

Uso básico de git y GitHub

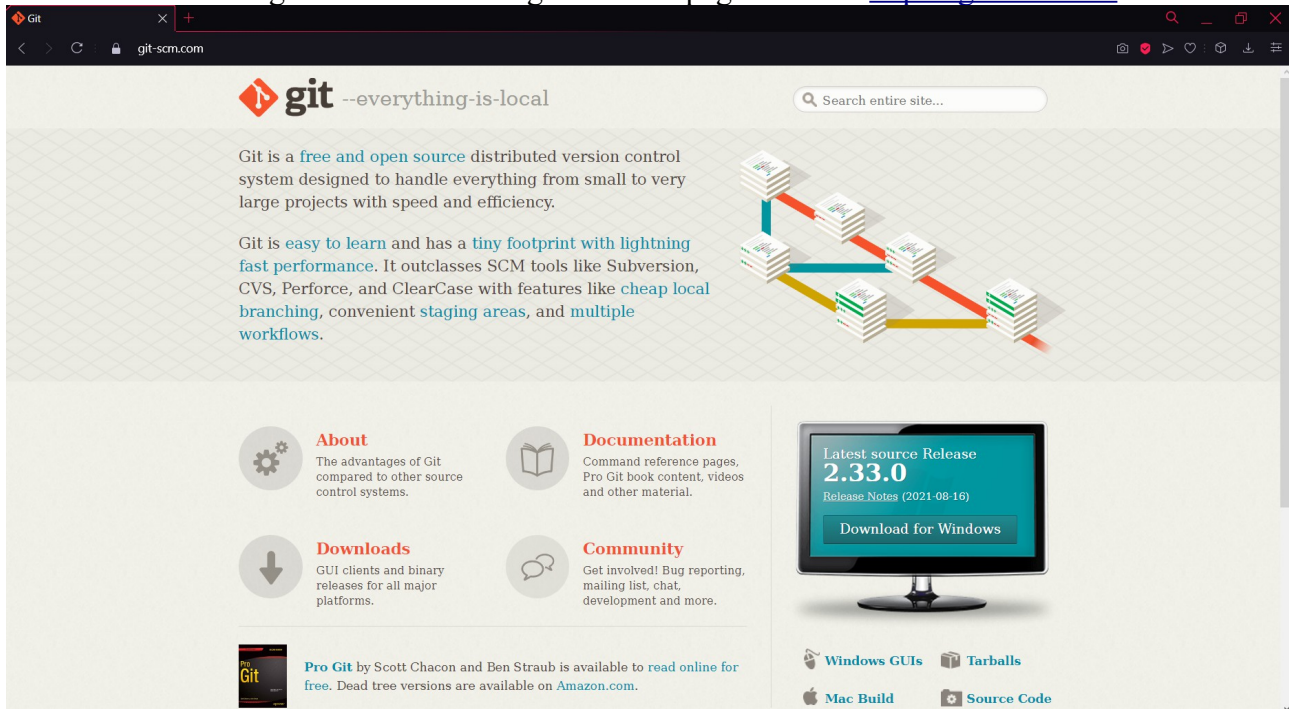
Iván Gasent Zarza

Indice

Indice	1
Instalación git	2
Uso básico de GitHub	3
Creación de una cuenta en Github	3
Creación de un repositorio	4
Actualizar el repositorio remoto	6
Actualizar el repositorio local	7
Clonar el repositorio remoto	7
Configuración README.md	8
Bibliografía	10

Instalación de git

Descargue el instalador de git desde su página oficial <https://git-scm.com>



Ejecute el instalador de git y elija las opciones que desee.

Compruebe que git esté bien instalado ejecutando el siguiente comando en PowerShell:

```
git --version
```

