

GIT, CÓMO USAR

Fernando Hernández



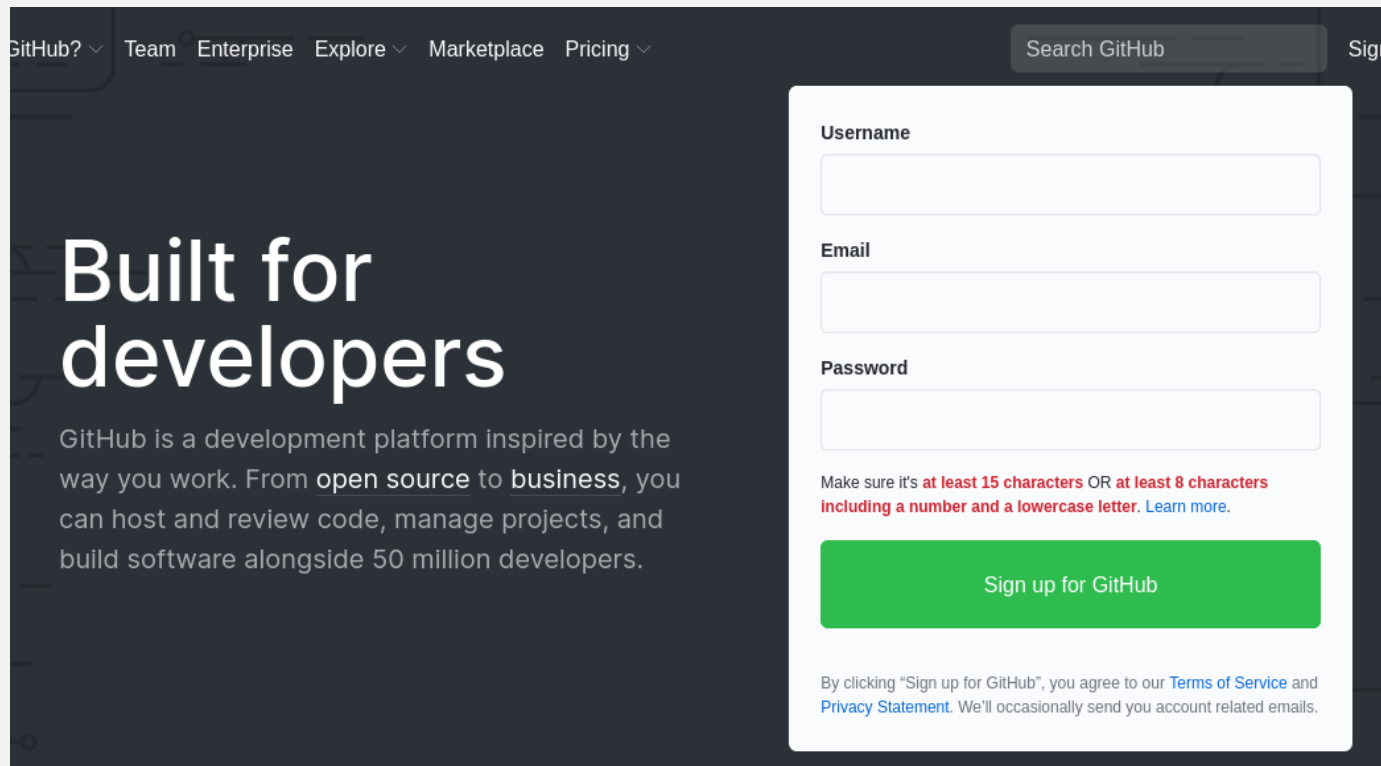
QUÉ ES GIT?

- Es un software de control de versiones.
- Su propósito es tener el registro de los cambios de en archivos de computadora y coordinar el trabajo que varias personas pueden estar realizando.
- Puede trabajar con repositorios locales o remotos.
- Es muy útil cuando el individuo o el grupo de personas sabe que va a estar realizando cambios en el software con el tiempo.



