

Tiempo para aprender juntos

Dirección de Servicios de
Infraestructura y Operaciones
Septiembre 2020

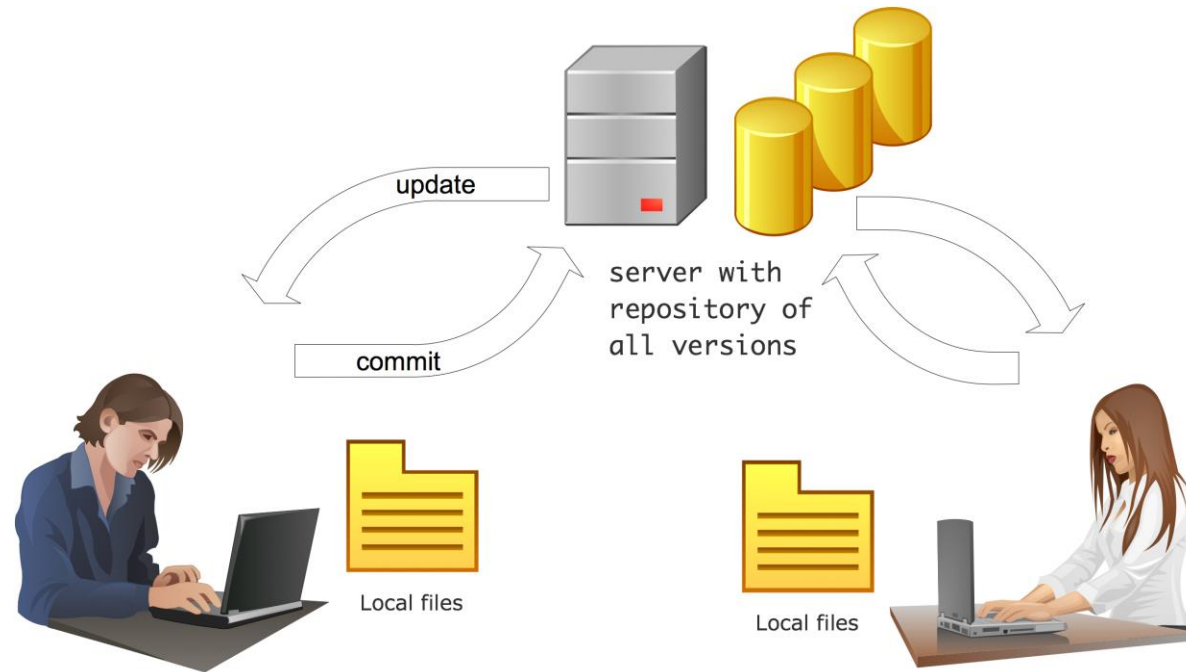


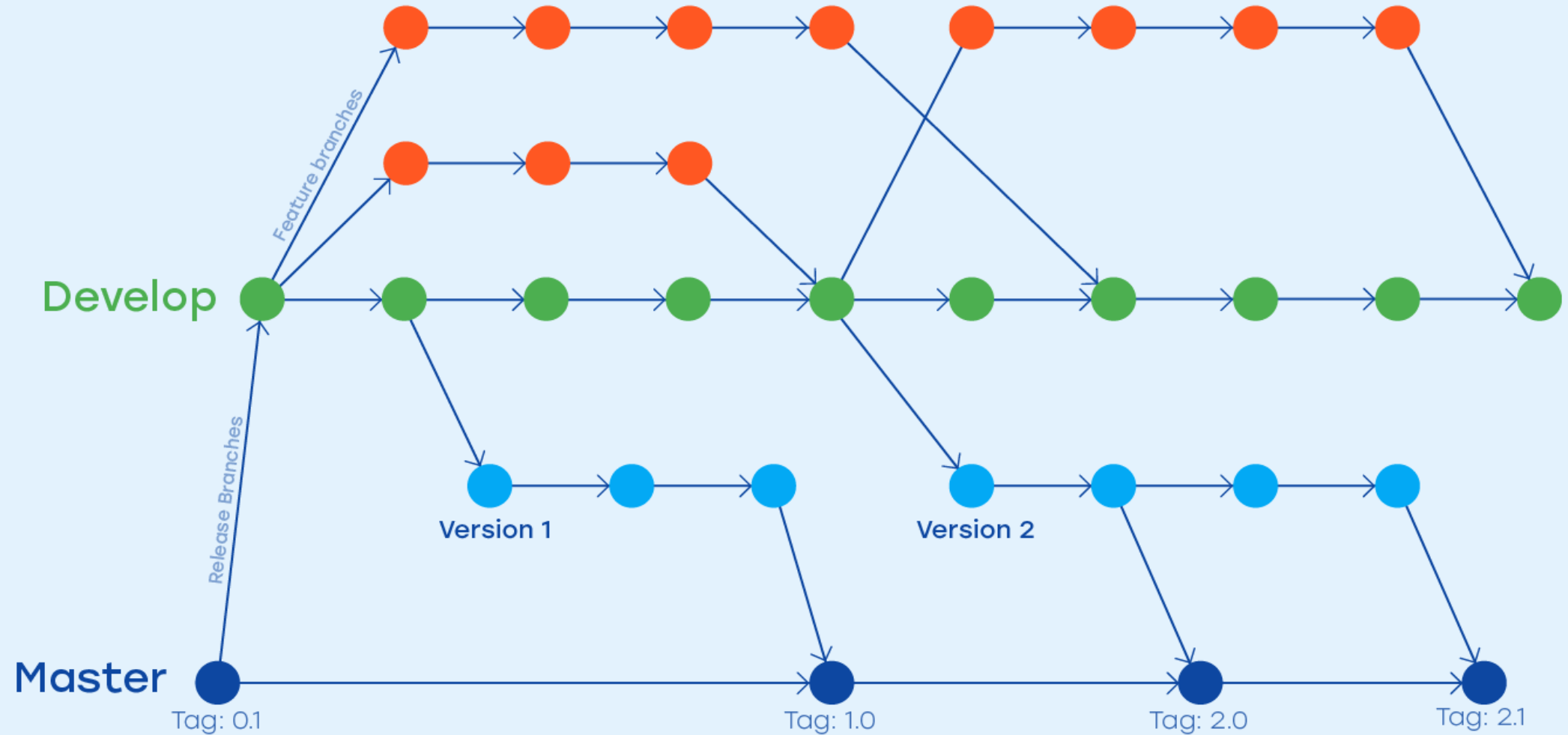
Source Code Management

- ¿Qué es SCM?
- ¿Qué es Git?
- ¿Qué es GitHub?
- Laboratorios
- Quiz

¿Qué es SCM?

- **Source Code Management (SCM)** es la practica encargada de dar *seguimiento y gestión* a los cambios hecho al código.
- Provee un historial de la ejecución del desarrollo de código.
- Ayuda a **resolver conflictos** al fusionar contribuciones de *múltiples fuentes*.