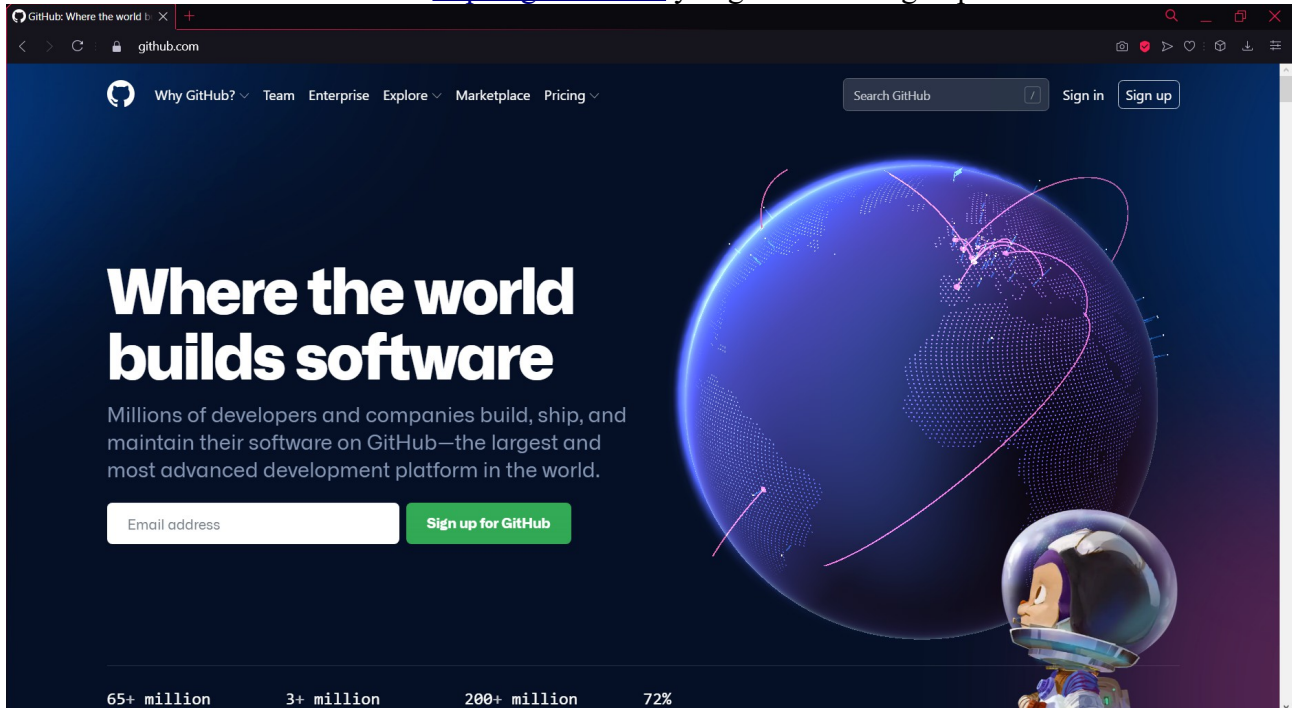
Si git está bien instalado deberá aparecer esta línea por la consola. El número de versión puede ser distinto.

```
PS C:\> git --version
git version 2.33.0.windows.2
```

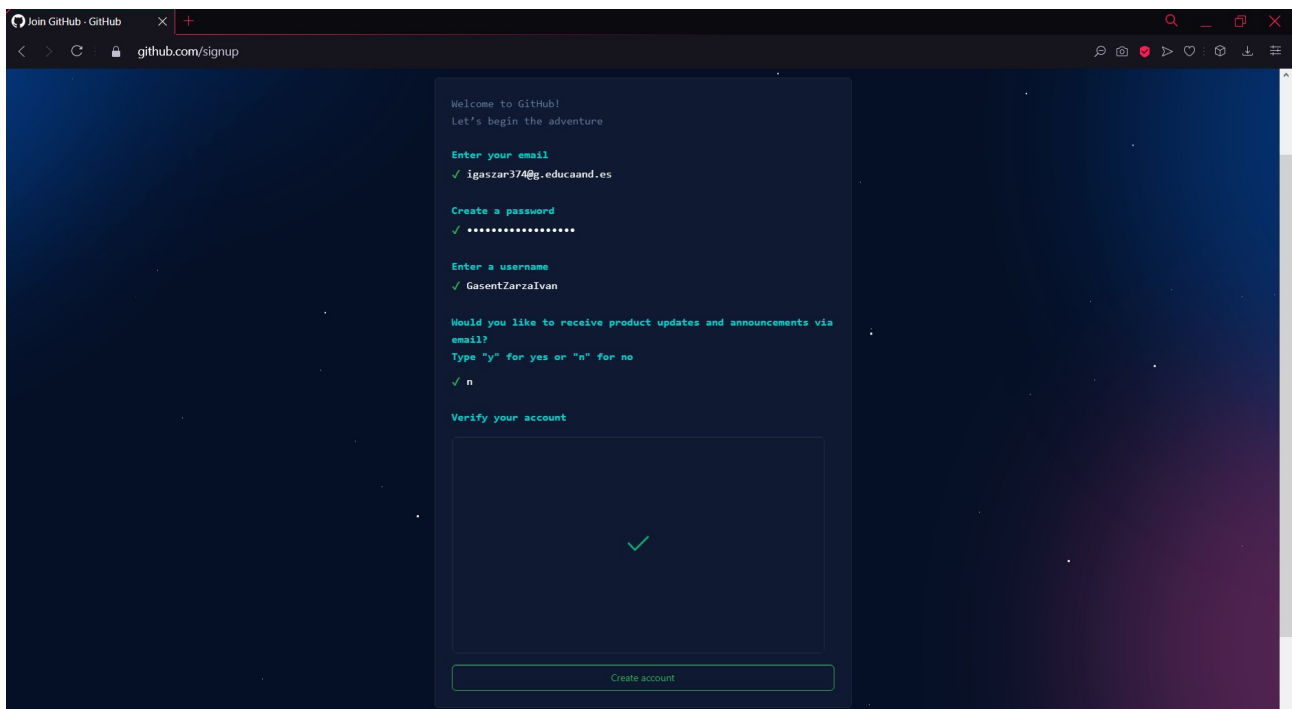
Uso básico de GitHub

Creación de una cuenta en GitHub

Entre en <https://github.com> y haga click en Sign up.