GITHUB

Es una plataforma de desarrollo colaborativo que utiliza el sistema de control de versiones de Git.

A screenshot of the GitHub website's sign-up page. The background is dark with the text "Built for developers" in large white font. Below it, a paragraph describes GitHub as a development platform. On the right, there is a white sign-up form with fields for Username, Email, and Password. A green button labeled "Sign up for GitHub" is at the bottom of the form. Above the button, there is a note about password requirements. The top of the page shows navigation links like "Team", "Enterprise", "Explore", "Marketplace", and "Pricing", along with a search bar and a "Sign in" link.

GitHub? ▾ Team Enterprise Explore ▾ Marketplace Pricing ▾ Search GitHub Sign in

Built for developers

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 50 million developers.

Username

Email

Password

Make sure it's **at least 15 characters** OR **at least 8 characters including a number and a lowercase letter**. [Learn more](#).

Sign up for GitHub

By clicking "Sign up for GitHub", you agree to our [Terms of Service](#) and [Privacy Statement](#). We'll occasionally send you account related emails.



CÓMO INSTALAR GIT

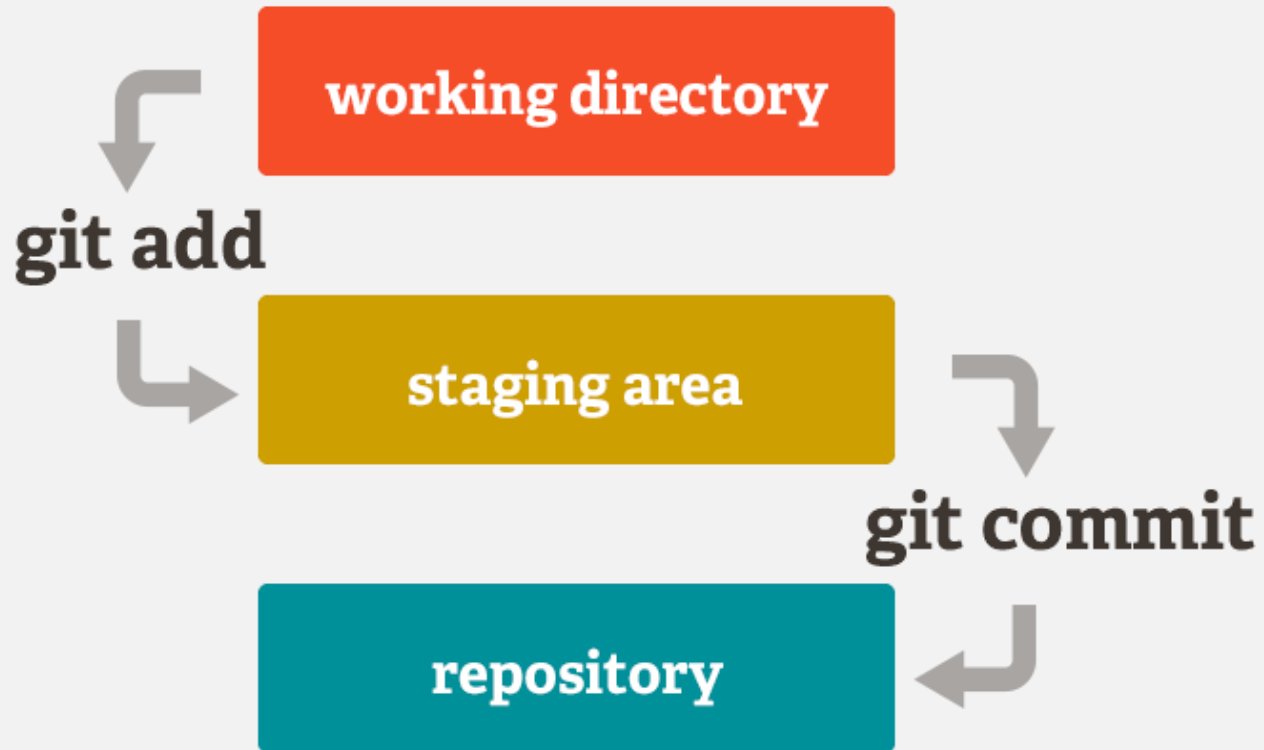
```
+ x Desktop Home  
fernando-h@fernando-h:~/Desktop$ sudo apt install git
```

