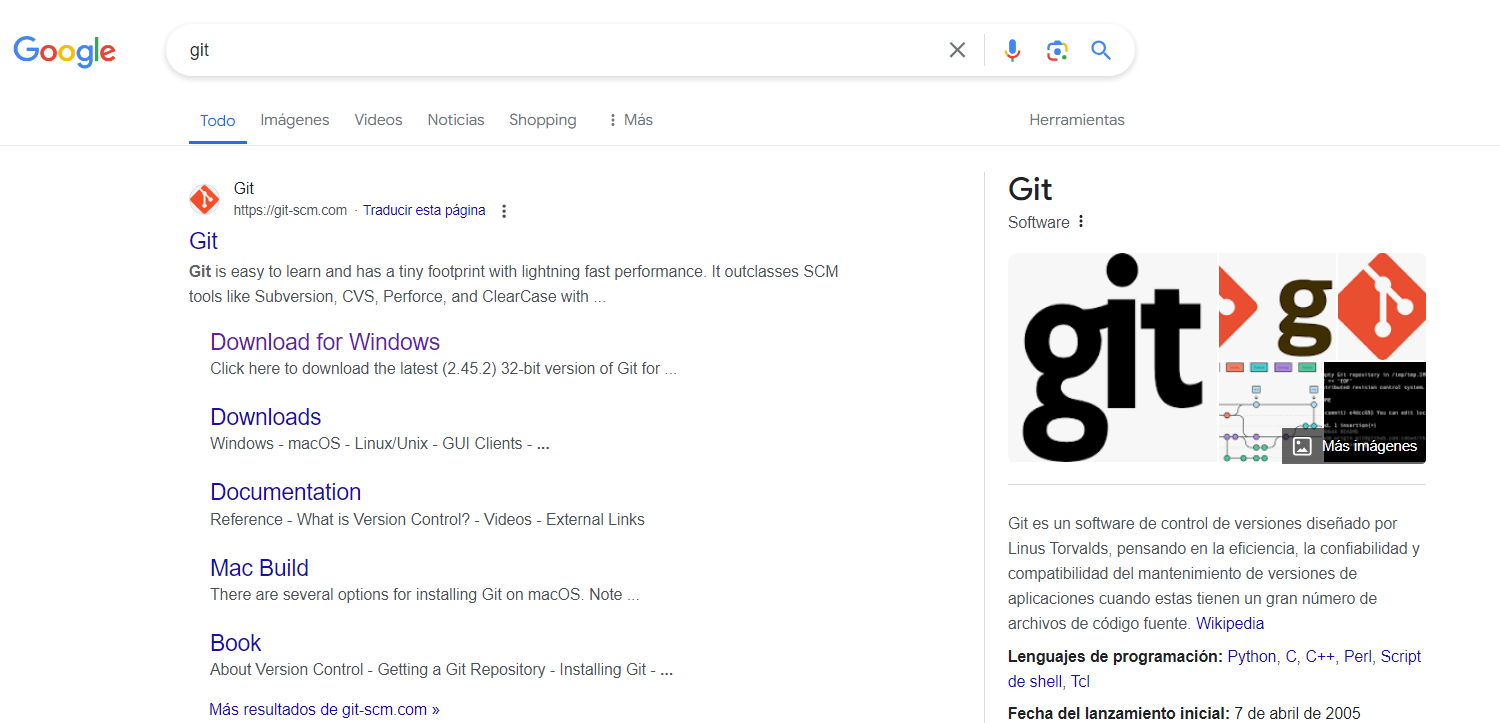
Git y GitHub

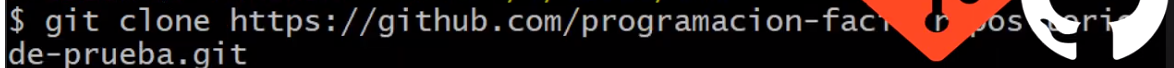
Comados más usados e git empleados para las carpetas o proyectos de la aplicación de GitHub