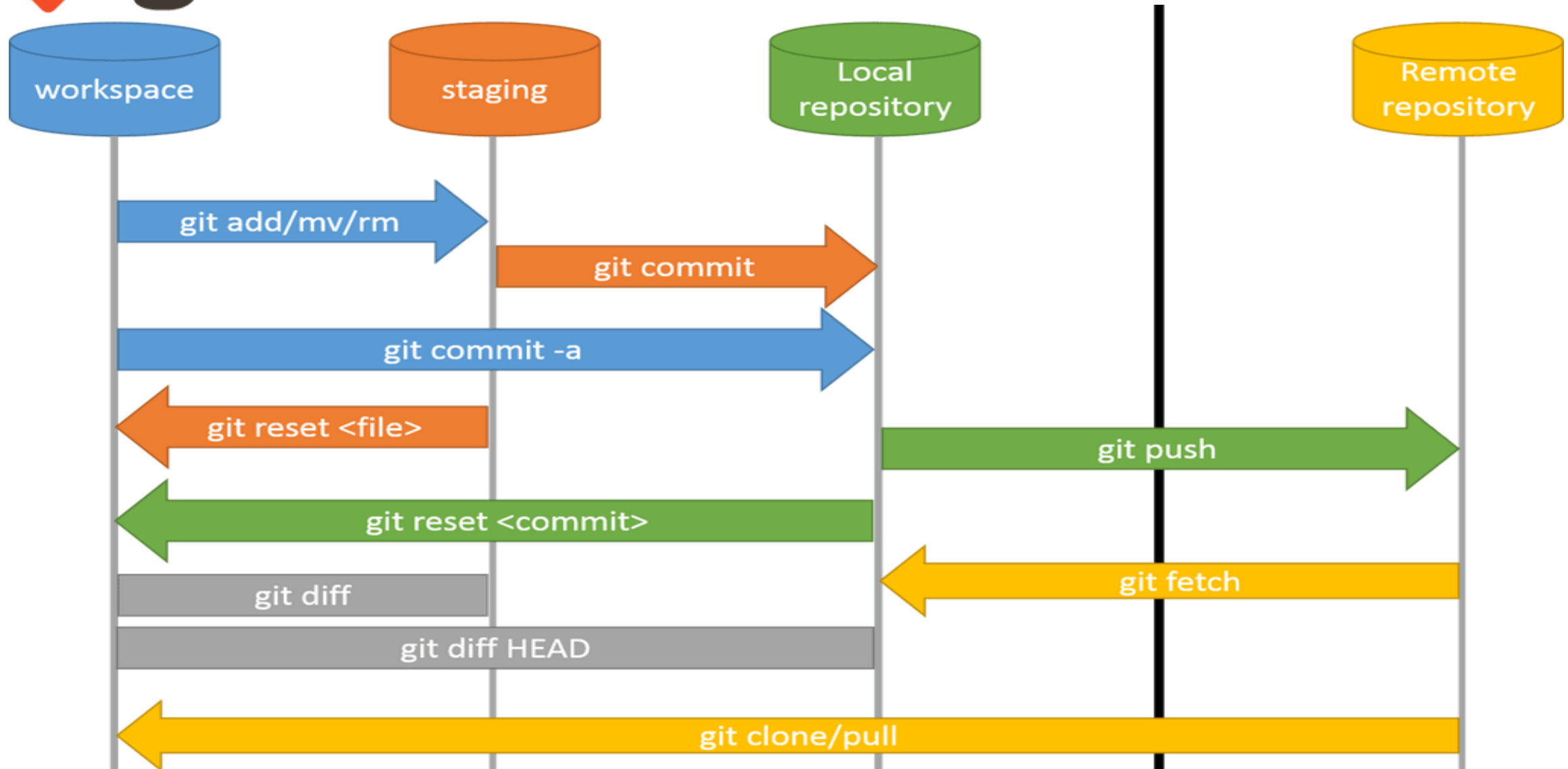
Standard development



- Es un SCM *open-source*.
- Permite crear una copia de tu repositorio (**branch**).
- Puedes trabajar en ajustar el código de manera independiente para su posterior actualización (**commit**).
- Puedes empujar (**push**) o jalar (**pull**) actualizaciones hacia/de otros contribuidores.
- Puedes fusionar los cambios en el repositorio central (**merge**).



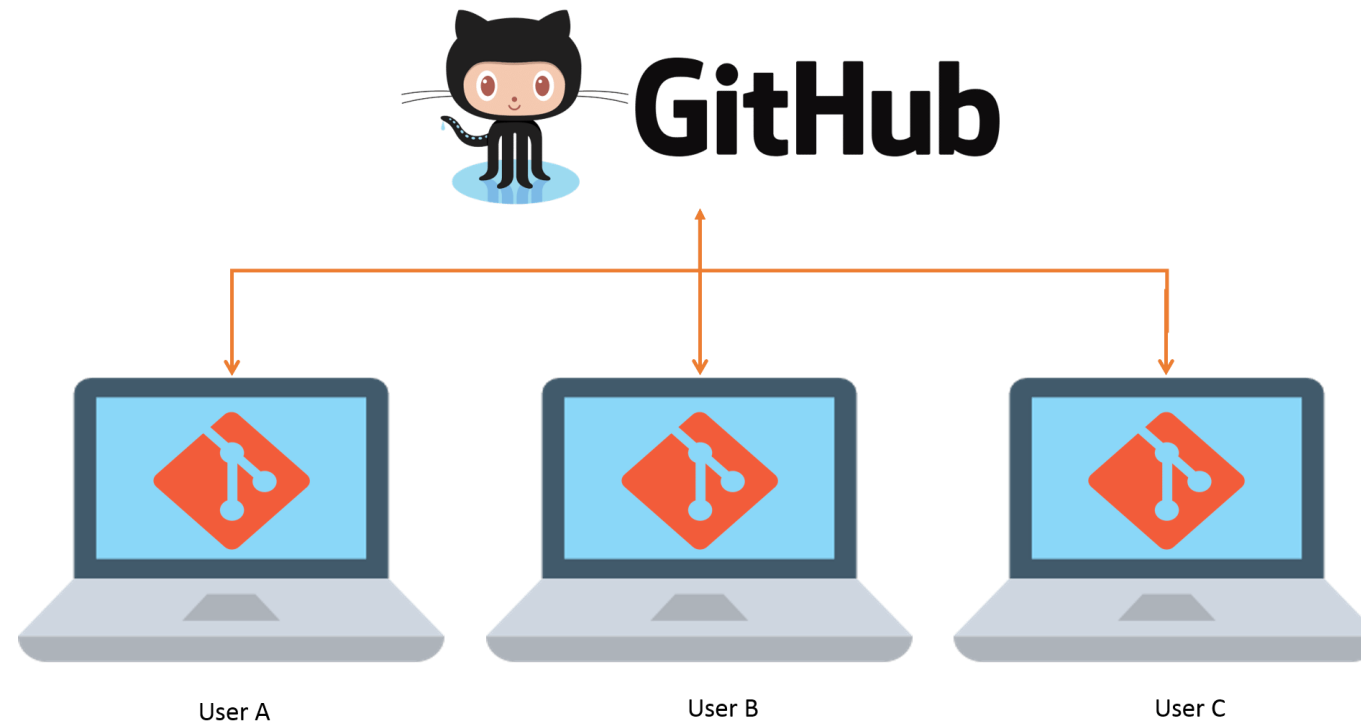
Comandos utilizados en Git



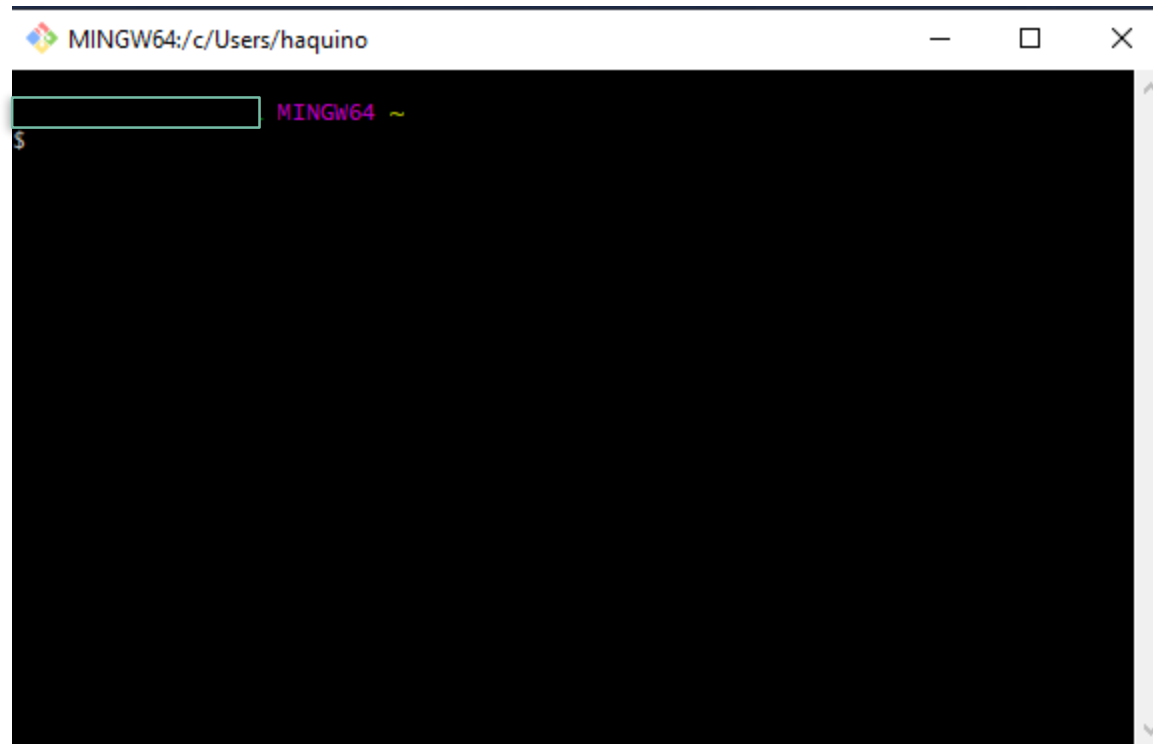


¿Qué es GitHub?

- Es una **plataforma** de **desarrollo colaborativo** para alojar proyectos utilizando Git.
- El código de los proyectos alojados en GitHub se almacena típicamente de forma **pública**.
- En 2018 Microsoft compró GitHub por **7,500 millones de dólares**.