Para verificar que esté instalada correctamente:

```
fernando-h@fernando-h:~/Desktop$ git --version  
git version 2.17.1
```

ESTADOS DE GIT

- Working directory: Donde cada quien individualmente trabaja con sus archivos.
- Staging area: Donde se agregan los archivos que se preparan para finalmente lanzarlos al repositorio.
- Repository: Lugar en donde se encuentran las versiones de software que decidimos subir.



COMANDOS BÁSICOS

Este comando siempre nos ayuda cuando se nos olvida algún comando o cuando queremos saber más sobre alguno en específico.

Lista los comandos básicos y qué hacen.

```
fernando-h@fernando-h:~/Desktop$ git --help
uso: git [--version] [--help] [-C <ruta>] [-c <nombre>=<valor>]
      [--exec-path[=<ruta>]] [--html-path] [--man-path] [--info-path]
      [-p | --paginate | --no-pager] [--no-replace-objects] [--bare]
      [--git-dir=<ruta>] [--work-tree=<ruta>] [--namespace=<nombre>]
      <comando> [<args>]
```

CONFIGURACIÓN DE GIT

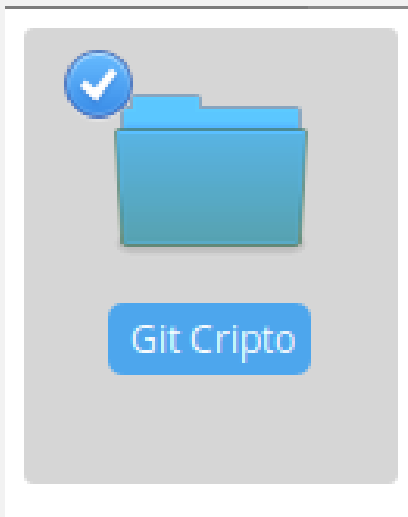
Necesitas tener una cuenta en GitHub para poder configurar el usuario. Cada vez que se hace un commit, el Git pedirá tu usuario o correo, para que el sistema sepa quién hizo cierto cambio en las carpetas.

Estos comandos se recomiendan ejecutar siempre en nuestra computadora personal.

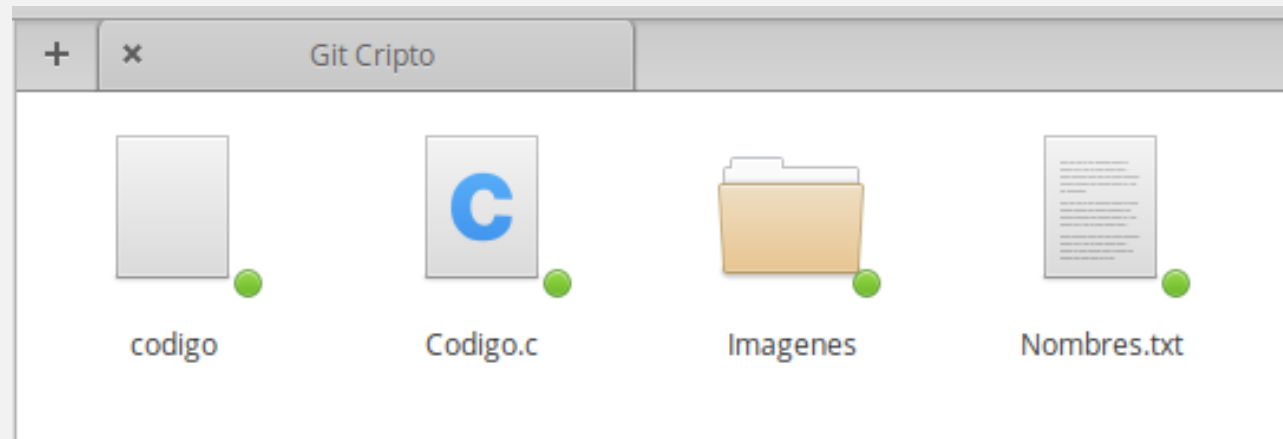
```
fernando-h@fernando-h:~/Desktop/Git Cripto$ git config --global user.email "hernandezF@live.com.mx"  
fernando-h@fernando-h:~/Desktop/Git Cripto$ git config --global user.name "FerHernandez13"
```