**NOTA:** Para poder usar GitHub te debes registrar en la página y obtener tu usuario y password

**NOTA**: Para poder usar git y el powersell que es la consola debe instalarse en la computadora y configurarse. (si no lo configuras no funciona)



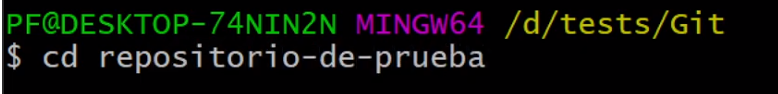
* Clonar un repositorio
* git clon URL



Para poder actualizar los cambios que le han hecho al repositorio clonado se usa:

**Nota**: checar que estamos dentro de la carpeta que creo github

* cd carpeta\_del\_proyecto

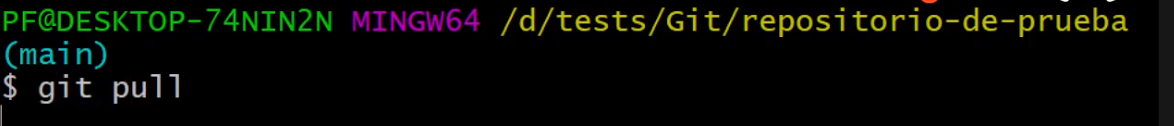


* git pull origin main



O con git pull (bajo los cambios hechos por todos)

* git pull



* Para crear y trabajar con un repositorio local y después subirlo al GitHub

Al inicializar el git, solo funciona en la carpeta y subcarpetas donde ejecutas esta init git, no en carpetas superiores u otras, porque se supone que esa es la carpeta de tu proyecto

Se inicializa git con el siguiente comando (dentro de la carpeta que contiene tu proyecto)

* git init

con gid add agregas la información de los cambios que has realizado, git add nombr\_de\_archivo solo agrega los cambios del Archivo nombre\_de\_archivo a los cambios, con git add . (punto) agrega todos los cambios realizados.

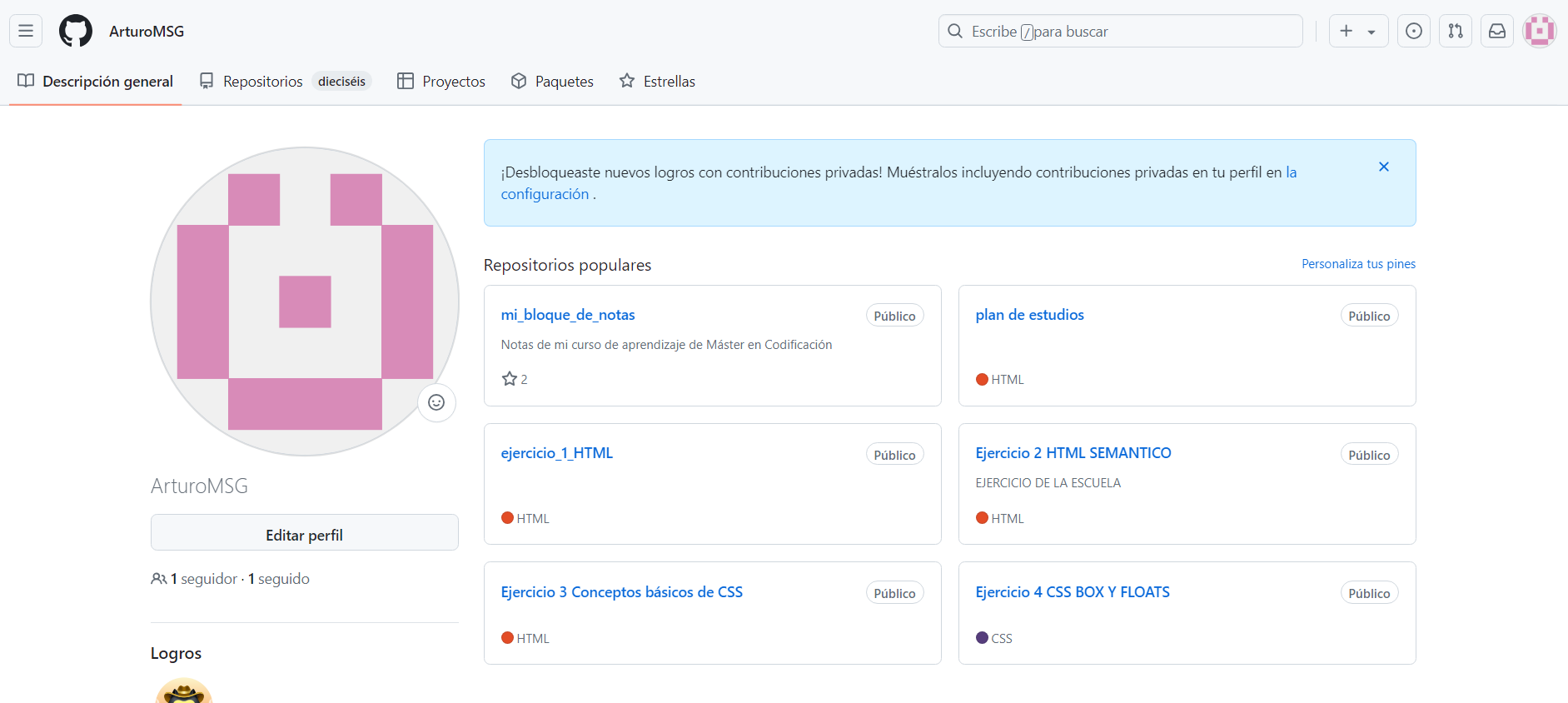
Para poder registrar los cambios en tu proyecto con gi tcommit –m, debes haber agregado el archivo con git add, los registros de los cambios llevan un comentario de que se hizo el comentario que desees.