- Instalar **Git** en tu laptop: <https://git-scm.com/download/win>
 - Instalarlo siguiendo las indicaciones del wizard
 - Abrir **Git Bash** o la consola para usar Git



- Comprobar la versión instalada: "git --version"

```
$ git --version  
git version 2.28.0.windows.1
```


- Configurar Git: nombre, correo electrónico y editor
 - `git config --global user.name "Tu nombre y apellido"`
 - `git config --global user.email "Tu dirección de correo electrónico personal"`
 - `git config --global core.editor "nano -w"`
- Para revisar la configuración
 - `git config --list`
 - `cat ~/.gitconfig`

```
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager
user.name=Hugo Aquino
user.email=hugo.aquino.navarrete@gmail.com
user.mail=hugo.aquino.navarrete@gmail.com
core.editor=nano -w
```

```
$ cat ~/.gitconfig
[user]
    name = Hugo Aquino
    email = hugo.aquino.navarrete@gmail.com
    mail = hugo.aquino.navarrete@gmail.com
[core]
    editor = nano -w
```

Laboratorio – Primer repositorio

- Ve al directorio \$HOME, crea un directorio llamado “repositorio”, ve a ese directorio y muestra en pantalla el directorio donde te encuentras
 - `cd ~`
 - `mkdir repositorio`
 - `cd repositorio`
 - `pwd`
- Valida el estado de git, ¿qué te aparece?
- Inicializa git y valida nuevamente su estado
 - `git init`
 - `git status`

```
MINGW64 ~/repositorio
$ git status
fatal: not a git repository (or any of the parent directories): .git

MINGW64 ~/repositorio
$ git init
Initialized empty Git repository in c:/Users/haquino/repositorio/.git/

MINGW64 ~/repositorio (master)
```

```
$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
```

Laboratorio – El directorio “.git”

- Observa el contenido del directorio oculto “.git”
 - `ls -la .git`

```
root@MRN-SDF1:~/gitrepo# tree ../.git/
```

```
../.git/
├── HEAD
├── branches
├── config
├── description
├── hooks
│   ├── applypatch-msg.sample
│   ├── commit-msg.sample
│   ├── fsmonitor-watchman.sample
│   ├── post-update.sample
│   ├── pre-applypatch.sample
│   ├── pre-commit.sample
│   ├── pre-merge-commit.sample
│   ├── pre-push.sample
│   ├── pre-rebase.sample
│   ├── pre-receive.sample
│   ├── prepare-commit-msg.sample
│   └── update.sample
├── info
│   └── exclude
├── objects
│   ├── info
│   └── pack
└── refs
    ├── heads
    └── tags
```

```
9 directories, 16 files
```

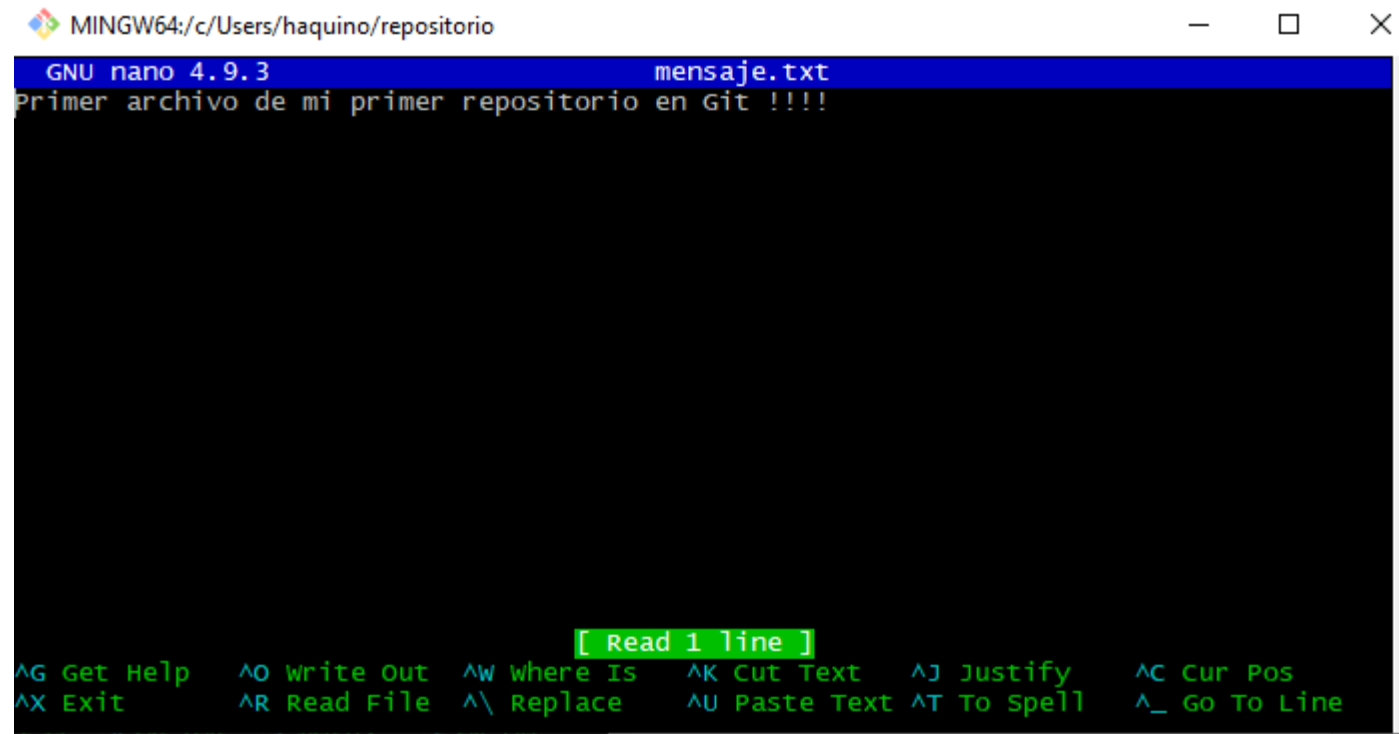
```
root@MRN-SDF1:~/gitrepo#
```

```
$ ls -la .git
```

```
total 11
```

drwxr-xr-x	1	1049089	0	Sep 18 18:03	./
drwxr-xr-x	1	1049089	0	Sep 18 18:03	../
-rw-r--r--	1	1049089	130	Sep 18 18:01	config
-rw-r--r--	1	1049089	73	Sep 18 18:01	description
-rw-r--r--	1	1049089	23	Sep 18 18:01	HEAD
drwxr-xr-x	1	1049089	0	Sep 18 18:01	hooks/
drwxr-xr-x	1	1049089	0	Sep 18 18:01	info/
drwxr-xr-x	1	1049089	0	Sep 18 18:01	objects/
drwxr-xr-x	1	1049089	0	Sep 18 18:01	refs/

- Crea un archivo llamado “mensaje.txt” y agrégale un sencillo texto
 - nano mensaje.txt



```
MINGW64:/c/Users/haquino/repositorio
GNU nano 4.9.3 mensaje.txt
Primer archivo de mi primer repositorio en Git !!!!

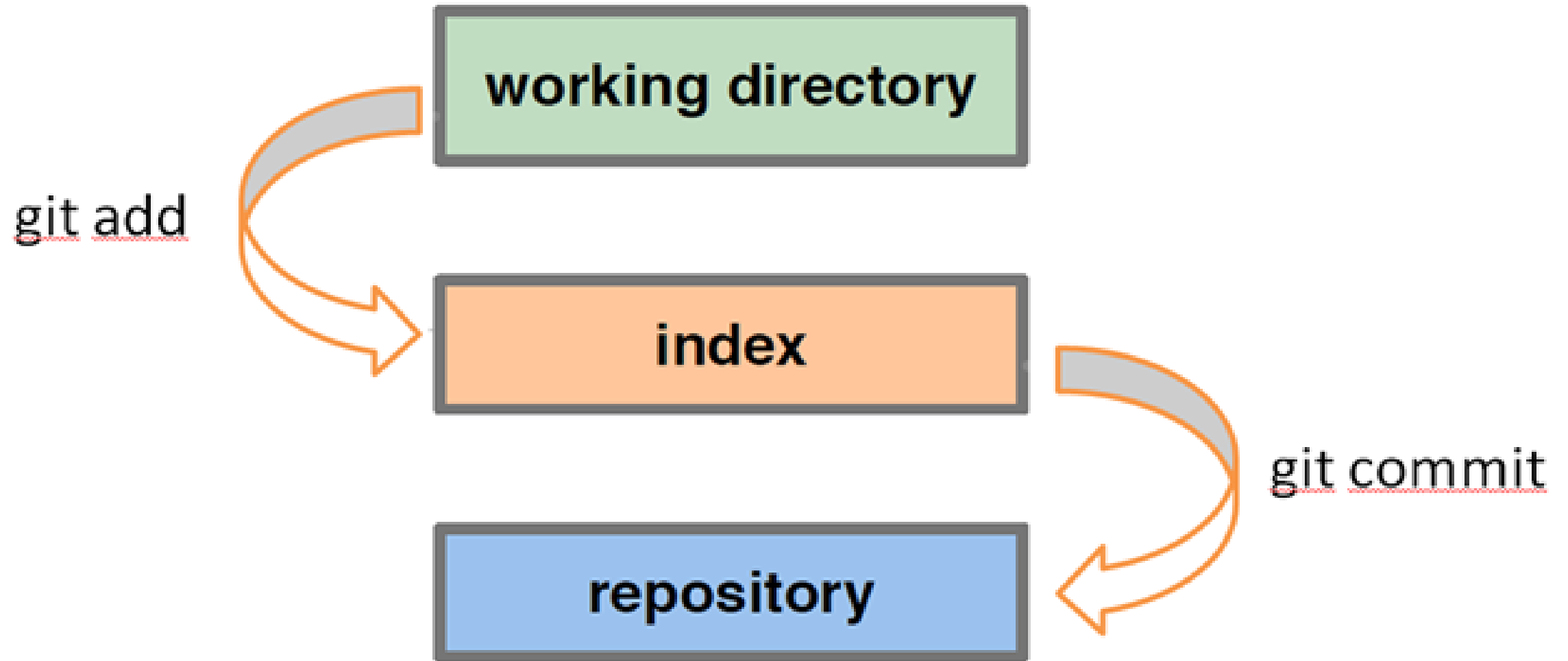
[ Read 1 line ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Paste Text ^T To Spell  ^_ Go To Line
```

- Para salvar la oración teclea “Control + O” seguido de un “Enter” y para salir “Control + X”
- Valida el contenido del archivo tecleando “cat mensaje.txt”

```
$ cat mensaje.txt
Primer archivo de mi primer repositorio en Git !!!!
```

Laboratorio – “git add y git commit”

- El movimiento de archivos entre directorios y los comandos que utilizaremos.



