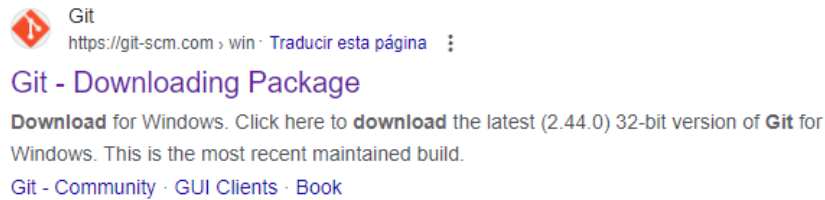


Guía para creación de cuenta de GitHub:

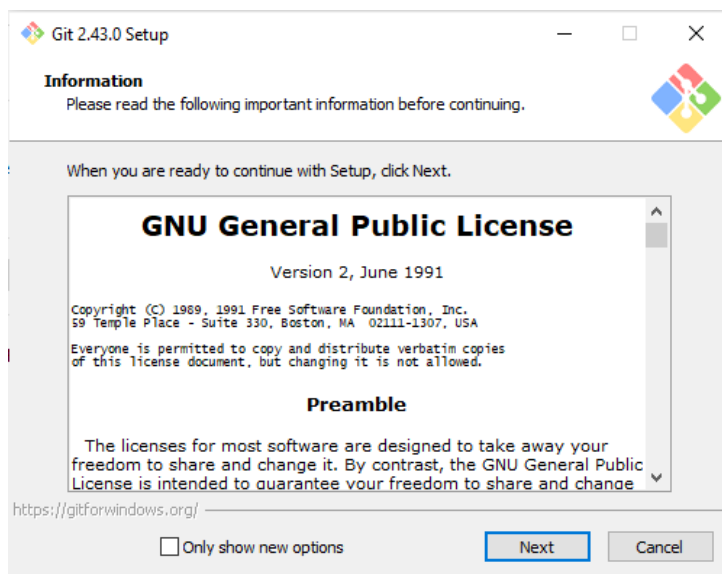
1-En el buscador realizar la búsqueda de "Git Download" y acceder al siguiente enlace:



2-



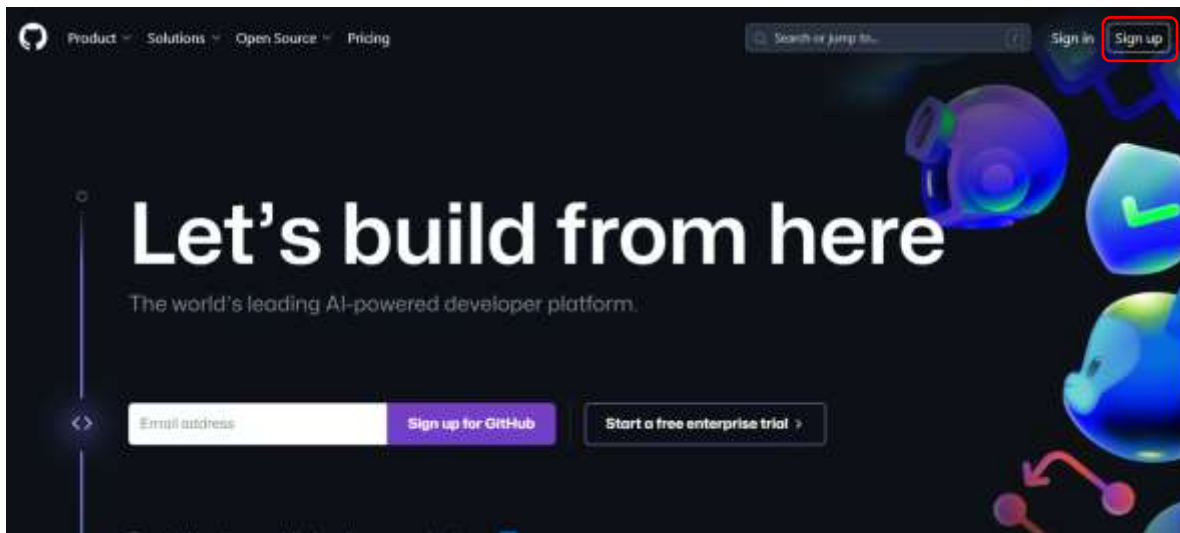
Al hacer click comenzará una descarga, al terminar se debe iniciar el programa y realizar la instalación del mismo:



Luego acceder al siguiente enlace:

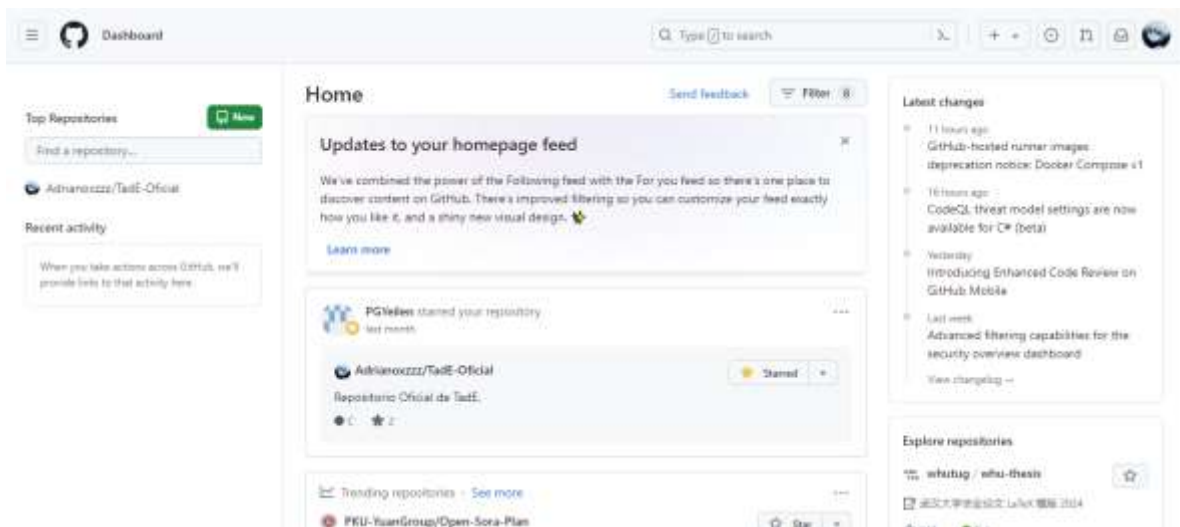
<https://github.com>

Aparecerá esta ventana donde deberemos loguearnos:



La plataforma ofrece un modo interactivo bastante fácil para familiarizarse en la creación de la cuenta.

Luego de creada la cuenta aparecerá la ventana principal del GitHub:



Nos dirigiremos al símbolo de + que se muestra en la esquina superior derecha. Luego de esto iremos a New Repository para la creación del repositorio.

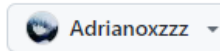
4-

Create a new repository

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

Required fields are marked with an asterisk ().*

Owner *



Repository name *

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

Description (optional)

En los primeros dos espacios se corresponde al nombre del repositorio y su descripción, la cual es opcional.

☒ 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:

☐ Add a README file

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

Add .gitignore

.gitignore template: None ▾

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

Choose a license

License: None ▾

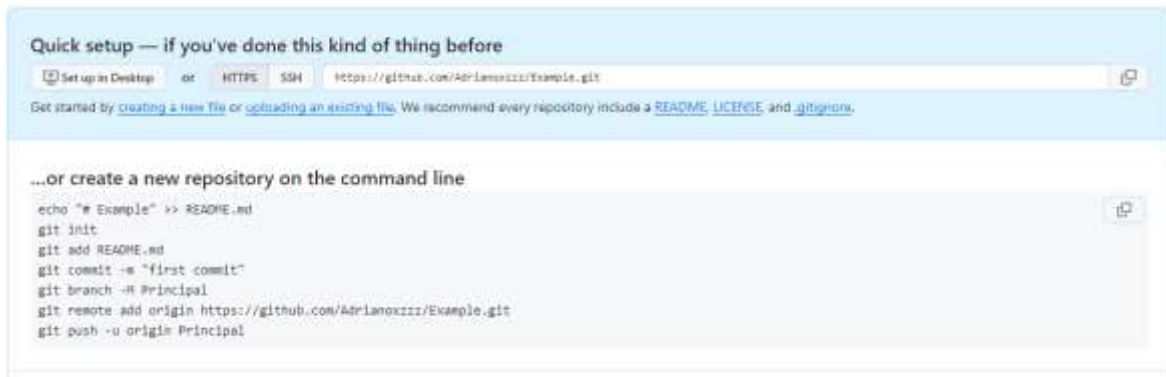
A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

You are creating a public repository in your personal account.

