

GIT DESDE CERO

LECCIÓN 1: INTRODUCCIÓN

DURACIÓN ESTIMADA 50 MIN

Agenda

- ¿Qué es git
- Conceptos básicos
- Comandos básicos
- Práctica



¿QUÉ ES GIT?

LECCIÓN 1: INTRODUCCIÓN

git



Trabajo historia





Trabajo historia



Trabajo historia
versión 2





Trabajo historia



Trabajo historia
versión 2



Trabajo historia
versión 2 revisado





Trabajo historia



Trabajo historia
versión 2



Trabajo historia
versión 2 revisado



Trabajo historia
versión 2 revisado
final



Trabajo historia



Trabajo historia
versión 2



Trabajo historia
versión 2 revisado



Trabajo historia
versión 2 revisado
final



Trabajo historia
versión 2 revisado
final definitivo

¿Existe una mejor forma?



Trabajo historia



Trabajo historia
versión 2



Trabajo historia
versión 2 revisado



Trabajo historia
versión 2 revisado
final



Trabajo historia
versión 2 revisado
final definitivo

Git



- Es un sistema de control de versiones distribuido, rápido y escalable
- Creado por Linus Torvalds
- Ideal para el trabajo colaborativo
- Extendido e integrado a diversas herramientas

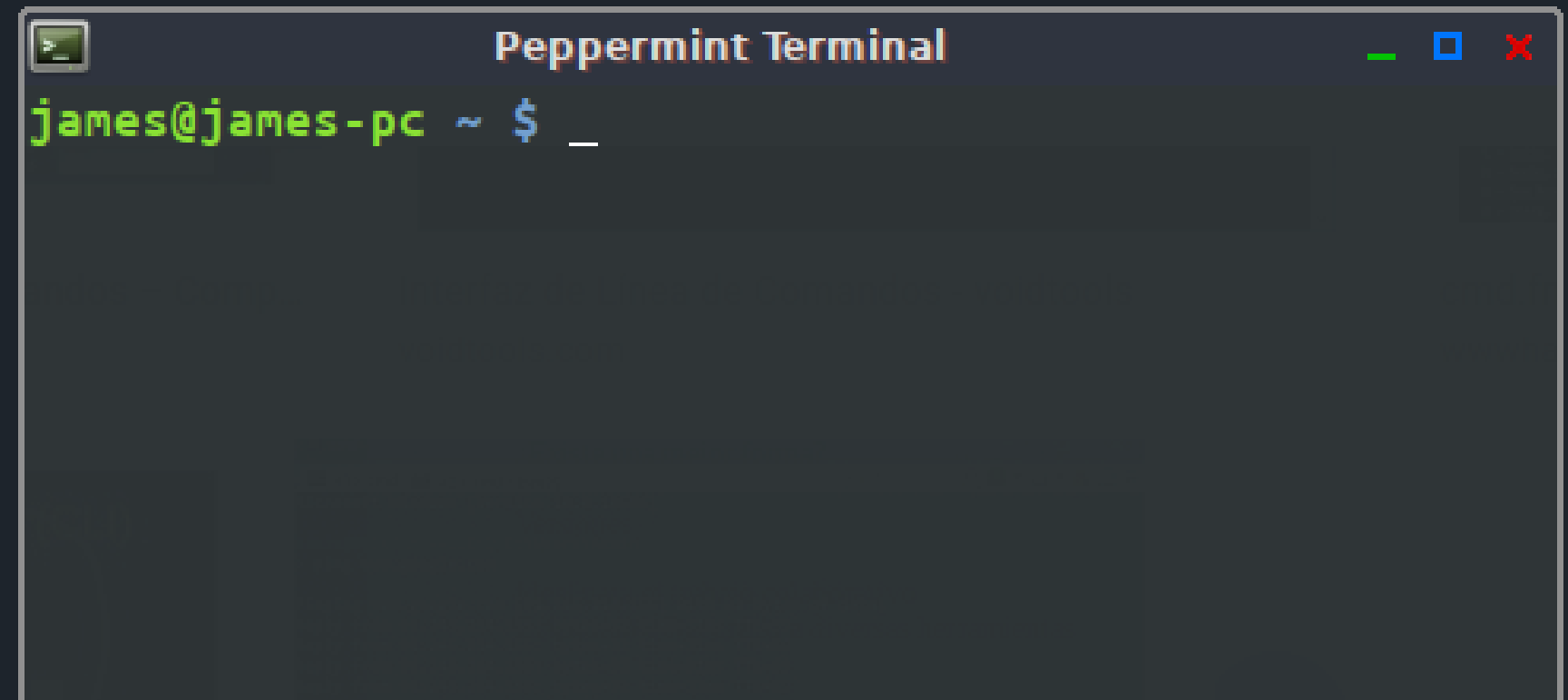
CONCEPTOS BÁSICOS

LECCIÓN 1: INTRODUCCIÓN



Linea de comandos

- Es una interfaz que permite a los usuarios dar instrucciones al sistema operativo por medio de texto simple