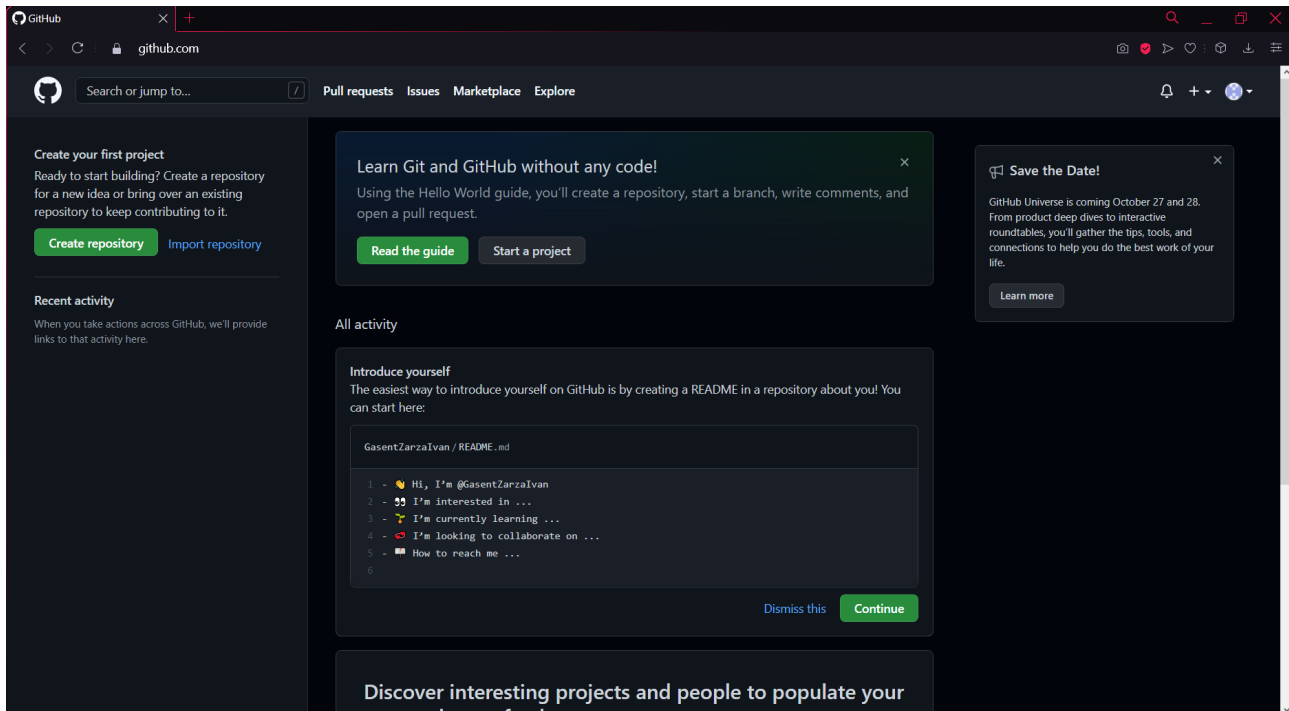
Rellene la información de su cuenta y verifique que no es un robot.



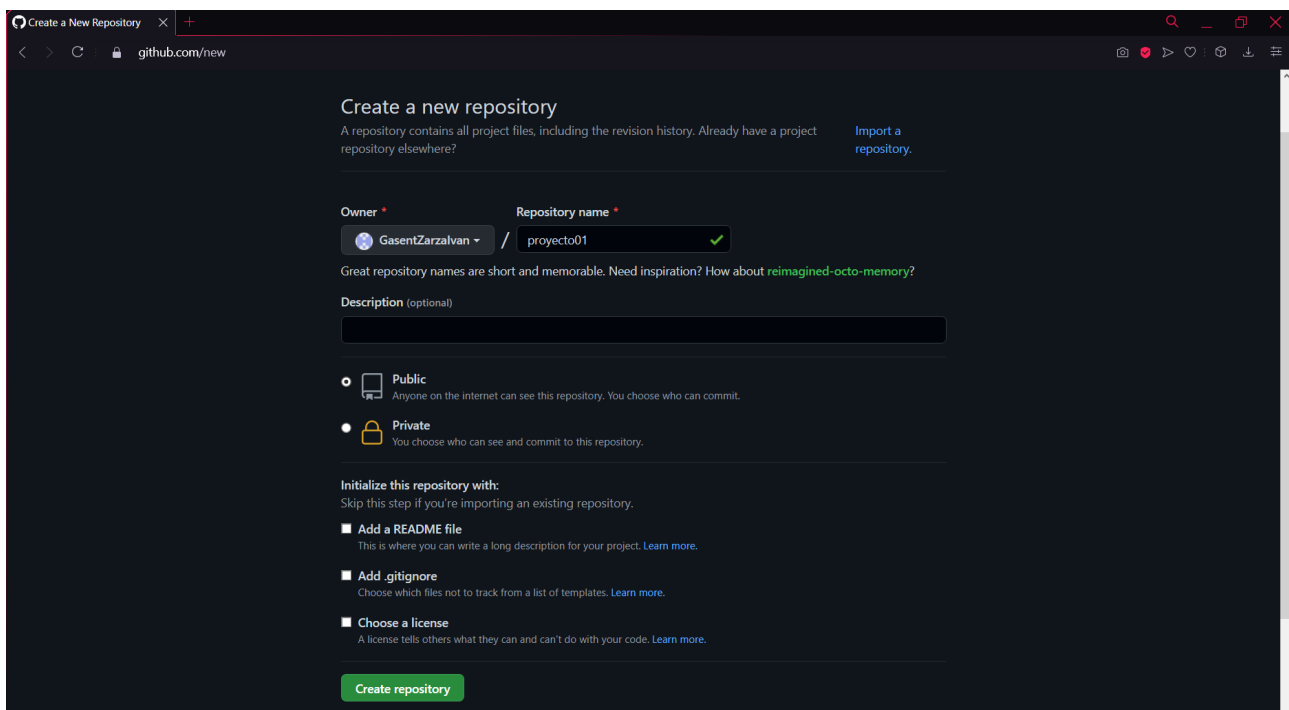
Introduzca el código que ha sido enviado a su correo para verificar su cuenta.