INICIAR UN PROYECTO

Creamos una nueva carpeta y ahí abrimos la terminal.
Procedemos a crear un proyecto con el comando init.
Justo en la carpeta donde hicimos el init, es nuestro Working Directory.



```
fernando-h@fernando-h:~/Desktop/Git Cripto$ git init  
Inicializado repositorio Git vacío en /home/fernando-h/Desktop/Git Cripto/.git/
```



COMANDO GIT STATUS

Los archivos en rojo quieren decir que están en el Working Directory, aún no pasan al Staging Area.

```
+ x Git Cripto: git
fernando-h@fernando-h:~/Desktop/Git Cripto$ git status
En la rama master

No hay commits todavía

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

Codigo.c
Imagenes/
Nombres.txt
codigo

no hay nada agregado al commit pero hay archivos sin seguimiento presentes (usa "git add" para hacerles seguimiento)
```

COMANDO ADD

Comando Add agrega los archivos al Staging Area, esperando a ser empujados al repositorio.

```
fernando-h@fernando-h:~/Desktop/Git Cripto$ git addCodigo.c
fernando-h@fernando-h:~/Desktop/Git Cripto$ 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:  Codigo.c

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

    Imagenes/
    Nombres.txt
    codigo
```

COMANDO ADD

Si prefieres agregar todos los archivos, usa un punto después del comando Add.

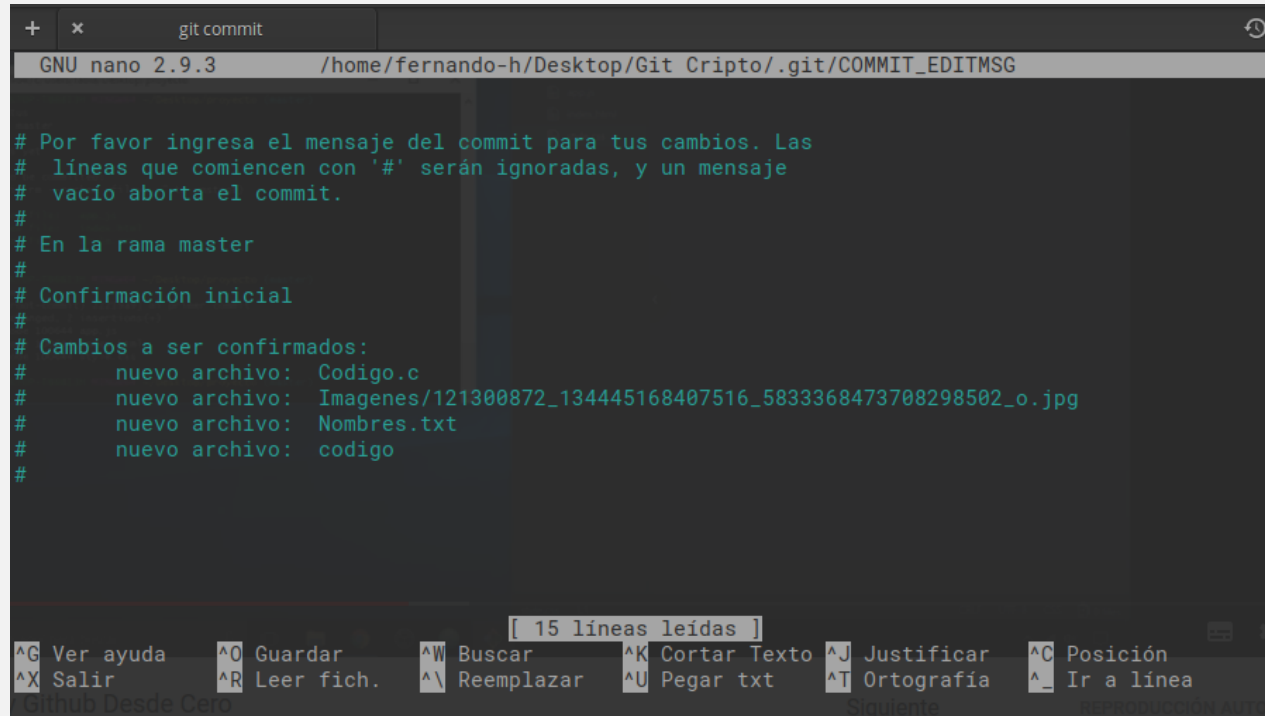
```
fernando-h@fernando-h:~/Desktop/Git Cripto$ git add .
fernando-h@fernando-h:~/Desktop/Git Cripto$ git status
En la rama master

No hay commits todavía

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

nuevo archivo: Codigo.c
nuevo archivo:  Imagenes/121300872_134445168407516_5833368473708298502_o.jpg
nuevo archivo:  Nombres.txt
nuevo archivo:  codigo
```