* git commit -m "comentario\_sobre\_los\_cambios\_realizados"

con esto ya llevas un control sobre los cambios locales que vas realizando a tu proyecto.

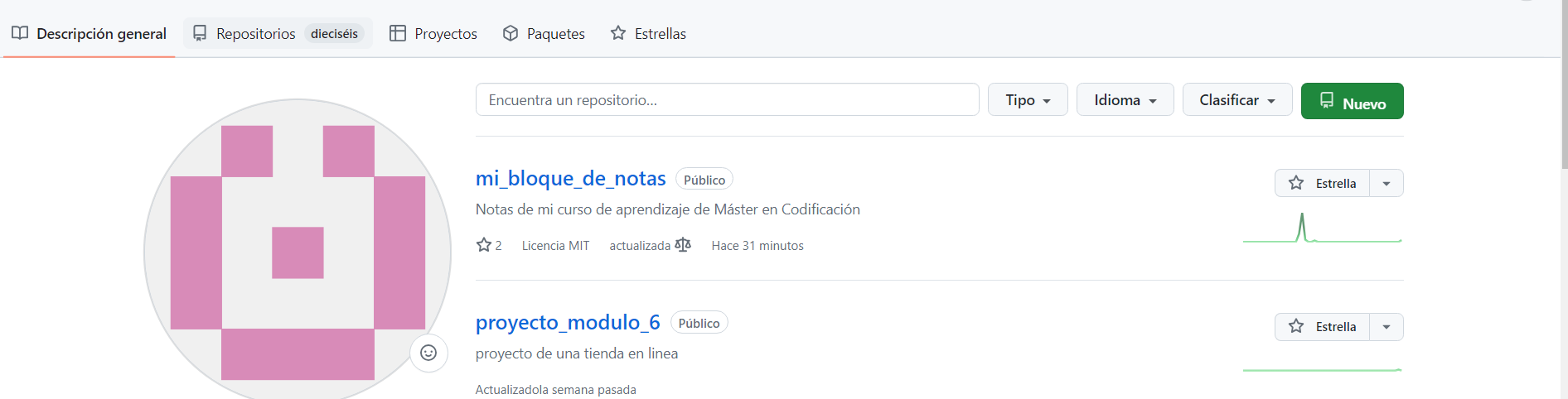
* Para subir tu proyecto a githu, se requiere que hayas hecho los pasos anteriores de git init ,git add y git commit se siguen los siguientes pasos,

Primero entras a tu sesión.