Repositorio

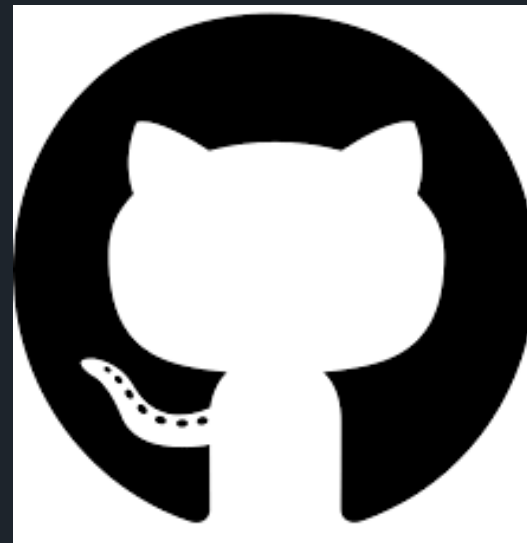


- Es un almacenamiento virtual que permite guardar y acceder a versiones de tu proyecto



Repositorio remoto

- Repositorio que se encuentra almacenado en alguna plataforma web



GitHub

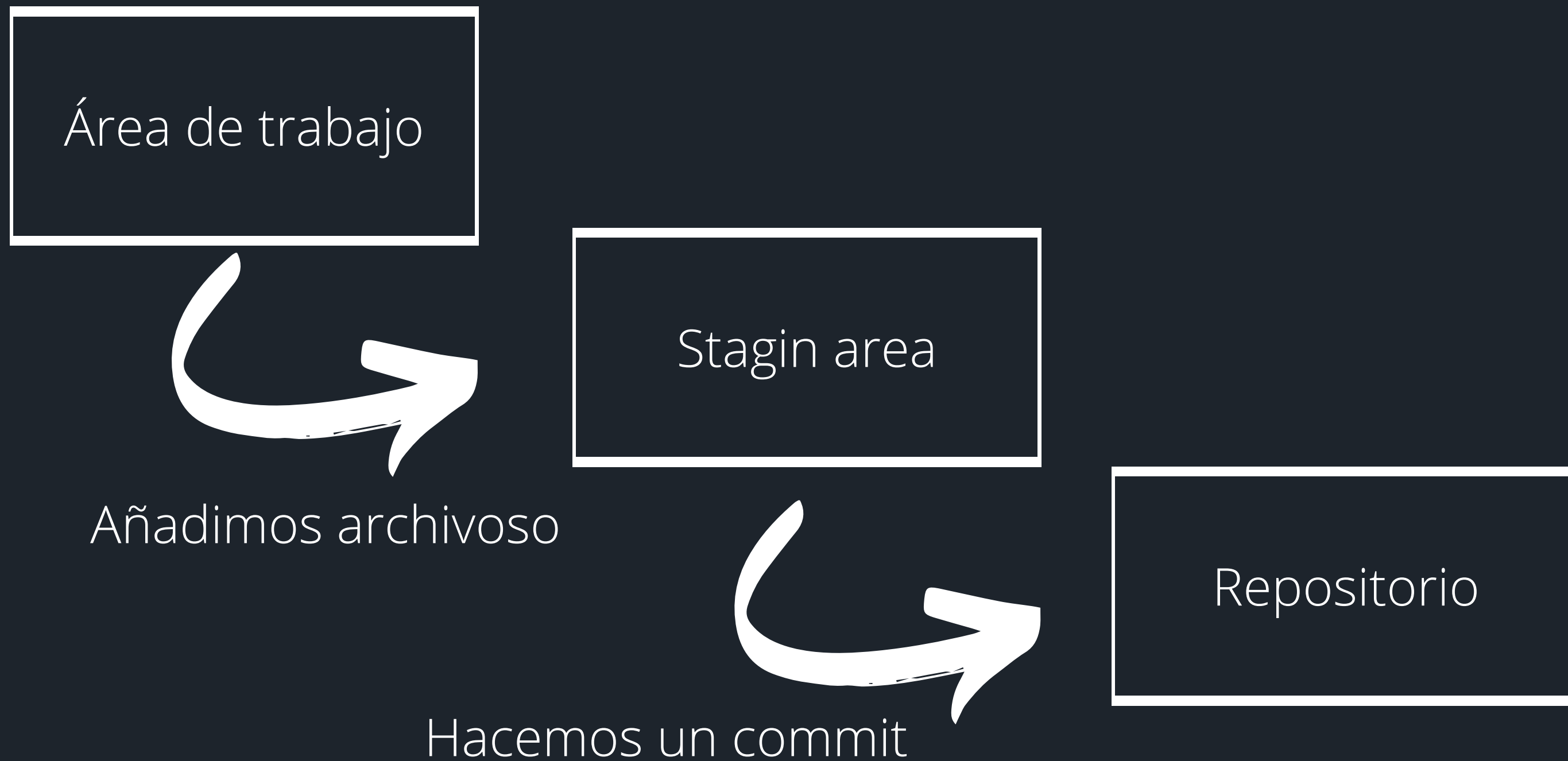


GitLab

Flujo de trabajo



Creamos o clonamos el repositorio



COMANDOS BÁSICOS

LECCIÓN 1: INTRODUCCIÓN



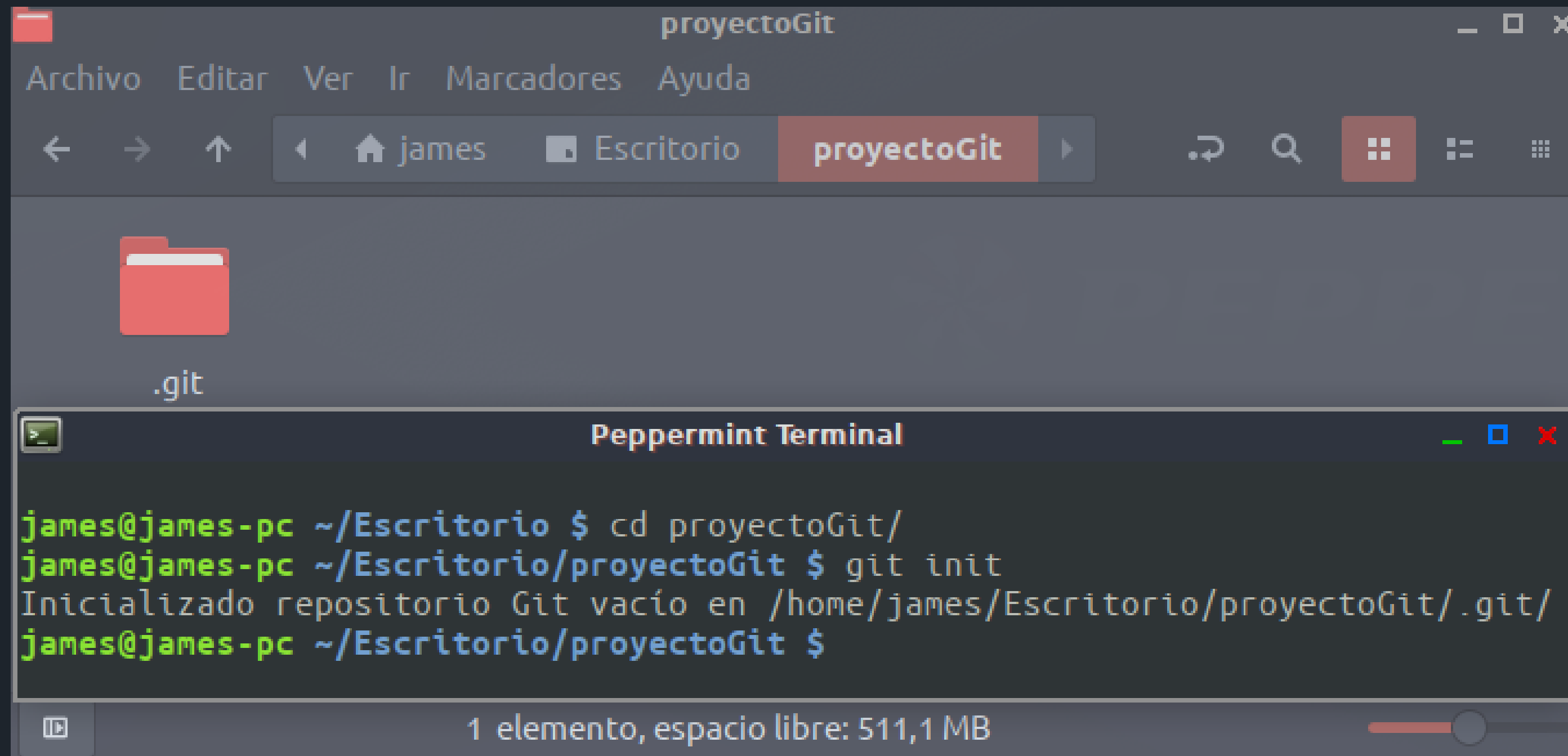
Lista de comandos

- git init
- git add
- git status
- git commit
- git push
- git push
- git pull

git init



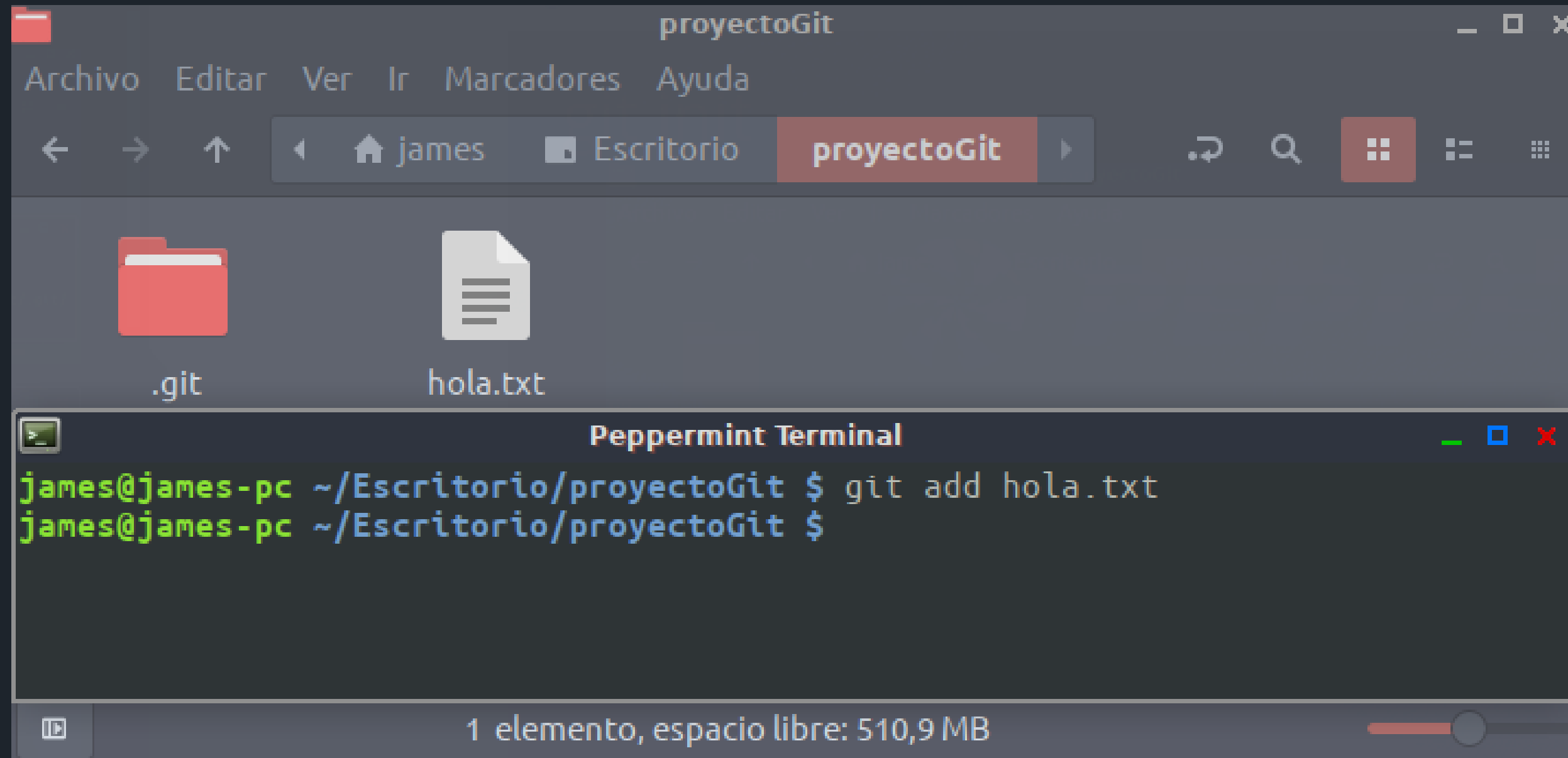
Inicia un repositorio de git



Añadir un poco de texto

git add

Añade un archivo al área de preparación



git status



Muestra el estado de los archivos según el flujo de trabajo

```
Peppermint Terminal
james@james-pc ~/Escritorio/proyectoGit $ touch adios.txt
james@james-pc ~/Escritorio/proyectoGit $ git status
En la rama master

No hay commits todavía

Cambios a ser confirmados:
  (usa "git rm --cached <archivo>..." para sacar del área de stage)

    nuevo archivo:  hola.txt