- Valida el status de git
 - `git status`

```
$ git status
on branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    mensaje.txt

nothing added to commit but untracked files present (use "git add" to track)
```

- Pon el archivo en el “**staging area**”
 - `git add mensaje.txt`

```
$ git add mensaje.txt
warning: LF will be replaced by CRLF in mensaje.txt.
The file will have its original line endings in your working directory

MINGW64 ~/repositorio (master)
$ git status
on branch master

No commits yet

changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   mensaje.txt
```

- Se creo un objeto nuevo bajo el directorio “objects”
 - `ls -l .git/objects/ee`

```
$ ls -l .git/objects/ee
total 1
-r--r--r-- 1 1049089 62 Sep 18 18:20 6ded383cf3f618205415669a803e882f4ef8e9
```

- Crea un commit agregando un mensaje “Iniciando notas del primer archivo en git”
 - `git commit -m “Iniciando notas del primer archivo en git”`

```
$ git commit -m "Iniciando notas del primer archivo en git"
[master (root-commit) 6e45418] Iniciando notas del primer archivo en git
1 file changed, 1 insertion(+)
create mode 100644 mensaje.txt
```

- Valida el estado del repositorio

- `git status`

```
$ git status
On branch master
nothing to commit, working tree clean
```

- Verifica el registro de los commits llevados a cabo

- `git log`

```
$ git log
commit 6a47849b0bb68d1d9b33b6ea2af62d920fe80326 (HEAD -> master)
Author: 
Date: 
    Iniciando notas del primer archivo en git
```

```
$ ls -lR
.:
total 0
drwxr-xr-x 1 0 0 6a/
drwxr-xr-x 1 0 0 b6/
drwxr-xr-x 1 0 0 ee/
drwxr-xr-x 1 0 0 info/
drwxr-xr-x 1 0 0 pack/

./6a:
total 1
-r--r--r-- 1 156 6a 47849b0bb68d1d9b33b6ea2af62d920fe80326

./b6:
total 1
-r--r--r-- 1 56 6a 64d4d32f49e6abd104059742513934e3065795

./ee:
total 1
-r--r--r-- 1 62 6a 6ded383cf3f618205415669a803e882f4ef8e9
```

- Observa el directorio “.git”

Laboratorio – Edita nuevamente archivo

- Agrega otro texto al archivo “mensaje.txt” y verifica el contenido del archivo
 - nano mensaje.txt
 - cat mensaje.txt
- Valida el estado del repositorio

- git status

```
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   mensaje.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

- Ve las diferencias entre el estado actual del archivo y la versión guardada más reciente

- git diff

```
$ git diff
warning: LF will be replaced by CRLF in mensaje.txt.
The file will have its original line endings in your working directory
diff --git a/mensaje.txt b/mensaje.txt
index ee6ded3..8ea6e75 100644
--- a/mensaje.txt
+++ b/mensaje.txt
@@ -1,2 @@
Primer archivo de mi primer repositorio en Git !!!!
+Estamos aprendiendo Git
```


- Haz otro commit agregando una breve descripción
 - `git commit -m "se agrega otra linea a mensaje.txt"`

```
$ git commit -m "Se agrega otra linea al archivo mensaje.txt"
on branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   mensaje.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

- Haz el commit agregando el archivo
 - `git add mensaje.txt`
 - `git commit -m "se agrega otra linea a mensaje.txt"`

```
$ git commit -m "se agrega otra linea a mensaje.txt"
[master f90686e] se agrega otra linea a mensaje.txt
1 file changed, 1 insertion(+)
```

- Mira el contenido del directorio ".git"

```
haquino@HDQITDIRO1 MINGW64 ~/repositorio/.git/objects (GIT_DIR!)
$ ls -lR
.:
total 0
drwxr-xr-x 1 0 Sep 18 18:34 6a/
drwxr-xr-x 1 0 Sep 18 18:51 8e/
drwxr-xr-x 1 0 Sep 18 18:51 ac/
drwxr-xr-x 1 0 Sep 18 18:34 b6/
drwxr-xr-x 1 0 Sep 18 18:31 ee/
drwxr-xr-x 1 0 Sep 18 18:51 f9/
drwxr-xr-x 1 0 Sep 18 18:31 info/
drwxr-xr-x 1 0 Sep 18 18:31 pack/
```

Laboratorio – Edita, staging y commit

- Agreguen otra línea al archivo “mensaje.txt”
 - `nano mensaje.txt`
- Revisa las diferencias
 - `git diff`
- Agrega el archivo al “staging area”
 - `git add mensaje.txt`
- Muestra la diferencia entre el último cambio que sí hizo commit y lo que está en el “staging area”
 - `git diff --staged`
- Haz el commit agregando el archivo
 - `git commit -m “Seguimos aprendiendo”`
- Revisa el status y los logs reduciendo la cantidad de información
 - `git status`
 - `git log --oneline`

```
diff --git a/mensaje.txt b/mensaje.txt
index 8ea6e75..bad7c07 100644
--- a/mensaje.txt
+++ b/mensaje.txt
@@ -1,2 +1,3 @@
 Primer archivo de mi primer repositorio en Git !!!!
 Estamos aprendiendo Git
+Entre mas se practica mejor se asimilan los conceptos
```

```
$ git log --oneline
64b346c (HEAD -> master) Seguimos aprendiendo
f90686e se agrega otra linea a mensaje.txt
6a47849 Iniciando notas del primer archivo en git
```

Laboratorio – Commit a varios archivos

- Agrega otra línea al archivo “mensaje.txt”
- Crea dos archivos adicionales “libro.txt” y “Netflix.txt” y agrégales una línea a cada uno
- Agrega los 3 archivos al “staging area”
- Haz el commit
- Revisa el status y los logs reduciendo la cantidad de información

```
$ git log --oneline
fb298b9 (HEAD -> master) Agregando 2 nuevos archivos: libro.txt y Netflix.txt y modificando mensaje.txt
64b346c Seguimos aprendiendo
f90686e se agrega otra linea a mensaje.txt
6a47849 Iniciando notas del primer archivo en git
```

Laboratorio – Explorando el History

- Uno puede ver los commits previos

```
$ git log --oneline
fb298b9 (HEAD -> master) Agregando 2 nuevos archivos: libro.txt y Netflix.txt y modificando mensaje.txt
64b346c Seguimos aprendiendo
f90686e se agrega otra linea a mensaje.txt
6a47849 Iniciando notas del primer archivo en git
```

- Una manera sencilla es a través del ID del commit (al menos los primeros 4 caracteres) y el nombre del archivo, por ejemplo
 - `git show fb29 libro.txt`
- Ahora sin querer queriendo, sobre escriban el archivo “mensaje.txt”
 - `echo “sin querer queriendo” > mensaje.txt`
- Valida el estado de git y el contenido de “mensaje.txt”

```
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   mensaje.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

```
$ git diff
warning: LF will be replaced by CRLF in mensaje.txt.
The file will have its original line endings in your working directory
diff --git a/mensaje.txt b/mensaje.txt
index b406261..c8e9a40 100644
--- a/mensaje.txt
+++ b/mensaje.txt
@@ -1,4 +1 @@
-Primer archivo de mi primer repositorio en Git !!!!
-Estamos aprendiendo Git
-Entre mas se practica mejor se asimilan los conceptos
-Seguimos practicando los comandos "git" para que no se olviden
+sin querer queriendo
```

Laboratorio – Explorando el History

- Para regresar a la versión anterior usaremos “git checkout <id_commit> <archivo>”

```
$ git log --oneline
fb298b9 (HEAD -> master) Agregando 2 nuevos archivos: libro.txt y Netflix.txt y modificando mensaje.txt
64b346c Seguimos aprendiendo
f90686e se agrega otra linea a mensaje.txt
6a47849 Iniciando notas del primer archivo en git
```

- git checkout fb29 mensaje.txt

- Valida el status de git y el contenido de “mensaje.txt”
 - git status
 - cat mensaje.txt