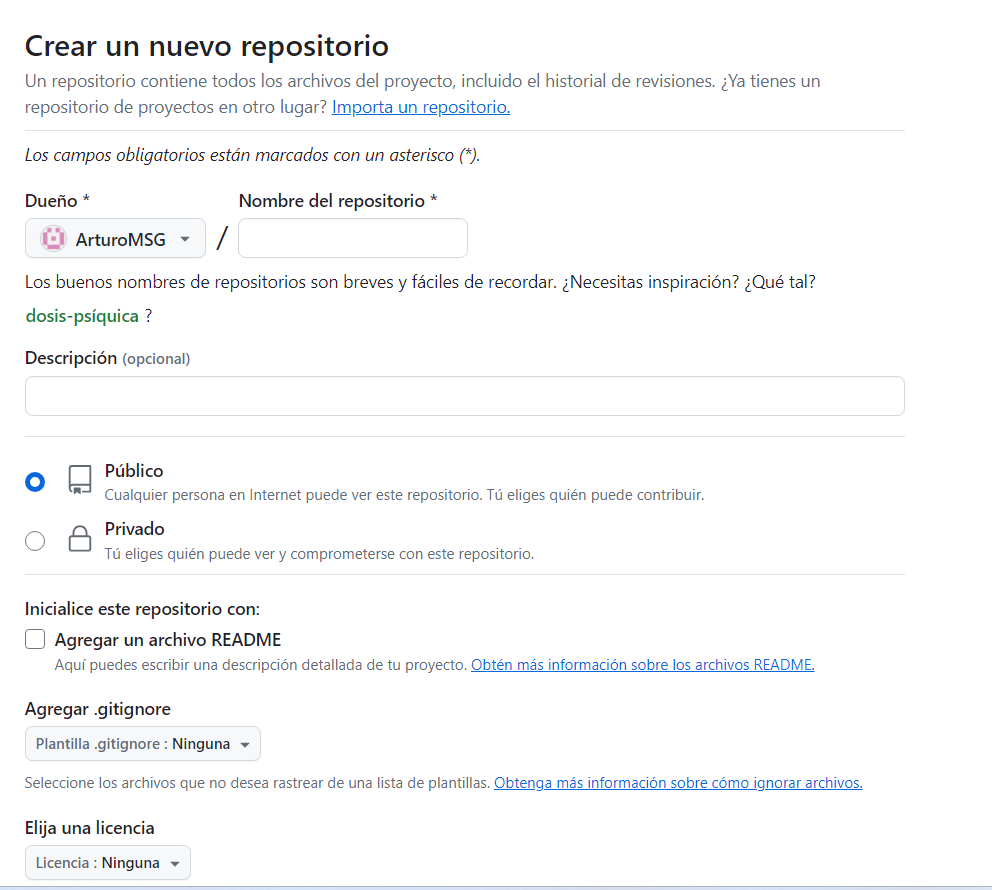
creas el repositorio de tu proyecto en GitHub.

Das click en la pestaño de Repositorios y te aparece la posibilidad de crear un nuevo repositorio y le das click a nuevo (botón verde)