Archivos sin seguimiento:
  (usa "git add <archivo>..." para incluirlo a lo que se será confirmado)

    adios.txt

james@james-pc ~/Escritorio/proyectoGit $
```

git commit



Guarda el estado de la versión actual en el repositorio

```
Peppermint Terminal
james@james-pc ~/Escritorio/proyectoGit $ git commit
[master (commit-raíz) fa3a085] primer commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 hola.txt
james@james-pc ~/Escritorio/proyectoGit $ git status
En la rama master
Archivos sin seguimiento:
  (usa "git add <archivo>..." para incluirlo a lo que se será confirmado)

    adios.txt

no hay nada agregado al commit pero hay archivos sin seguimiento presentes (usa
a "git add" para hacerles seguimiento)
james@james-pc ~/Escritorio/proyectoGit $
```

git commit



```
Peppermint Terminal
/home/james/Escritorio/proyectoGit/.git/COMMIT_EDITMSG Modificado

primer commit
# Por favor ingresa el mensaje del commit para tus cambios. Las
# líneas que comiencen con '#' serán ignoradas, y un mensaje
# vacío aborta el commit.
#
# En la rama master
#
# Confirmación inicial
#
# Cambios a ser confirmados:
#     nuevo archivo:  hola.txt
#

[ 15 líneas leídas ]