```
$ cat mensaje.txt
Primer archivo de mi primer repositorio en Git !!!!
Estamos aprendiendo Git
Entre mas se practica mejor se asimilan los conceptos
Seguimos practicando los comandos "git" para que no se olviden
```

Laboratorio – Deshaciendo un commit

- Sobre escribe el archivo “libro.txt” con un mensaje, agrégalo y haz un commit
 - echo “me equivoque” > libro.txt
 - git status
 - git add libro.txt
 - git diff --staged
 - git commit -m “Sigo agregando contenido”

```
$ git diff --staged
diff --git a/libro.txt b/libro.txt
index e75680e..cf08b6d 100644
--- a/libro.txt
+++ b/libro.txt
@@ -1,1 @@
-Leaders eat last - Simon Sinek
+me equivoque
```

```
$ git log --oneline
442d7d7 (HEAD -> master) Sigo agregando contenido
fb298b9 Agregando 2 nuevos archivos: libro.txt y Netflix.txt y modificando mensaje.txt
64b346c Seguimos aprendiendo
f90686e se agrega otra linea a mensaje.txt
6a47849 Iniciando notas del primer archivo en git
```

- Para regresar el estado anterior vamos hacer un checkout del HEAD -1
 - git checkout HEAD libro.txt
 - git checkout HEAD~1 libro.txt

```
$ git checkout HEAD libro.txt
Updated 0 paths from e5b59eb
```

```
$ git checkout HEAD~1 libro.txt
Updated 1 path from 5618d1d
```

- Valida el estado de git
 - git status

```
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   libro.txt
```

- Restaura del “staging area”, muestra los logs y el estado de git
 - git restore --staged libro.txt
 - git log --oneline
 - git status

```
$ git log --oneline
442d7d7 (HEAD -> master) Sigo agregando contenido
fb298b9 Agregando 2 nuevos archivos: libro.txt y Netflix.txt y modificando mensaje.txt
64b346c Seguimos aprendiendo
f90686e se agrega otra linea a mensaje.txt
6a47849 Iniciando notas del primer archivo en git

MINGW64 ~/repositorio (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   libro.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

Laboratorio – Deshaciendo un commit

- Ve las diferencias del último commit para el archivo libro.txt
 - `git diff 0941 libro.txt`

```
$ git diff 442d libro.txt
diff --git a/libro.txt b/libro.txt
index cf08b6d..e75680e 100644
--- a/libro.txt
+++ b/libro.txt
@@ -1,1 @@
-me equivoque
+Leaders eat last - Simon Sinek
```

- Agrega libro.txt y haz un commit
 - `git add libro.txt`
 - `git commit -m "Regresando libro.txt al contenido original"`
 - `git status`
 - `git log --oneline`

```
$ git commit -m "Regresando libro.txt al contenido original"
[master 13093da] Regresando libro.txt al contenido original
1 file changed, 1 insertion(+), 1 deletion(-)

MINGW64 ~/repositorio (master)

$ git status
On branch master
nothing to commit, working tree clean

MINGW64 ~/repositorio (master)

$ git log --oneline
13093da (HEAD -> master) Regresando libro.txt al contenido original
442d7d7 Sigo agregando contenido
fb298b9 Agregando 2 nuevos archivos: libro.txt y Netflix.txt y modificando mensaje.txt
64b346c Seguimos aprendiendo
f90686e se agrega otra linea a mensaje.txt
6a47849 Iniciando notas del primer archivo en git
```


Laboratorio – Creando un nuevo archivo antes de ir a GitHub

- Crea un archivo que se llame “README.md” y agrega brevemente algunas de tus habilidades técnicas.

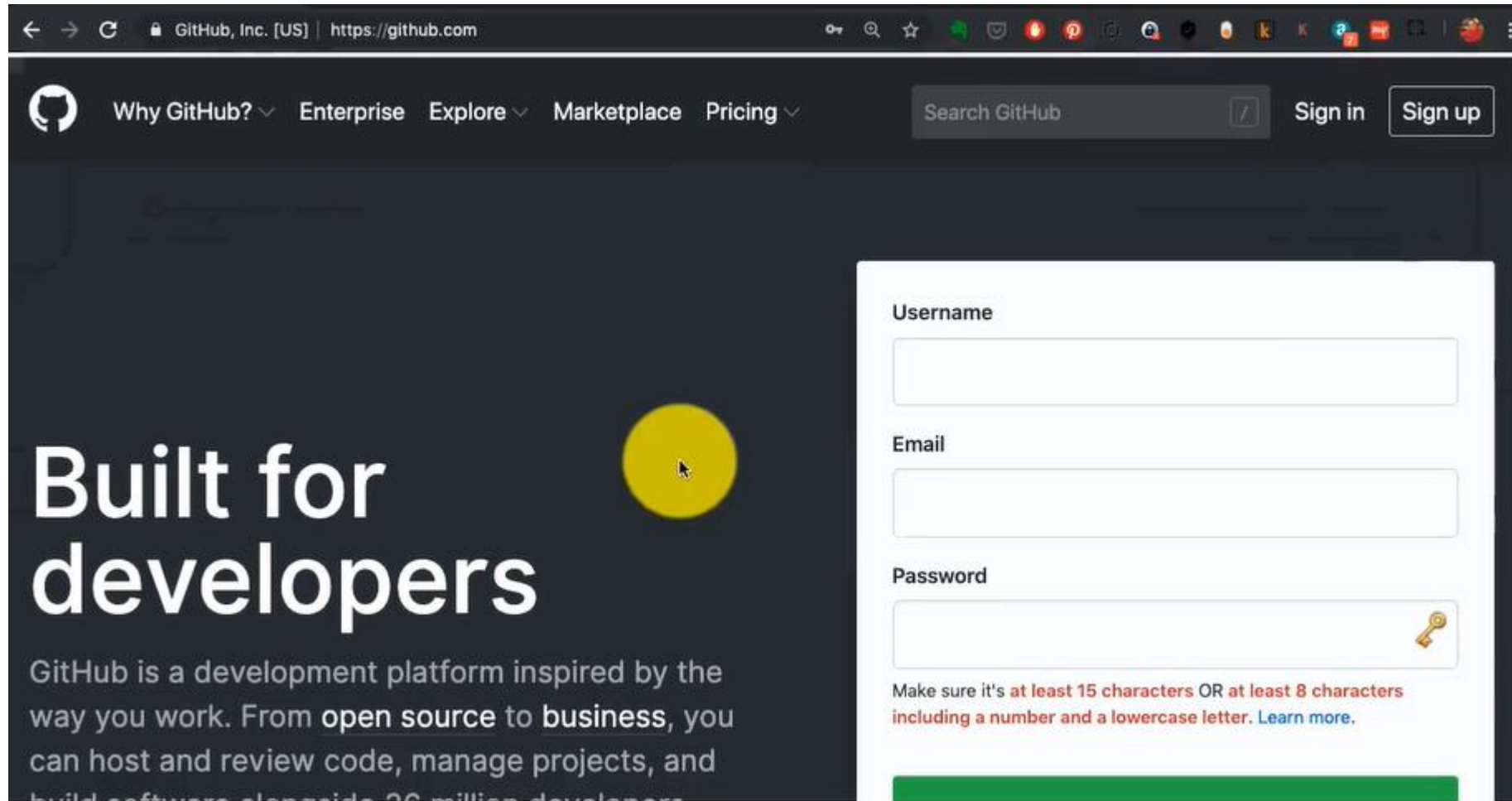
```
$ cat README.md
Habilidades Técnicas
- Programación: C/C++, Perl, Java, Python, R
- Routing & Switching
```

- Agrega el archivo, haz un commit y muestra los logs.

```
$ git log --oneline
a0f9853 (HEAD -> master) Agregando inventario de habilidades tecnicas
13093da Regresando libro.txt al contenido original
442d7d7 Sigo agregando contenido
fb298b9 Agregando 2 nuevos archivos: libro.txt y Netflix.txt y modificando mensaje.txt
64b346c Seguimos aprendiendo
f90686e se agrega otra linea a mensaje.txt
6a47849 Iniciando notas del primer archivo en git
```

Laboratorio – Conociendo GitHub

- Ve a <https://github.com/>, creen una cuenta usando su cuenta de correo personal.