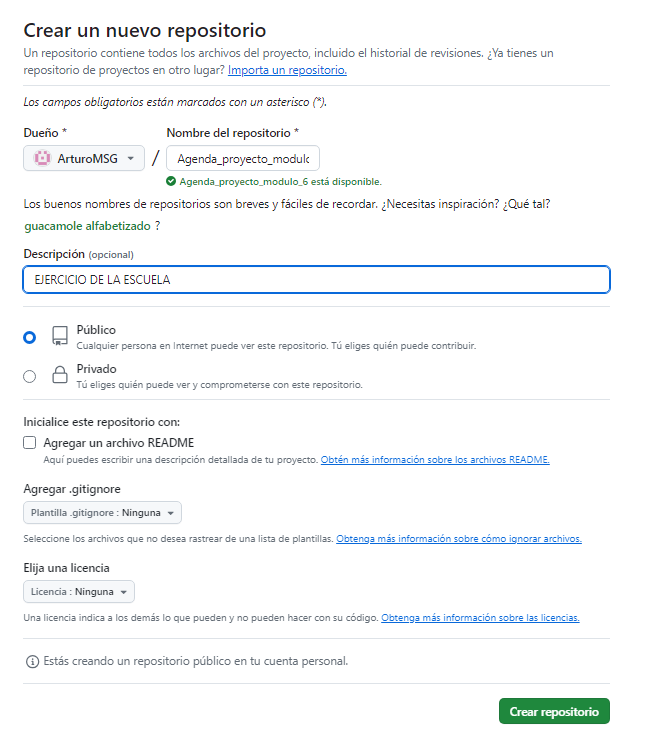


Aparece la siguiente pantalla la cual debes llenar de la siguiente forma

* + Nombre del repositorio: el nombre que le quieras dar a tu proyecto, todas las palabras deben estar unidas por ion bajo ( \_ ) o ion medio ( - ), para evitar que genere errores.
  + Descripción: aquí describes tu proyecto
  + Agregar archivo rearme, no lo señales, porque vas a crear el archivo de forma local-
  + Agregar .gitnore: ninguna (esto es para archivos que no se suben a GitHub de tu proyecto, pero esto se generara de forma local)
  + Público o Privado☹si lo haces publico todos pueden verlo al igual que tu código, si es privado no se puede ver tu código solo el proyecto.
  + Licencia:ninguna.

