag <jsibajagranados2@gmail.com>
```

Para salir se
presiona la tecla q

git log <archivo>



Muestra solo los commits donde se modificó el archivo

```
Peppermint Terminal
james@james-pc ~/Escritorio/ESCRITORIO/nuevogit $ git log text.txt
commit 503a7eff0a15fb96e991e6f99d005d9d8d87eb81
Author: jfsibajag <jsibajagranados2@gmail.com>
Date:   Wed Jun 16 13:17:11 2021 -0600

    pepe pepe

commit 474da3036a988dd69b04aa16365915362be6c21b
Author: jfsibajag <jsibajagranados2@gmail.com>
Date:   Wed Jun 16 13:13:09 2021 -0600

    terer commit

commit 9f8f05946a8d04ca6b7fbd4851acf5a0ec15eb23
Author: jfsibajag <jsibajagranados2@gmail.com>
Date:   Wed Jun 16 13:12:01 2021 -0600

    otro mensaje
```

git diff

Muestra las diferencias específicas entre el área de trabajo y el último commit



```
Peppermint Terminal
james@james-pc ~/Escritorio/ESCRITORIO/nuevogit $ git diff
diff --git a/text.txt b/text.txt
index 60a073e..6b81736 100644
--- a/text.txt
+++ b/text.txt
@@ -1,3 +1,5 @@
  Adios james
  Je je je
- pepe pepe
\ No newline at end of file
+ pepe pepe
+ más cosas diferentes
+ mucho mucho
\ No newline at end of file
```

git diff <commit1> <commit2>



Muestra las diferencias específicas entre dos commit.

También se puede:

- `git diff <archivo>`
- `git diff <commit1> <commit2> <archivo>`

para ver las diferencias en un solo archivo

```
Peppermint Terminal
james@james-pc ~/Escritorio/ESCRITORIO/nuevogit $ git diff 503a7ef 474da30
diff --git a/text.txt b/text.txt
index 60a073e..6e77741 100644
--- a/text.txt
+++ b/text.txt
@@ -1,3 +1,2 @@
  Adios james
- Je je je
- pepe pepe
\ No newline at end of file
+ Je je je
\ No newline at end of file
james@james-pc ~/Escritorio/ESCRITORIO/nuevogit $
```

git reset



Git

Commit 1:

Trabajo historia

Commit 2: (Actual)

Trabajo historia versión 2

git reset



Git

Commit 1: (Actual)

Trabajo historia

git reset



- **git reset --soft <commit>**
- **git reset --hard <commit>**

La diferencia es que uno mantiene intacta el área de preparación (--soft) y el otro la borra (--hard)

git checkout



Revierte todos los cambios del área de trabajo al último commit

git checkout <commit>



Git

Commit 1:

Trabajo historia

Commit 2: (Actual)

Trabajo historia versión 2

git checkout <commit>



Git

Commit 1: (Actual)

Trabajo historia

Commit 2:

Trabajo historia versión 2

ARCHIVO .GITIGNORE

LECCIÓN 2: MANEJO DEL HISTORIAL

Archivo .gitignore



- Paso 1: Crear un archivo con el nombre .gitignore
- Paso 2: Agregar una lista de archivos que se quieran excluir del repositorio

.gitignore

Archivo 1

Archivo 2

Archivo 3

...

PRACTICA II: EL HISTORIAL

LECCIÓN 2: MANEJO DEL HISTORIAL

Práctica



- Escribir git pull en el repositorio del taller
- Terminar práctica I corregida (el último punto puede quedar pendiente)
- Abrir pdf con la practica II y seguir sus pasos