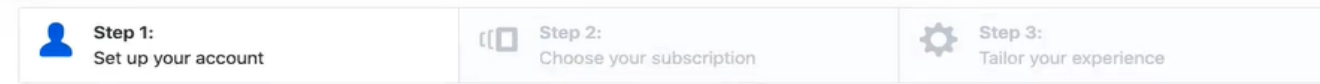
The screenshot shows the GitHub homepage in a web browser. The browser's address bar displays "GitHub, Inc. [US] | https://github.com". The navigation bar includes links for "Why GitHub?", "Enterprise", "Explore", "Marketplace", and "Pricing", along with a "Search GitHub" input field and "Sign in" and "Sign up" buttons. The main content area features the GitHub logo and the text "Built for developers". Below this, it states: "GitHub is a development platform inspired by the way you work. From open source to business, you can host and review code, manage projects, and build software alongside 26 million developers". On the right side, there is a sign-up form with the following fields and instructions:

- Username**: A text input field.
- Email**: A text input field.
- Password**: A text input field with a key icon on the right.
- Instructions**: "Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)"

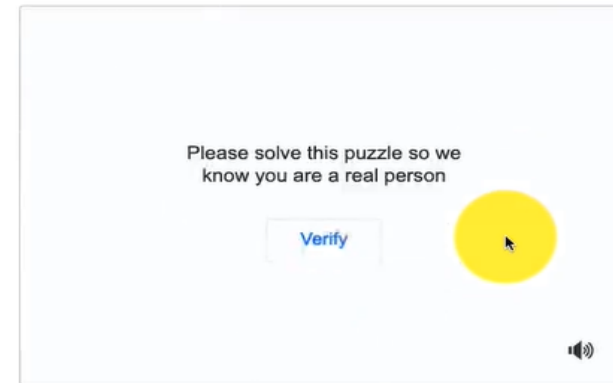
A green button is visible at the bottom of the sign-up form.

Laboratorio – Conociendo GitHub

- Verifica la cuenta, escogiendo la suscripción gratuita, contestando el survey y validando en tu correo.

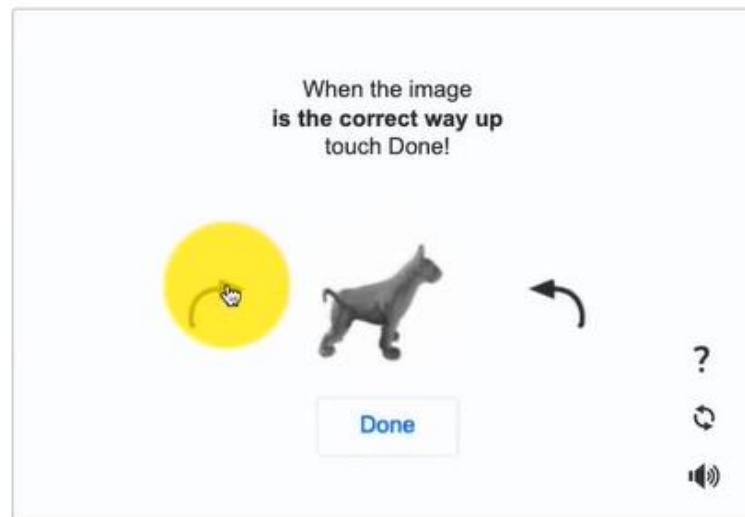


Verify account



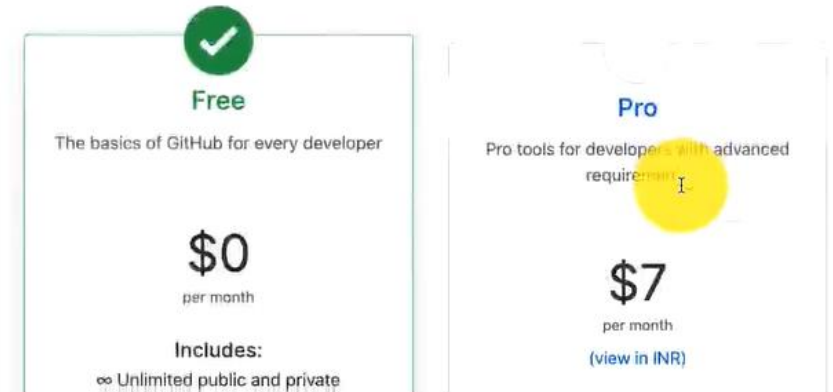
You'll love GitHub

- Unlimited public repositories
- Unlimited private repositories
- ✓ Limitless collaboration
- ✓ Frictionless development
- ✓ Open source community



Choose your subscription

With tools developers love and the world's largest open source community, there's no wrong choice.

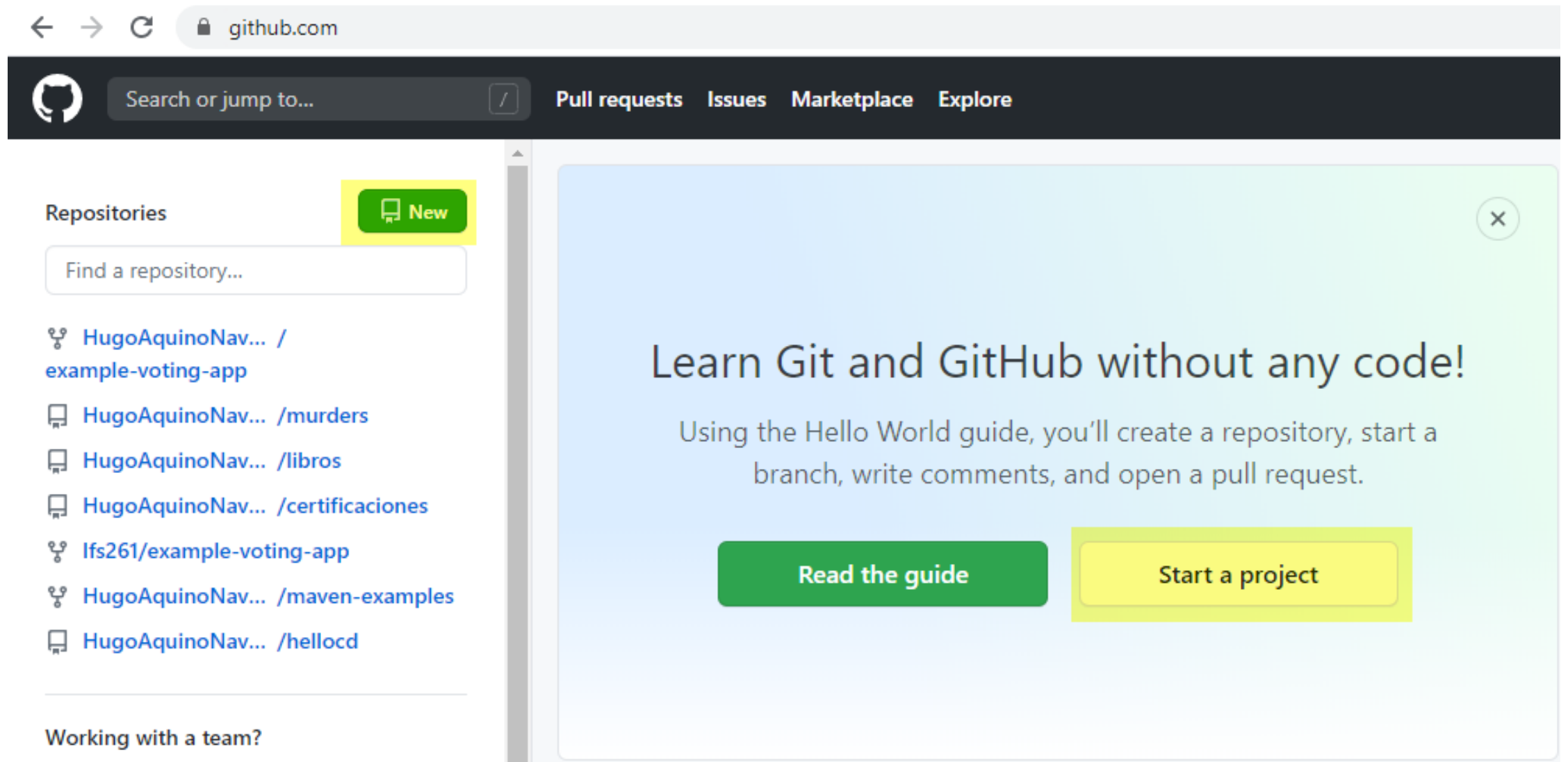


Please verify your email address

Before you can contribute on GitHub, we need you to verify your email address.

Laboratorio – Tu primer repositorio

- Una vez en GitHub, vamos a crear un repositorio



- Que se llame “repositorio”

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner *



HugoAquinoNavarrete ▾

Repository name *

/ repositorio ✓

Great repository names are short and memorable. Need inspiration? How about [musical-computing-machine?](#)

Description (optional)

Repositorio de pruebas para laboratorios de Git



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.