Creación de un repositorio

Haga click en Create repository



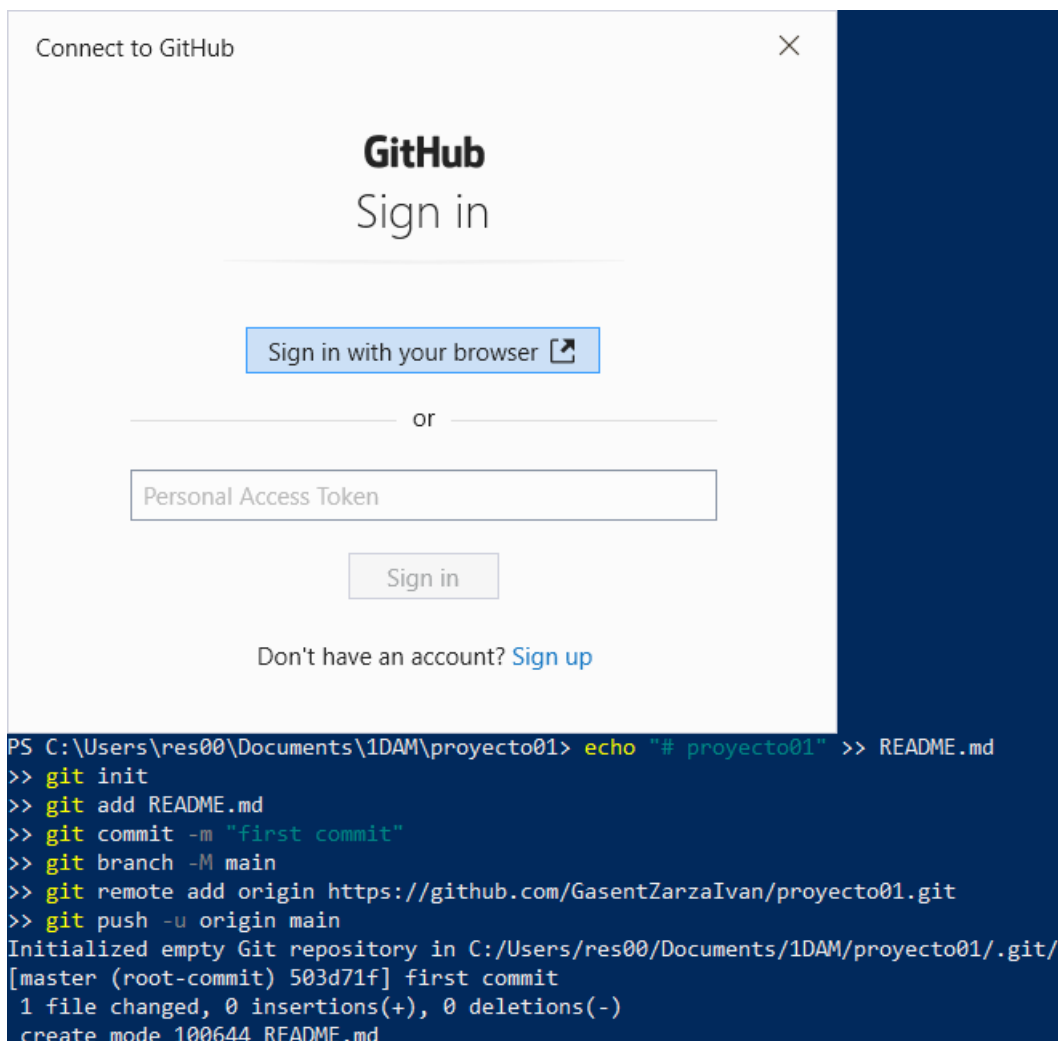
Escriba un nombre para el repositorio y haga click en Create repository.