Luego las demás opciones de mayor importancia. El repositorio TIENE que ser de tipo público, para que pueda ser accedido por los profesores.

El archivo readme es opcional, aunque recomendado.

La ventana de Add. gitignore se dejará como por defecto mientras que en la licencia se puede escoger cualquiera que se desee, no tendrá mayor importancia, aunque se recomienda estudio acerca de los privilegios que ofrecen ciertas licencias en específico.



Nos abrirá la siguiente ventana, la cual corresponde para la creación de un destino remoto en nuestro ordenador, el cual vincularemos con el repositorio de GitHub.

Para ello:

Crearemos una carpeta en cualquier lugar de nuestro ordenador, donde realizaremos los updates pertinentes.



Luego estando dentro de la ruta de la carpeta y haciendo click sobre esta escribiremos cmd como se muestra en la siguiente figura:



Esto nos abrirá la consola de comandos con la dirección de nuestra carpeta ya incorporada:

```
Microsoft Windows [Versión 10.0.19045.3448]  
(c) Microsoft Corporation. Todos los derechos reservados.  
  
F:\Example>_
```

En ella escribiremos el siguiente comando:

```
F:\Example>git -v  
git version 2.43.0.windows.1
```

Git -v nos permite ver la correcta instalación del git, si nos aparece "git no se reconoce como un comando externo ni interno" se debe revisar la instalación realizada.

Luego pasaremos al comando

```
F:\Example>git init  
Initialized empty Git repository in F:/Example/.git/
```

Git init nos permite inicializar un nuevo repositorio dentro de esta carpeta.

Luego realizaremos el comando git add . (Importante el punto para añadir todos los archivos en la carpeta). Para luego pasar a los siguientes comandos:

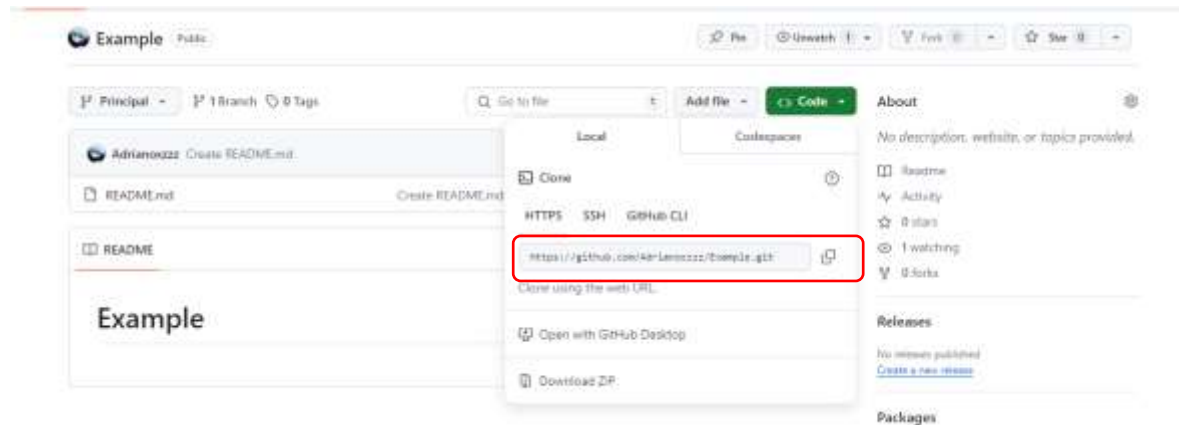
```
F:\Example>git status  
On branch Principal  
  
No commits yet  
  
Changes to be committed:  
  (use "git rm --cached <file>..." to unstage)  
    new file:   Example.txt  
  
F:\Example>git commit -m "Testing"  
[Principal (root-commit) fbdfe42] Testing  
1 file changed, 1 insertion(+)  
create mode 100644 Example.txt
```