COMANDO COMMIT

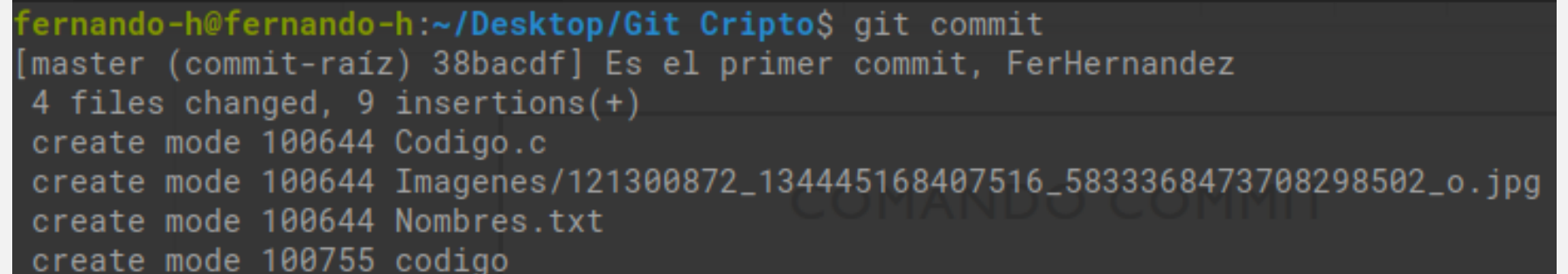


```
GNU nano 2.9.3 /home/fernando-h/Desktop/Git Cripto/.git/COMMIT_EDITMSG

# 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: Codigo.c
#   nuevo archivo: Imagenes/121300872_134445168407516_5833368473708298502_o.jpg
#   nuevo archivo: Nombres.txt
#   nuevo archivo: codigo
#
```

Los archivos ya están en el repositorio.