Elija un directorio local para convertir en su repositorio local y acceda a el mediante PowerShell.
Introduzca los siguientes comandos:

```
echo "# repositorio" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin [URL DE TU REPOSITORIO]
git push -u origin main
```

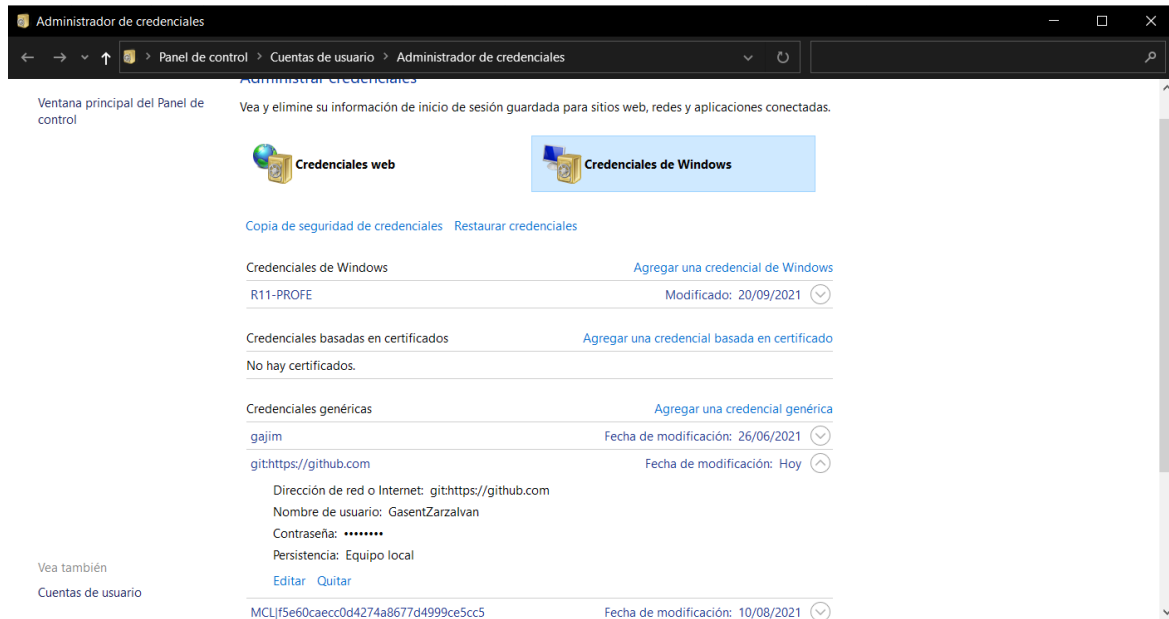
Al ejecutar los comandos aparecerá una ventana para acceder y guardar sus credenciales de GitHub en su sistema.



Si es la primera vez que accede a GitHub mediante git tendrá que usar los siguientes comandos:

```
git config --global user.email "you@example.com"
git config --global user.name "Your Name"
```

Si quiere borrar las credenciales de su sistema tendrá que acceder al administrador de credenciales.



Actualizar el repositorio remoto

Para actualizar el repositorio remoto con los cambios realizados en su repositorio local, use los siguientes comandos:

```
git add .
git commit -m "[TEXTO BREVE DE LOS CAMBIOS HECHOS]"
git push -u origin main
```

```
PS C:\Users\res00\Documents\1DAM\proyecto01> git add .
>> git commit -m "Directorio src y doc creados. Código de ejemplo agregado."
>> git push -u origin main
[main 7e2844b] Directorio src y doc creados. Código de ejemplo agregado.
 1 file changed, 1 insertion(+)
 create mode 100644 src/ejemplo01.txt
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 398 bytes | 398.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/GasentZarzaIvan/proyecto01.git
 503d71f..7e2844b  main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.
PS C:\Users\res00\Documents\1DAM\proyecto01>
```

Con el siguiente comando se pueden ver si hay cambios seleccionados que aun no se hayan actualizado:

git status

```
PS C:\Users\res00\Documents\1DAM\proyecto01> git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    doc/documentacion01.pdf

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

Actualizar el repositorio local

Para actualizar su repositorio local con los cambios realizados en el repositorio remoto, puede usar varios comandos:

`git fetch` (descarga archivos nuevos sin modificar los archivos en el repositorio local)
o
`git pull` (descarga y modifica el repositorio local con los archivos nuevos)