Como observamos al escribir el comando git status nos debe salir como aparece en verde los files que se han actualizado desde la última subida (En este caso solo el txt que hemos creado)

El próximo comando será: git commit -m "Testing". Obsérvese aquí que lo que esta entre comillas es simple texto, puede aparecer cualquier cosa, ya que lo que estamos haciendo aquí es actualizar todo lo que tengamos en esta carpeta para

que sea subido al repositorio, luego lo que aparece entre comillas será el nombre con el que aparecerá dentro del repositorio.

El próximo paso que es de vital importancia será vincular los repositorios, para ello tomaremos el link que aparece en donde se indica en la siguiente figura y escribiremos el comando:



`git remote add origin direccion copiada`

Para luego realizar el comando

`Git push origin Principal`. En caso de que aparezca el siguiente error lo que se debe hacer es escribir el comando:

```
F:\Example>git push -u origin Principal
To https://github.com/Adrianoxzzz/Example.git
! [rejected]        Principal -> Principal (fetch first)
error: failed to push some refs to 'https://github.com/Adrianoxzzz/Example.git'
hint: Updates were rejected because the remote contains work that you do not
hint: have locally. This is usually caused by another repository pushing to
hint: the same ref. If you want to integrate the remote changes, use
hint: 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
```

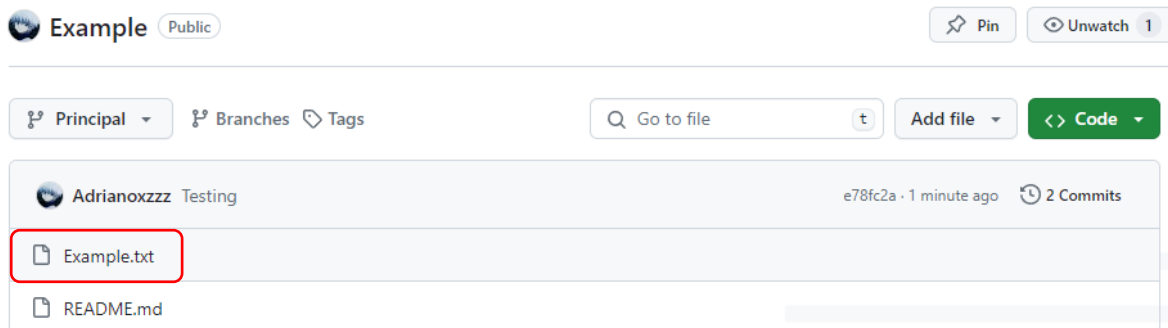
`Git pull --rebase origin Principal`

```
F:\Example>git pull --rebase origin Principal
From https://github.com/Adrianoxzzz/Example
* branch            Principal -> FETCH_HEAD
Successfully rebased and updated refs/heads/Principal.
```

Y de esta forma se podrá realizar el commit

```
F:\Example>git push origin Principal
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 283 bytes | 283.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Adrianoxzzz/Example.git
ad91a43..e78fc2a Principal -> Principal
```

Si vamos al repositorio veremos que el commit se ha realizado satisfactoriamente



A partir de aquí para realizar cualquier commit solo seguir los mismos pasos hasta abrir el cmd (Ojo siempre en la misma carpeta donde se creó el repositorio remoto) y a partir de aquí ir escribiendo cada uno de los siguientes comandos:

- Git add .
- Git status (si se desea ver si se añadió correctamente la información)
- Git commit -m "(nombre del comentario sin los paréntesis)"
- Git push origin Principal

Nota Importante: Deben fijarse cuando se cree el repositorio, ya que es muy probable que el nombre de la rama principal en su caso no sea Principal, sino Master u Main, por lo tanto, todos los comandos en los que ha aparecido Principal deberán reemplazar esta palabra por la del Branch principal de su repositorio.