Se debe agregar un texto, el cual aparecerá en el historial.



```
fernando-h@fernando-h:~/Desktop/Git Cripto$ git commit
[master (commit-raíz) 38bacdf] Es el primer commit, FerHernandez
4 files changed, 9 insertions(+)
create mode 100644 Codigo.c
create mode 100644 Imagenes/121300872_134445168407516_5833368473708298502_o.jpg
create mode 100644 Nombres.txt
create mode 100755 codigo
```

COMANDO COMMIT

Para evitar que se abra un editor de texto, ponemos un `-m` y añadimos el mensaje en el mismo comando.

```
fernando-h@fernando-h:~/Desktop/Git Cripto$ git add Nombres.txt
fernando-h@fernando-h:~/Desktop/Git Cripto$ git commit -m "Segundo cambio"
[master 1f7453a] Segundo cambio
 1 file changed, 1 insertion(+), 1 deletion(-)
fernando-h@fernando-h:~/Desktop/Git Cripto$ git log
commit 1f7453ad836232a0e130200ca1e709a0793799fd (HEAD -> master)
Author: FerHernandez13 <hernandezF@live.com.mx>
Date:   Thu Oct 15 17:15:20 2020 -0500

    Segundo cambio

commit 38bacdf4cf97d80460bf6f664d4bfd945123cee5
Author: FerHernandez13 <hernandezF@live.com.mx>
Date:   Thu Oct 15 16:08:21 2020 -0500

    Es el primer commit, FerHernandez
```


SUBIR PROYECTOS A GITHUB

Github es una herramienta muy importante, ya que se puede ver el código en tiempo real sin necesidad de descargar.

Create a new repository

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

Owner *

 FerHernandez13 ▾

Repository name *

/ Git_Cripto ✓

Great repository names are short and memorable. Need inspiration? How about [jubilant-spoon?](#)

Description (optional)

Es para ejemplificar cosas 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.

FerHernandez13 / **Git_Cripto**

<> Code

! Issues

🔗 Pull requests

▶ Actions

📁 Projects

📖 Wiki

🛡️ Settings

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

or ☐ HTTPS ☐ SSH `https://github.com/FerHernandez13/Git_Cripto.git`

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

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

```
echo "# Git_Cripto" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/FerHernandez13/Git_Cripto.git
git push -u origin main
```