Add a README file

This is where you can write a long description for your project. [Learn more.](#)



Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)



Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

Create repository

Laboratorio – Tu primer repositorio

- Vamos a sincronizarlo con el que tenemos en la laptop

Quick setup — if you've done this kind of thing before



Set up in Desktop

or

HTTPS

SSH

<https://github.com/HugoAquinoNavarrete/repositorio.git>



Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# repositorio" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M master
git remote add origin https://github.com/HugoAquinoNavarrete/repositorio.git
git push -u origin master
```



...or push an existing repository from the command line

```
git remote add origin https://github.com/HugoAquinoNavarrete/repositorio.git
git branch -M master
git push -u origin master
```



...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

[Import code](#)

Laboratorio – Tu primer repositorio

- Vamos a sincronizarlo con el que tenemos en la laptop
 - `git remote show`
 - `git remote add origin https://github.com/HugoAquinoNavarrete/repositorio.git`
 - `git branch -M master`
 - `git push -u origin master`
 - `git remote show`

```
$ git remote show
1 MINGW64 ~/repositorio (master)
$ git remote add origin https://github.com/HugoAquinoNavarrete/repositorio.git
1 MINGW64 ~/repositorio (master)
$ git branch -M master
1 MINGW64 ~/repositorio (master)
$ git push -u origin master
Enumerating objects: 21, done.
Counting objects: 100% (21/21), done.
Delta compression using up to 4 threads
Compressing objects: 100% (15/15), done.
Writing objects: 100% (21/21), 1.94 KiB | 110.00 KiB/s, done.
Total 21 (delta 5), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (5/5), done.
To https://github.com/HugoAquinoNavarrete/repositorio.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.

1 MINGW64 ~/repositorio (master)
$ git remote show
origin

1 MINGW64 ~/repositorio (master)
$ git remote show origin
* remote origin
  Fetch URL: https://github.com/HugoAquinoNavarrete/repositorio.git
  Push URL: https://github.com/HugoAquinoNavarrete/repositorio.git
  HEAD branch: master
  Remote branch:
    master tracked
  Local branch configured for 'git pull':
    master merges with remote master
  Local ref configured for 'git push':
    master pushes to master (up to date)
```



Laboratorio – Tu primer repositorio

- Felicidades has creado tu primer repositorio en Git sincronizado con GitHub

The screenshot shows the GitHub interface for a repository named 'repositorio' by user 'HugoAquinoNavarrete'. The browser address bar shows 'github.com/HugoAquinoNavarrete/repositorio'. The repository page includes navigation tabs for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below these, it shows the 'master' branch with 1 branch and 0 tags. A commit history table lists recent changes, and the README.md file content is displayed below.

github.com/HugoAquinoNavarrete/repositorio

HugoAquinoNavarrete / repositorio

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

master 1 branch 0 tags Go to file Add file Code

	HugoAquinoNavarrete Agregando inventario de habilidades tecnicas	dcf737b 1 hour ago	7 commits
Netflix.txt	Se edita mensaje.txt y se agregan 2 nuevos archivos	12 hours ago	
README.md	Agregando inventario de habilidades tecnicas	1 hour ago	
libro.txt	Regresando libro.txt al contenido original	1 hour ago	
mensaje.txt	Se edita mensaje.txt y se agregan 2 nuevos archivos	12 hours ago	

README.md

Habilidades Técnicas

- Programación: C/C++, Perl, Java, Python, R
- Routing & Switching

Laboratorio – Vamos por más – Clonar repositorio

- Vamos a clonar un repositorio para mostrar el trabajo colaborativo en GitHub y Git.
- En la consola de Git en tu laptop, ve al directorio HOME, crea un directorio que se llame “projects”, clona el repositorio “git_concepts”, ve al directorio “git_concepts” y has un listado recursivo de su contenido.
 - `cd ~`
 - `mkdir projects`
 - `git clone https://github.com/HugoAquinoNavarrete/git_concepts.git`
 - `cd git_concepts`
 - `ls -lR`

```
MINGW64 ~/projects/git_concepts (master)
$ ls -lR
.:
total 5
drwxr-xr-x 1  1049089  0 Sep  5 14:17 cfg/
drwxr-xr-x 1  1049089  0 Sep  5 14:17 img/
-rw-r--r-- 1  1049089 405 Sep  5 14:17 README.md

./cfg:
total 1
-rw-r--r-- 1  1049089 41 Sep  5 14:17 comica.txt

./img:
total 256
-rw-r--r-- 1  1049089 104546 Sep  5 14:17 Branches.png
-rw-r--r-- 1  1049089  79765 Sep  5 14:17 Comandos_git.png
-rw-r--r-- 1  1049089  39841 Sep  5 14:17 GitHub.png
-rw-r--r-- 1  1049089  31182 Sep  5 14:17 Uso_colaborativo_de_Git.jpg
```



- Crea un Branch que se llame “test”
 - `git checkout -b test`

```
MINGW64 ~/projects/git_concepts/cfg (master)
$ git checkout -b test
Switched to a new branch 'test'

MINGW64 ~/projects/git_concepts/cfg (test)
```

- `git branch`

```
$ git branch
master
* test
```

- Edita el archivo “comica.txt” y agrega la información que se solicita, agrégalo, haz un commit e intenta hacer un push al repositorio en GitHub
 - `cd cfg`
 - `nano comica.txt`
 - `git commit -am "Agregando comica de"`
 - `git push origin test`

```
ubuntu@ubuntu-lts:~/repositorios/git_concepts/cfg$ git push origin test
Username for 'https://github.com': 
Password for 'https://@github.com': 
remote: Permission to HugoAquinoNavarrete/git_concepts.git denied to 
fatal: unable to access 'https://github.com/HugoAquinoNavarrete/git_concepts.git/': The requested URL returned error: 403
```

Laboratorio – Permisos para colaborar en el repositorio

- Por el chat favor escriban su nombre de usuario en GitHub para agregarlos, les llegará un correo para que por favor lo aprueben

The screenshot shows the GitHub repository settings for 'git_concepts' by user 'HugoAquinoNavarrete'. The 'Settings' tab is selected in the top navigation bar. On the left, a sidebar contains various settings categories: Options, Manage access (highlighted), Security & analysis, Branches, Webhooks, Notifications, Integrations, Deploy keys, Secrets, Actions, Moderation, and Interaction limits. The main content area is titled 'Who has access' and shows two sections: 'PUBLIC REPOSITORY' and 'DIRECT ACCESS'. The 'DIRECT ACCESS' section indicates that 2 users have access to the repository, with 1 collaborator and 1 invitation. Below this, the 'Manage access' section is visible, featuring a search bar to find collaborators and a list of current collaborators. One collaborator is listed with a checkbox, a GitHub icon, a redacted name, and a 'Pending Invite' status. A second entry is partially visible below it.

The screenshot shows an email invitation from GitHub. At the top, it features the profile picture of 'HugoAquinoNavarrete' and the GitHub logo. The main text reads: '@HugoAquinoNavarrete has invited you to collaborate on the HugoAquinoNavarrete/git_concepts repository'. Below this, it states: 'You can [accept or decline](#) this invitation. You can also head over to https://github.com/HugoAquinoNavarrete/git_concepts to check out the repository or visit [@HugoAquinoNavarrete](#) to learn a bit more about them.' A green button labeled 'View invitation' is prominently displayed. At the bottom, a note says: 'Note: This invitation was intended for [redacted] you were not expecting this invitation, you can ignore this email. If @HugoAquinoNavarrete is sending you too many emails, you can [block them](#) or [report abuse](#).' Another green button labeled 'View invitation' is located at the bottom right.

Laboratorio – Permisos para colaborar en el repositorio

- Intenta nuevamente hacer el “push”
 - `git push origin test`