```
PS C:\Users\res00\Documents\1DAM\proyecto01> git pull
remote: Enumerating objects: 7, done.
remote: Counting objects: 100% (7/7), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (4/4), 785 bytes | 71.00 KiB/s, done.
From https://github.com/GasentZarzaIvan/proyecto01
  6f8df9d..129600c  main      -> origin/main
Updating 6f8df9d..129600c
Fast-forward
 src/ejemplo01.txt | 3 ++-
 1 file changed, 2 insertions(+), 1 deletion(-)
PS C:\Users\res00\Documents\1DAM\proyecto01>
```

Clonar el repositorio remoto

Si quiere establecer un repositorio local de un repositorio remoto, puede hacer lo siguiente:

`cd [RUTA DEL DIRECTORIO DONDE SE COPIARA EL REPOSITORIO REMOTO]`
`git clone [URL DEL DIRECTORIO REMOTO]`

o

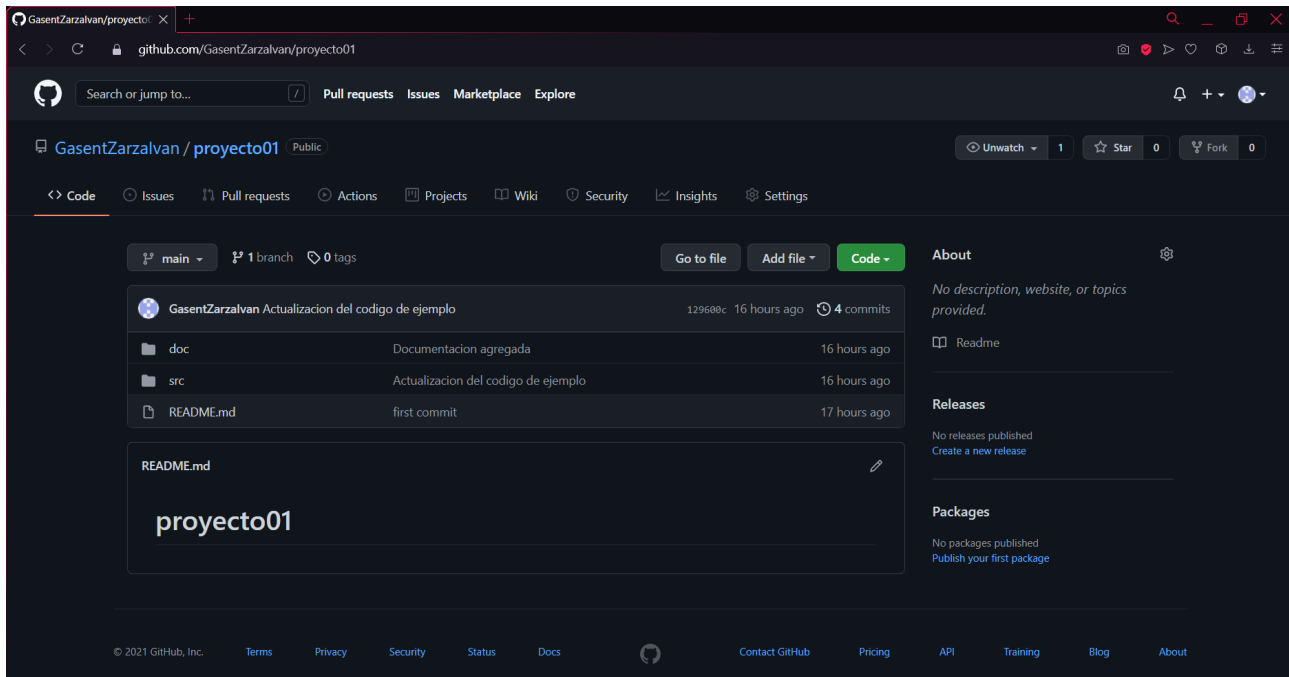
`git clone [URL DEL REPOSITORIO REMOTO] [DIRECCION DONDE ESTABLECER EL REPOSITORIO LOCAL]`