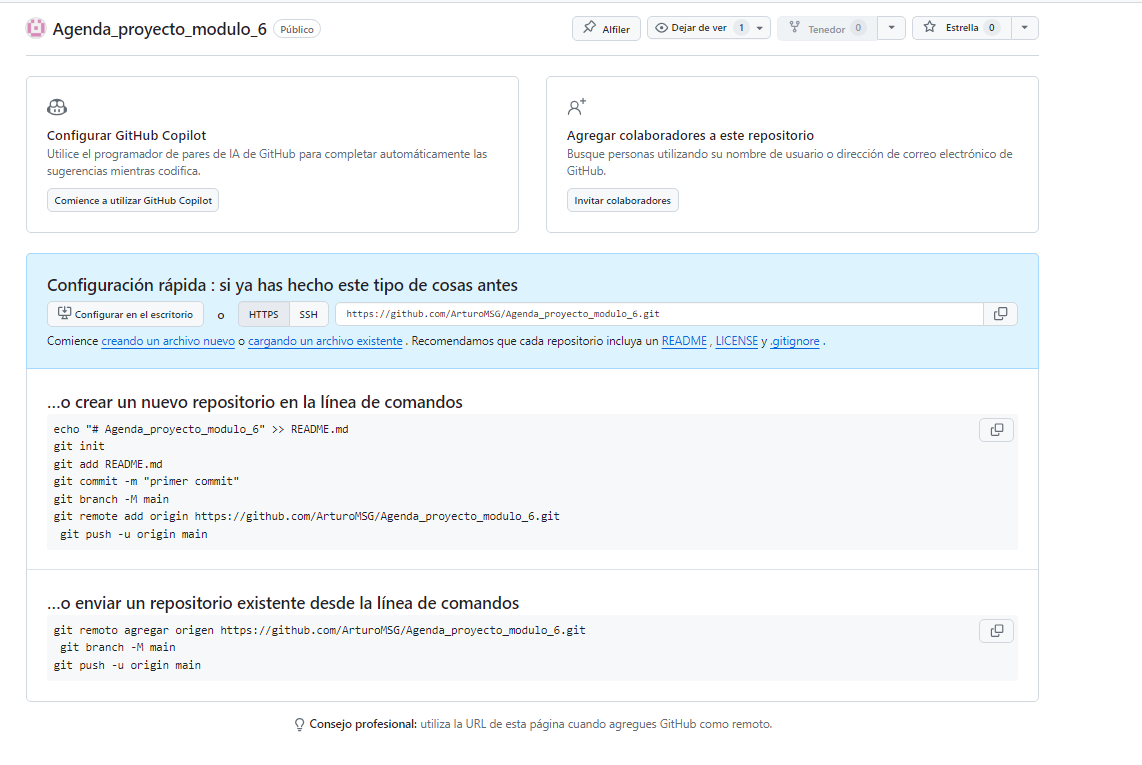


Ejemplo

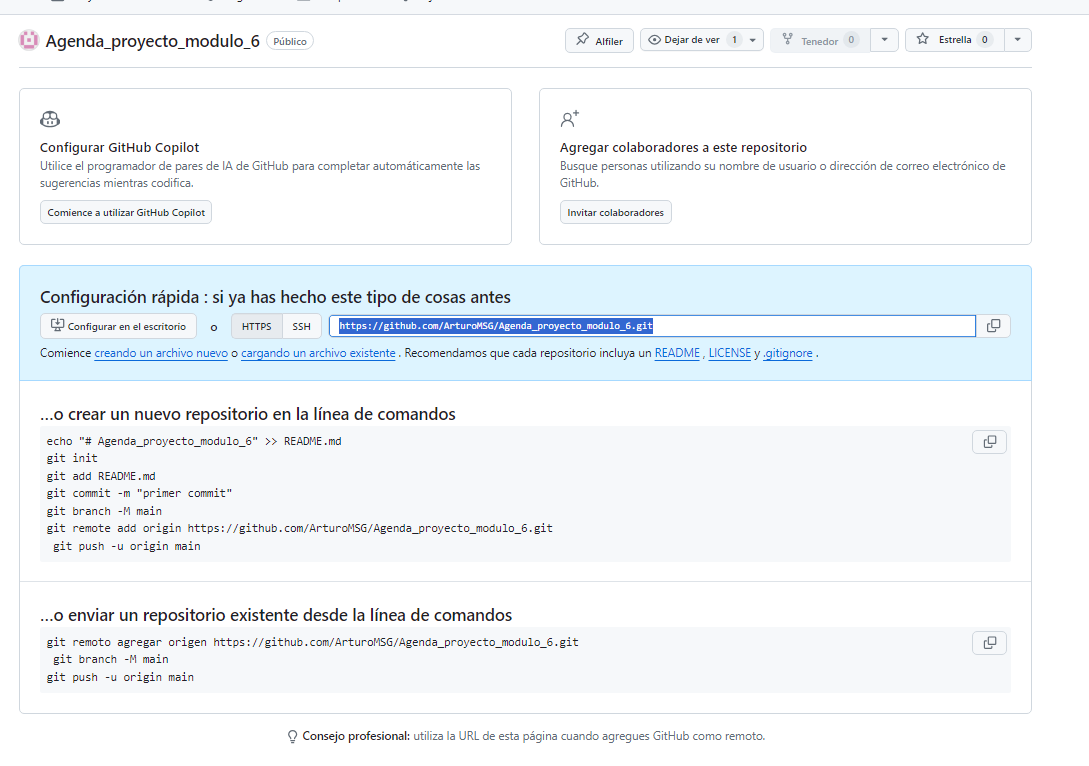


Se presiona en el botón crear repositorio

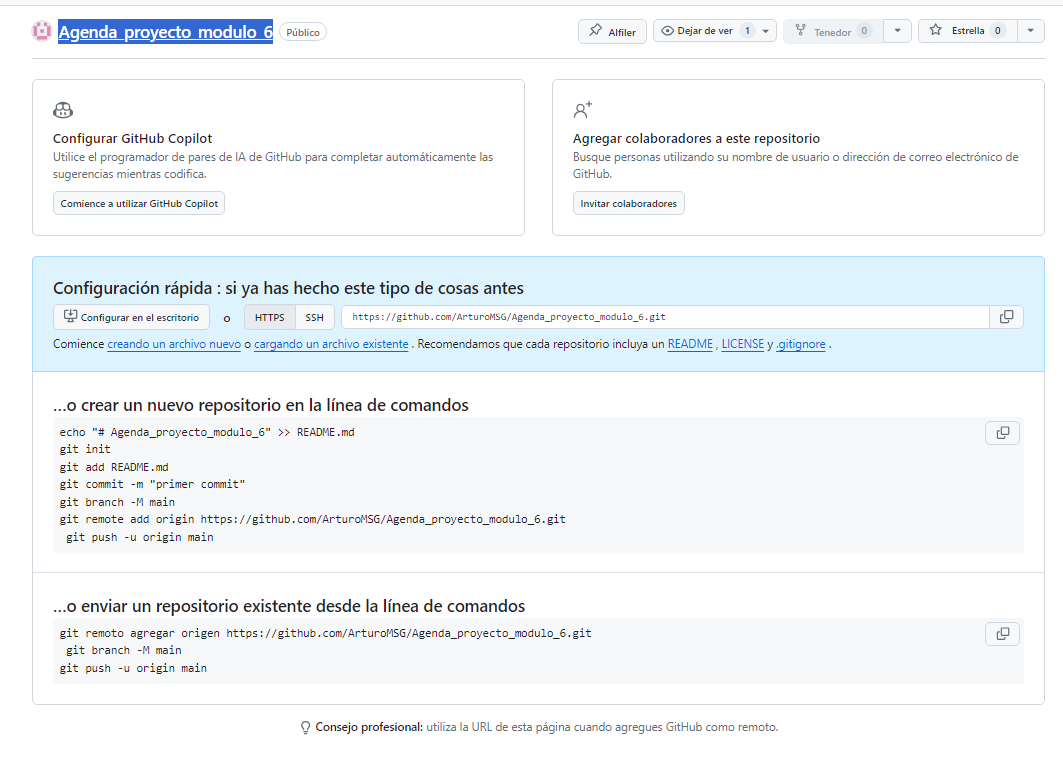
Se crea y te pone una página con indicaciones para que puedas subir tu proyecto o seguir con los pasos de este tutorial para lo mismo.



Lo marcado es tu URL que vas a requerir para subir el proyecto puedes copiarla



Lo remarcado es el nombre de proyecto, por lo que puedes ver la URL ya lleva el nombre del proyecto



Las carpetas creadas que no tengan archivos no son subidas, pero los archivos, aunque estén sin información, si se suben

Crear el archivo README.md (aunque este archivo no tiene información).

* echo "# proyecto\_modulo\_6" >> README.md

agregas el archivo README.dm o puedes agregar todos los cambios hechos con git add .

* git add README.md

ó

* git add .