```
ubuntu@ubuntu-lts:~/repositorios/git_concepts/cfg$ git push origin test
Username for 'https://github.com': 
Password for 'https://[redacted]@github.com': 
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 392 bytes | 98.00 KiB/s, done.
Total 4 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
```

HugoAquinoNavarrete / **git_concepts** Watch 1 Star 0 Fork 0

<> Code Issues Pull requests Actions Projects Wiki Security Insights

test had recent pushes 21 minutes ago [Compare & pull request](#)

test 2 branches 0 tags [Go to file](#) [Add file](#) [Code](#)

This branch is 1 commit ahead of master. [Pull request](#) [Compare](#)

Commit	Author	Message	Time
c848cef	AndreaAquinoR	Agregando comica de Andrea	33 minutes ago
			10 commits

File	Message	Time
cfg	Agregando comica de Andrea	33 minutes ago
doc	Agregando material de la sesion	8 days ago
img	Repositorio creado con README.me, imágenes y archivo para el labo...	14 days ago
README.md	Acomodando nuevamente el archivo README.md	14 days ago

About
Repositorio para el laboratorio colaborativo de Git
[Readme](#)

Releases
No releases published
[Create a new release](#)

Packages
No packages published
[Publish your first package](#)

nes

Laboratorio – Crear pull requests

- Observa lo que aparece en GitHub

The screenshot shows the GitHub interface for the repository 'HugoAquinoNavarrete/git_concepts'. At the top, there are navigation tabs: Code, Issues, Pull requests, Actions, Projects, Wiki, Security, and Insights. The 'Pull requests' tab is selected. Below the navigation bar, there's a yellow banner indicating that 'test' had recent pushes 21 minutes ago, with a 'Compare & pull request' button. Below this, there's a section showing 'test' with 2 branches and 0 tags, along with buttons for 'Go to file', 'Add file', and 'Code'. A status bar indicates 'This branch is 1 commit ahead of master.' with links for 'Pull request' and 'Compare'. The main content area shows a list of commits. The latest commit by 'AndreaAquinoR' is titled 'Agregando comica de Andrea' with commit hash 'c848cef', pushed 33 minutes ago, and has 10 commits in total. Below the commit list, there's a table showing the repository's structure:

cfg	Agregando comica de Andrea	33 minutes ago
doc	Agregando material de la sesion	8 days ago
img	Repositorio creado con README.me, imágenes y archivo para el labo...	14 days ago
README.md	Acomodando nuevamente el archivo README.md	14 days ago

On the right side, there's an 'About' section describing the repository as 'Repositorio para el laboratorio colaborativo de Git' with a 'Readme' link. Below that, there's a 'Releases' section stating 'No releases published' with a 'Create a new release' link. At the bottom, there's a 'Packages' section stating 'No packages published' with a 'Publish your first package' link.

Laboratorio – Crear pull requests

- Crea un “pull request”, agregando un texto y habilita a “HugoAquinoNavarrete” como “revisor”

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).


↺

base: master

←

compare: test









✓ **Able to merge.** These branches can be automatically merged.



Agregando comica de Andrea

Write

Preview

H B I  < >       


revisar comica preferida de Andrea

Attach files by dragging & dropping, selecting or pasting them.

Create pull request

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

Reviewers

 HugoAquinoNav...

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Milestone

No milestone

Linked issues

Use [Closing keywords](#) in the description to automatically close issues

- Así se ve tu GitHub

The screenshot displays a GitHub Pull Request (PR) interface. At the top, navigation tabs include 'Conversation' (0), 'Commits' (1), 'Checks' (0), and 'Files changed' (1). A status bar on the right shows '+1 -0' and a progress indicator. The main content area shows a commit history with a commit titled 'revisar comica preferida de Andrea' and a commit titled 'Agregando comica de Andrea' (c840cef). A comment from a collaborator says 'revisar comica preferida de Andrea'. Below the commit history, a message states 'Add more commits by pushing to the test branch on HugoAquinoNavarrete/git_concepts'. A warning box indicates 'This branch has not been deployed' with 'No deployments'. A green box highlights the 'Review requested' status, stating 'Review has been requested on this pull request. It is not required to merge. Learn more.' and '1 pending reviewer'. Below this, a green checkmark indicates 'This branch has no conflicts with the base branch' and 'Merging can be performed automatically.' A green button labeled 'Merge pull request' is visible, along with a link to 'view command line instructions.' On the right sidebar, sections include 'Reviewers' (HugoAquinoNav...), 'Assignees' (No one—assign yourself), 'Labels' (None yet), 'Projects' (None yet), 'Milestone' (No milestone), 'Linked issues' (Successfully merging this pull request may close these issues), and 'Notifications' (Customize, Unsubscribe). At the bottom, a note says 'You're receiving notifications because you'.

Laboratorio – Pull request

- Para aprobar los cambios

HugoAquinoNavarrete / git_concepts

<> Code ⓘ Issues 🔄 Pull requests 1 ⚙️ Actions 📁 Projects 📖 Wiki ⓘ Security 📈 Insights ⚙️ Settings

👤 master 2 branches 0 tags Go to file Add file ↕️ Code

HugoAquinoNavarrete Set theme jekyll-theme-architect ✓ 2fa5677 8 days ago ⌚ 9 commits

📁	cfg	Agregando otra comica	14 days ago
📁	doc	Agregando material de la sesion	8 days ago
📁	img	Repositorio creado con README.me, imágenes y archivo para el laborato...	14 days ago
📄	README.md	Acomodando nuevamente el archivo README.md	14 days ago
📄	_config.yml	Set theme jekyll-theme-architect	8 days ago

HugoAquinoNavarrete / git_concepts Unwatch 1

<> Code ⓘ Issues 🔄 Pull requests 1 ⚙️ Actions 📁 Projects 📖 Wiki ⓘ Security 📈 Insights ⚙️ Settings


Label issues and pull requests for new contributors Dismiss
Now, GitHub will help potential first-time contributors [discover issues](#) labeled with [good first issue](#)

Filters ⓘ is:pr is:open Labels 9 Milestones 0 New pull request

☐ 🔄 1 Open ✓ 2 Closed Author Label Projects Milestones Reviews Assignee Sort

☐ 🔄 Agregando comica de [REDACTED]
#3 opened 6 minutes ago by AndreaAquinoR

- Para aprobar los cambios

 HugoAquinoNavarrete / git_concepts Unwatch 1


[Code](#) [Issues](#) [Pull requests 1](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

AndreaAquinoR requested your review on this pull request. [Add your review](#)


Agregando comica de [redacted] #3 [Edit](#) [Open with](#)


[Open](#) [redacted] wants to merge 1 commit into [master](#) from [test](#)

[Conversation 0](#) [Commits 1](#) [Checks 0](#) [Files changed 1](#) +1 -0


 [redacted] commented 8 minutes ago Collaborator


revisar comica preferida de Andrea



 Agregando comica de Andrea c848cef


 [redacted] requested a review from HugoAquinoNavarrete 8 minutes ago


Add more commits by pushing to the [test](#) branch on [HugoAquinoNavarrete/git_concepts](#).


 **This branch has not been deployed**
No deployments


 **Review requested** [Show all reviewers](#)
Review has been requested on this pull request. It is not required to merge. [Learn more.](#)

Reviewers 
 HugoAquinoNava...
Still in progress? [Convert to draft](#)

Assignees 
No one—[assign yourself](#)

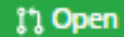
Labels 
None yet

Projects 
None yet

Milestone 
No milestone

- Para aprobar los cambios

Agregando comica de [REDACTED] #3

[Edit](#)[Open with](#)

AndreaAquinoR wants to merge 1 commit into `master` from `test`

Conversation 0

Commits 1

Checks 0

Files changed 1

+1 -0

Changes from all commits File filter... Jump to... ⚙

0 / 1 files viewed ⓘ

Review changes

1 cfg/comica.txt

```
... @@ -1,3 +1,4 @@
1 1 nombre,comica,edad
2 2 Hugo,Thundercats,10
3 3 Hugo,El pajaro loco,8
4 + Andrea,Backyardigans,5
```

ProTip! Use `n` and

Write Preview

H B I

OK, de acuerdo

Attach files by dragging & dropping, selecting or pasting them.

☐ Comment

Submit general feedback without explicit approval.

☒ Approve

Submit feedback and approve merging these changes.

☐ Request changes

Submit feedback that must be addressed before merging.

Submit review

- Para aprobar los cambios

The screenshot displays a GitHub pull request interface. At the top, a notification states that 'HugoAquinoNavarrete approved these changes now' with a 'View changes' link. Below this, a comment box shows a comment from the owner: 'OK, de acuerdo'. A message indicates that more commits can be added by pushing to the 'test' branch on the repository 'HugoAquinoNavarrete/git_concepts'. A status box notes that 'This branch has not been deployed' with no deployments. The main section, titled 'Changes approved', shows '1 approving review' and '1 approval'. It also confirms 'This branch has no conflicts with the base branch' and that merging can be performed automatically. A green 'Merge pull request' button is highlighted. Below this, a merge summary is shown: 'Merge pull request #3 from HugoAquinoNavarrete/test' and 'Agregando comica de Andrea'. At the bottom of the summary, there are 'Confirm merge' and 'Cancel' buttons. A final green notification box at the bottom states 'Pull request successfully merged and closed' and 'You're all set—the test branch can be safely deleted.', with a 'Delete branch' link.

HugoAquinoNavarrete approved these changes now [View changes](#)

HugoAquinoNavarrete left a comment [Owner](#) 😊 ⋮

OK, de acuerdo

Add more commits by pushing to the `test` branch on `HugoAquinoNavarrete/git_concepts`.

This branch has not been deployed
No deployments

Changes approved [Show all reviewers](#)
1 approving review [Learn more](#)

✓ 1 approval

✓ **This branch has no conflicts with the base branch**
Merging can be performed automatically.

Merge pull request ⌵ You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Merge pull request #3 from HugoAquinoNavarrete/test

Agregando comica de Andrea

Confirm merge Cancel

Pull request successfully merged and closed [Delete branch](#)
You're all set—the `test` branch can be safely deleted.

Laboratorio – Sincroniza los cambios

- En tu repositorio local, verifica en que “branch” te encuentras
 - `git branch`
- Si aun te aparece “test” elimínalo así:
 - `git branch -D test`

```
$ git branch -D test
Deleted branch test (was 56e86e9).
```

- Sincroniza el repositorio local con el branch “Master”
 - `git pull origin master`

```
$ git pull origin master
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 5 (delta 1), reused 4 (delta 1), pack-reused 0
Unpacking objects: 100% (5/5), 999 bytes | 16.00 KiB/s, done.
From https://github.com/HugoAquinoNavarrete/git_concepts
 * branch                master      -> FETCH_HEAD
   2fa5677..7633a91      master      -> origin/master
Updating 2fa5677..7633a91
Fast-forward
 cfg/comica.txt | 1 +
 1 file changed, 1 insertion(+)
```

- Verifica el contenido del archivo “comica.txt”

- Tiempo para repasar jugando



Gracias