^G Ver ayuda    ^O Guardar     ^W Buscar      ^K Cortar Texto ^J Justificar
^X Salir        ^R Leer fich.  ^\ Reemplazar  ^U Pegar txt    ^T Ortografía
```

git commit



```
Peppermint Terminal
james@james-pc ~/Escritorio/proyectoGit $ git add adios.txt
james@james-pc ~/Escritorio/proyectoGit $ git commit -m "segundo commit"
[master 53ec77b] segundo commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 adios.txt
james@james-pc ~/Escritorio/proyectoGit $ git status
En la rama master
nada para hacer commit, el árbol de trabajo está limpio
james@james-pc ~/Escritorio/proyectoGit $
```

git clone



Crea copia local exacta del repositorio web

```
Peppermint Terminal
james@james-pc ~/Escritorio $ ls
ESCRITORIO  proyectoGit
james@james-pc ~/Escritorio $ git clone https://gitlab.com/JSibaja/tallergitdede0.git
Clonando en 'tallergitdede0'...
warning: Parece haber clonado un repositorio sin contenido.
james@james-pc ~/Escritorio $ ls
ESCRITORIO  proyectoGit  tallergitdede0
james@james-pc ~/Escritorio $
```


git push



Actualiza el repositorio web con el repositorio local

```
Peppermint Terminal
james@james-pc ~/Escritorio/tallergitdede0 $ git push
Username for 'https://gitlab.com': JSibaja
Password for 'https://JSibaja@gitlab.com':
Contando objetos: 3, listo.
Escribiendo objetos: 100% (3/3), 211 bytes | 211.00 KiB/s, listo.
Total 3 (delta 0), reused 0 (delta 0)
To https://gitlab.com/JSibaja/tallergitdede0.git
 * [new branch]      master -> master
james@james-pc ~/Escritorio/tallergitdede0 $
```

git pull



Actualiza el repositorio local con el repositorio web

```
Peppermint Terminal
james@james-pc ~/Escritorio/tallergitdede0 $ git pull
Ya está actualizado.
james@james-pc ~/Escritorio/tallergitdede0 $
```



¿CÓMO INSTALARLO?

LECCIÓN 1: INTRODUCCIÓN

En Linux



```
sudo apt-get update  
sudo apt-get install git
```

En Windows



- Descargar <https://git-scm.com/download/win>
- Instalar
- *Seleccionar la opción de usar con la línea de comandos de unix

PRACTICA I: PRIMER REPOSITORIO

LECCIÓN 1: INTRODUCCIÓN