guardas los cambios de los archivos modificados con comentario.

* git commit -m "Comentario\_de:los\_cambios"

con el siguiente comando cambias al usuario principal y lo nombras “main”

* git branch -M main

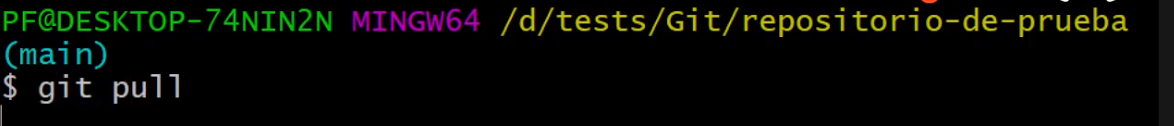
con el siguiente comando indicas la URL donde se alojara tu proyecto, es la URL del repositorio que creaste en GitHub

* git remote add origin <https://github.com/ArturoMSG/Agenda_proyecto_modulo_6>

con el siguiente comando se sube la información al repositorio de GitHub con todos los comentarios

* git push -u origin main

ó

* git push origin main
* para cuando el proyecto se modifica en GitHub y la localmente, debemos tener la siguiente consideración, antes de subir algún cambio realizado remotamente, se debe actualizar los cambios que están en GitHub con el siguiente 

Después ya puedo subir cambios con git push origin main o como corresponda.