```
PS C:\Users\res00\Documents\1DAM\proyecto01> cd "C:\Users\res00\Documents\1DAM\proyecto01 donde se usara clone"
PS C:\Users\res00\Documents\1DAM\proyecto01 donde se usara clone> git clone https://github.com/GasentZarzaIvan/proyecto01
Cloning into 'proyecto01'...
remote: Enumerating objects: 15, done.
remote: Counting objects: 100% (15/15), done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 15 (delta 1), reused 10 (delta 0), pack-reused 0R
Receiving objects: 100% (15/15), 852.56 KiB | 479.00 KiB/s, done.
Resolving deltas: 100% (1/1), done.
PS C:\Users\res00\Documents\1DAM\proyecto01 donde se usara clone>
```

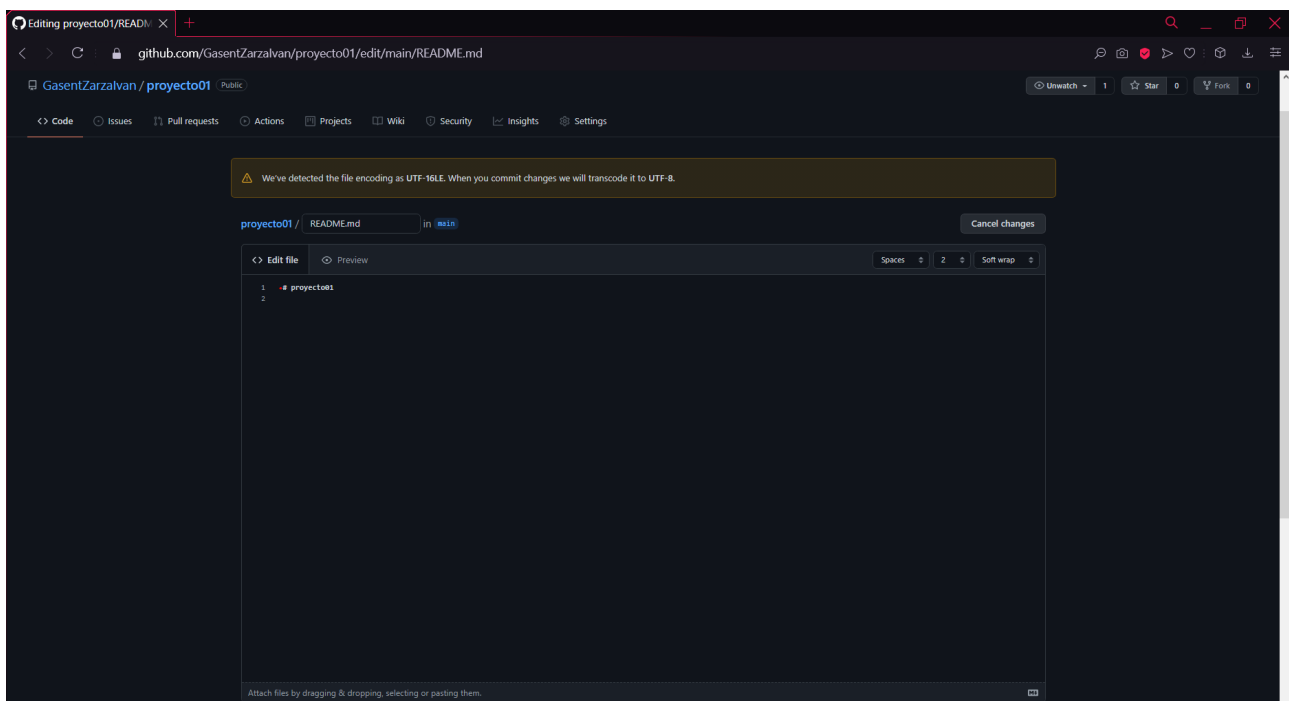
El comando clone creará una carpeta con una copia del repositorio remoto en el directorio elegido, esa carpeta será creada será el repositorio local.

Configuración README.md

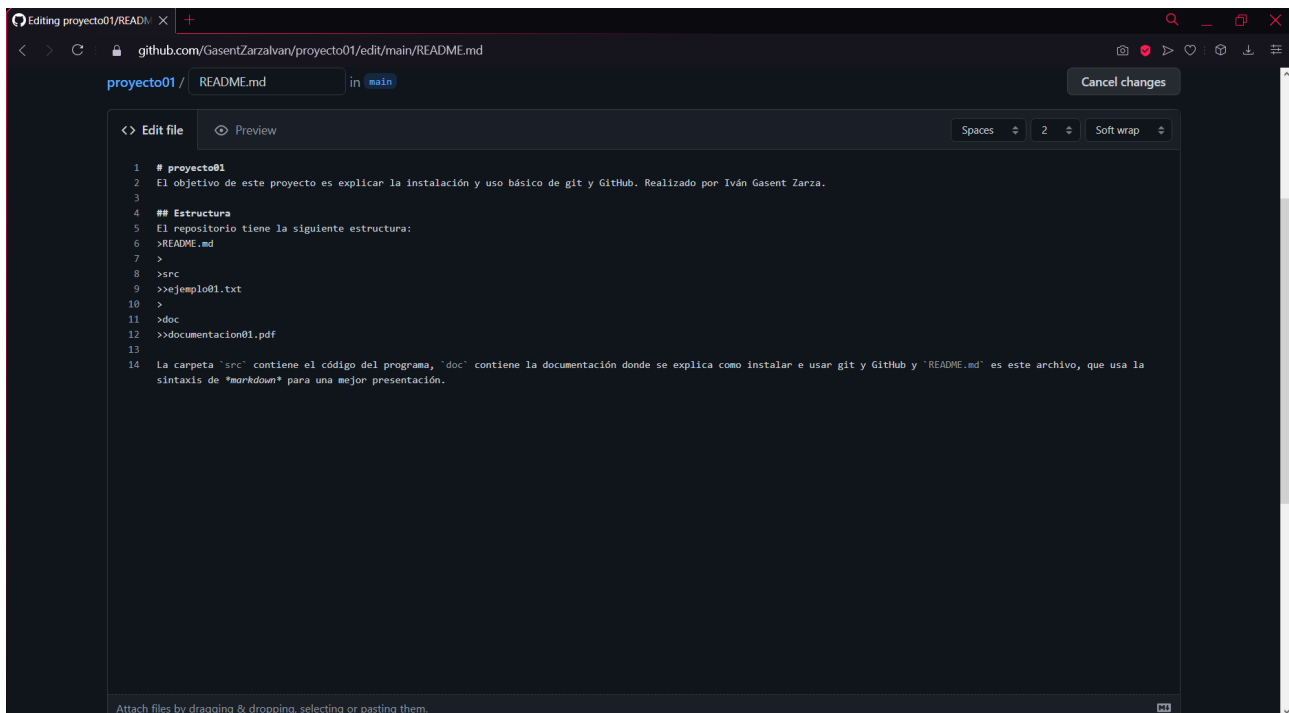
README.md es un archivo Markdown, por lo que podemos usar marcas para personalizar la presentación. GitHub mostrará la información de este archivo en nuestro repositorio.



Podemos editar el archivo README.md con cualquier editor de texto o haciendo click en el símbolo de lápiz en la esquina superior derecha del README.md en el repositorio.



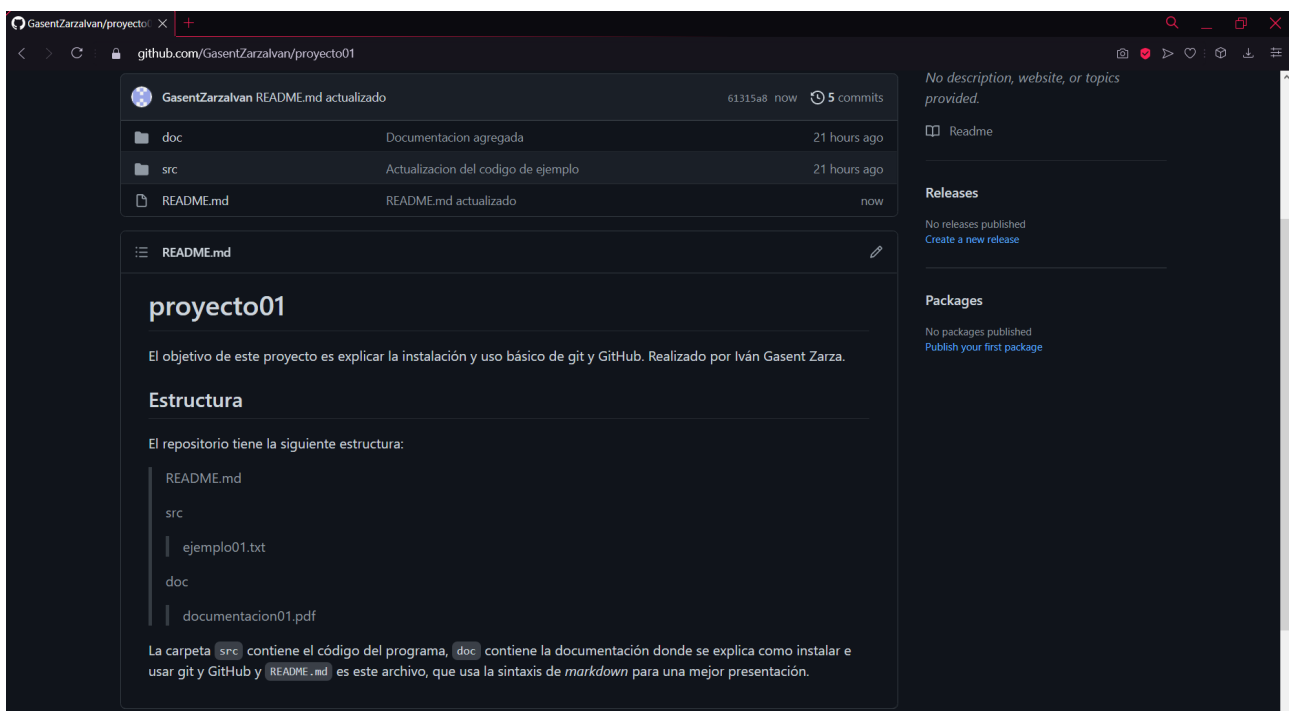
El archivo con sintaxis Markdown se verá de la siguiente forma desde un editor de texto:



The screenshot shows a web-based code editor for a file named 'README.md' in a repository 'proyecto01'. The editor has two tabs: 'Edit file' (active) and 'Preview'. The code is written in Markdown and includes a title, a description, a section header 'Estructura', a list of files, and a paragraph explaining the directory structure. The code is as follows:

```
1 # proyecto01
2 El objetivo de este proyecto es explicar la instalación y uso básico de git y GitHub. Realizado por Iván Gasent Zarza.
3
4 ## Estructura
5 El repositorio tiene la siguiente estructura:
6 >README.md
7 >
8 >src
9 >>ejemplo01.txt
10 >
11 >doc
12 >>documentacion01.pdf
13
14 La carpeta 'src' contiene el código del programa, 'doc' contiene la documentación donde se explica como instalar e usar git y GitHub y 'README.md' es este archivo, que usa la
sintaxis de "markdown" para una mejor presentación.
```

En GitHub se mostrará de la siguiente manera:



La versión de Markdown usada en GitHub tiene modificaciones para un mejor uso en la página, para conocer estos cambios y como usarlos puede visitar la guia oficial proporcionada por GitHub.

<https://guides.github.com/features/mastering-markdown/>

Bibliografia

<https://git-scm.com>

<https://github.com>

<https://www.atlassian.com/es/git/tutorials/setting-up-a-repository>

<https://www.markdownguide.org/basic-syntax/>