SUBIR PROYECTOS A GITHUB

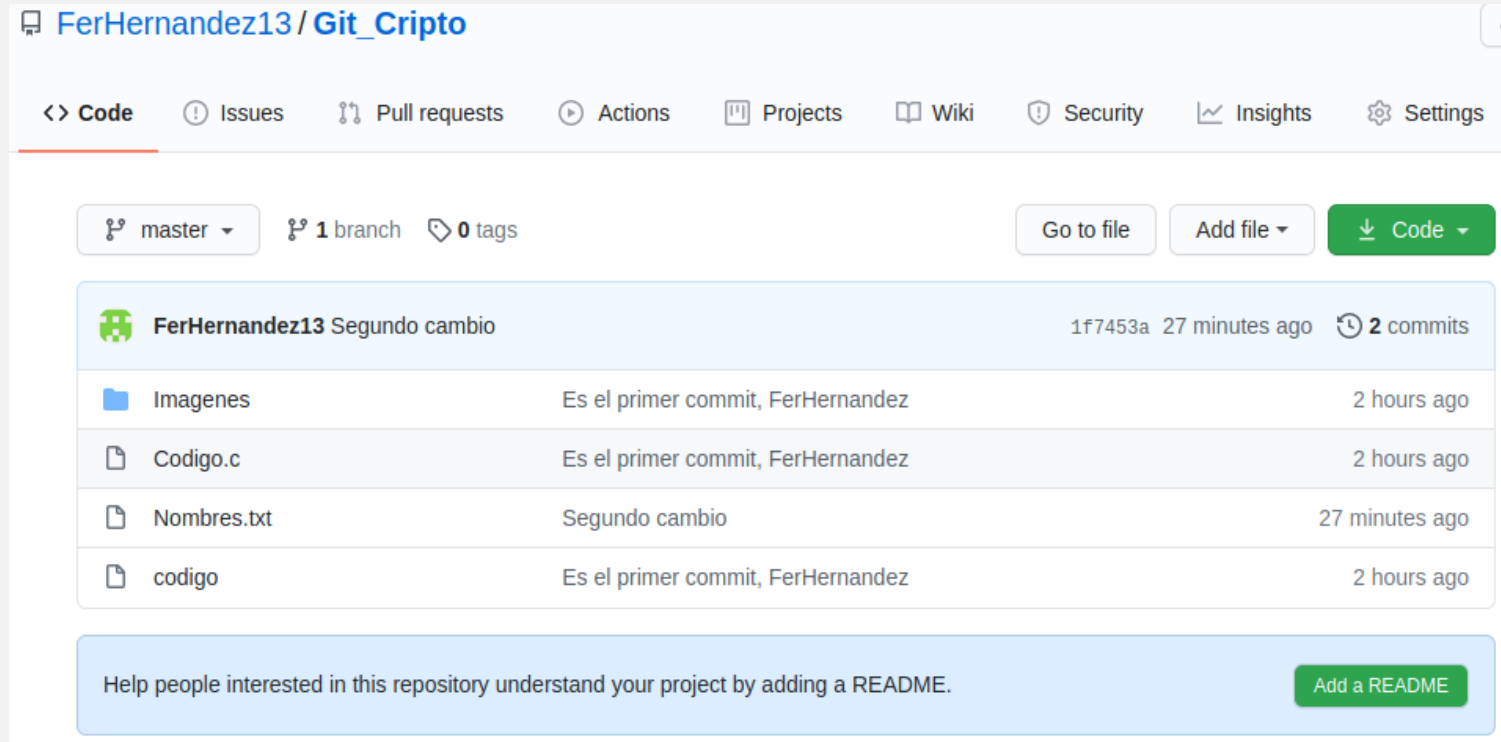
Una vez creado un nuevo repositorio nos dará un link al cual pertenece. Ejecutamos el comando desde la consola:

```
fernando-h@fernando-h:~/Desktop/Git Cripto$ git remote add origin https://github.com/FerHernandez13/Git_Cripto
```

```
fernando-h@fernando-h:~/Desktop/Git Cripto$ git push
Username for 'https://github.com': FerHernandez13
Password for 'https://FerHernandez13@github.com':
Contando objetos: 3, listo.
Delta compression using up to 4 threads.
Comprimiendo objetos: 100% (2/2), listo.
Escribiendo objetos: 100% (3/3), 283 bytes | 283.00 KiB/s, listo.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/FerHernandez13/Git_Cripto
   4be2b5e..089b576  master -> master
```

SUBIR PROYECTOS A GITHUB

En Github ahora está:



The screenshot shows the GitHub interface for a repository named 'Git_Cripto' by user 'FerHernandez13'. The repository is on the 'master' branch, has 1 branch and 0 tags. The commit history shows two commits: 'Segundo cambio' (1f7453a, 27 minutes ago) and 'Es el primer commit, FerHernandez' (2 hours ago). The repository contains four files: 'Imagenes' (folder), 'Codigo.c', 'Nombres.txt', and 'codigo'. A green 'Add a README' button is visible at the bottom.

FerHernandez13 / Git_Cripto

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

master 1 branch 0 tags

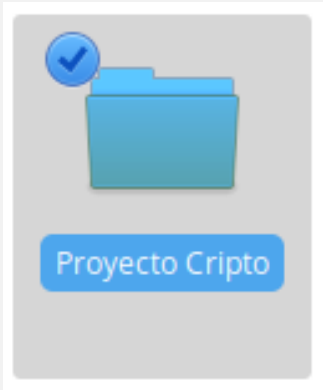
Go to file Add file Code

FerHernandez13 Segundo cambio 1f7453a 27 minutes ago 2 commits

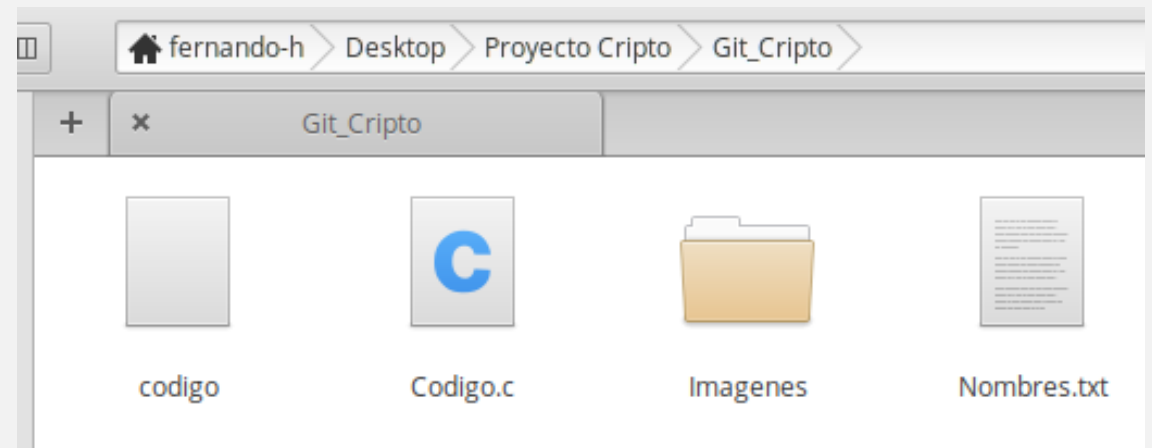
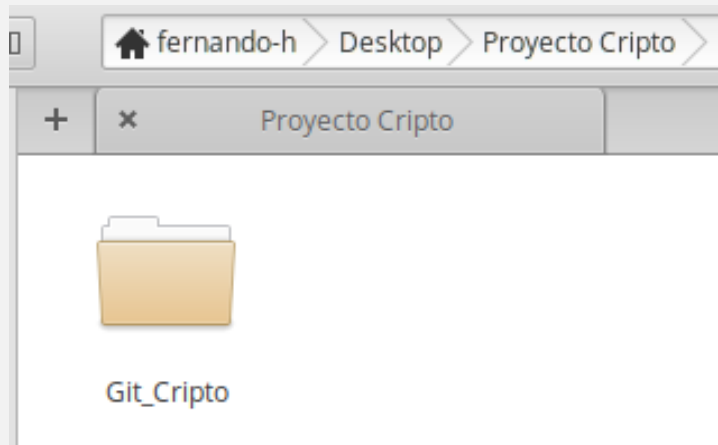
Imagenes	Es el primer commit, FerHernandez	2 hours ago
Codigo.c	Es el primer commit, FerHernandez	2 hours ago
Nombres.txt	Segundo cambio	27 minutes ago
codigo	Es el primer commit, FerHernandez	2 hours ago

Help people interested in this repository understand your project by adding a README. [Add a README](#)

CLONAR PROYECTOS DE GITHUB



```
fernando-h@fernando-h:~/Desktop/Proyecto Cripto$ git clone https://github.com/FerHernandez13/Git_Cripto
Clonando en 'Git_Cripto'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 10 (delta 1), reused 10 (delta 1), pack-reused 0
Desempaquetando objetos: 100% (10/10), listo.
```



ACTUALIZAR LOS POSIBLES CAMBIOS HECHOS POR OTROS USUARIOS.

Si alguien realizó cambios mientras nosotros también los hacemos y hacen un push, antes que nosotros; tenemos que realizar un pull para descargar los archivos con las actualizaciones que los demás realizaron.

```
fernando-h@fernando-h:~/Desktop/Git Cripto$ git pull  
Ya está actualizado.
```

ACTIVIDAD:

- Instalar Git.
- Clonar el repositorio del link.
- Añadir un printf en el código con su nombre.
- Añadir una imagen en la carpeta Imágenes.
- Realizar los pasos para para subir al repositorio los cambios que realizaron.
- Hacer un pull, para bajar los cambios hechos (si es necesario).