

# Instructivo para crear un repositorio en GITHUB

## Contenido

Descripción.....	2
Pasos para la instalación .....	3
Paso 1 .....	3
Paso 2 .....	3
Paso 3 .....	4
Paso 4 .....	5
Paso 5 .....	6
Paso 6 .....	8
Paso 7 .....	9
Paso 8 .....	10
Paso 9 .....	10
Paso 10.....	12
Paso 11.....	13

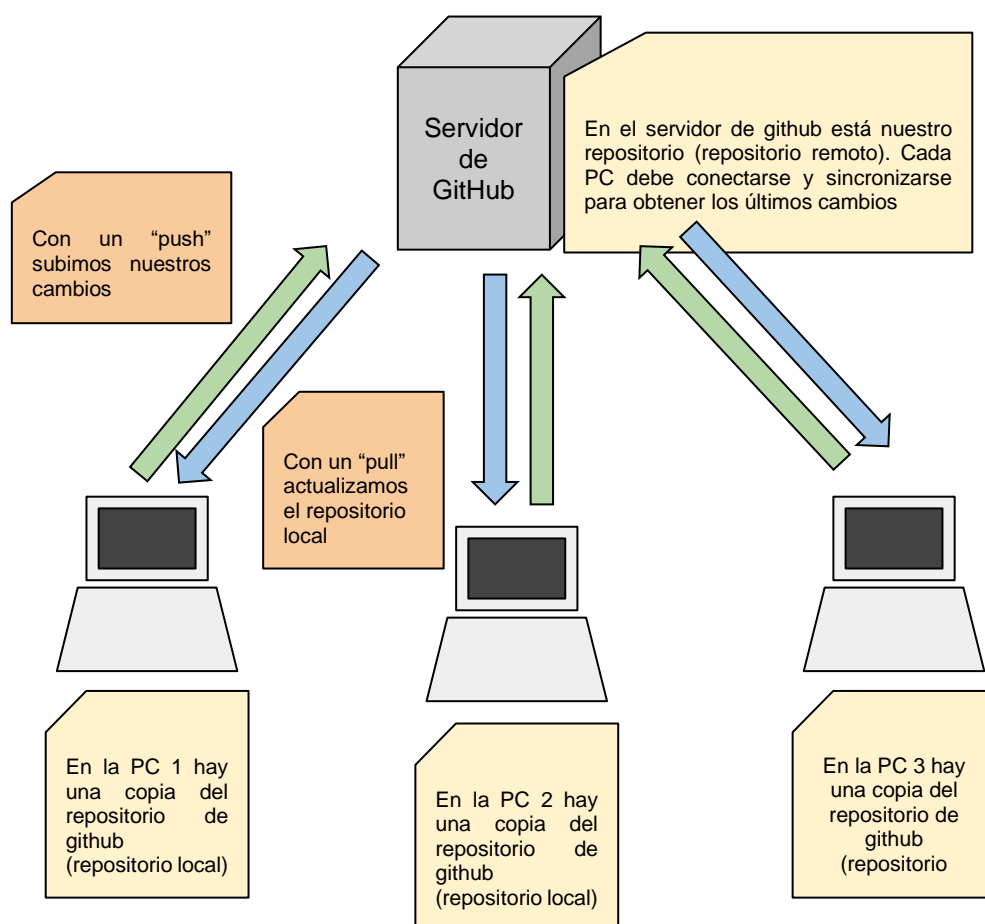
## Descripción

**GitHub:** Página web que nos permite crear repositorios para alojar archivos y versionarlos usando git. Existen otras páginas como bitbucket, gitlab, etc

**Git:** Herramienta que se utiliza para versionar archivos.

**Repositorio remoto:** Repositorio ubicado en un servidor.

**Repositorio local:** Repositorio ubicado en la computadora del programador



**¡IMPORTANTE!** Cuando trabajamos en equipo:

- siempre hacer un pull antes de empezar a programar, para tener nuestro repositorio local actualizado.
- siempre hacer un push luego de terminar de programar, para subir los cambios al repositorio remoto

## Pasos para la instalación

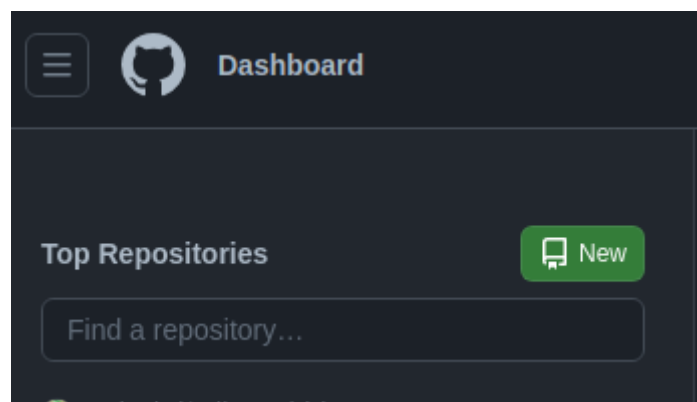
### Paso 1

Crear una cuenta en [github.com](https://github.com) y/o iniciamos sesión

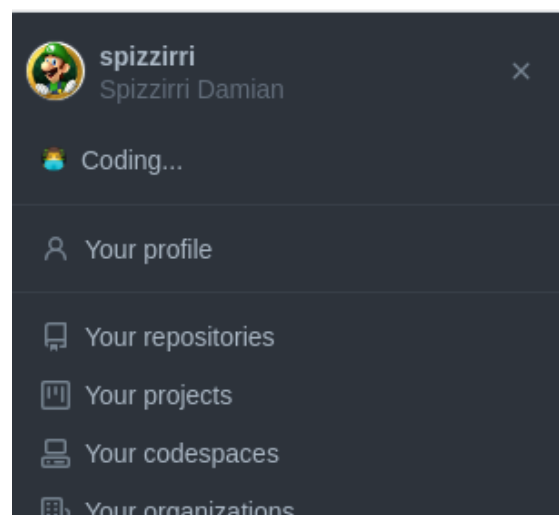
### Paso 2

Navegamos a la vista para crear un repositorio.

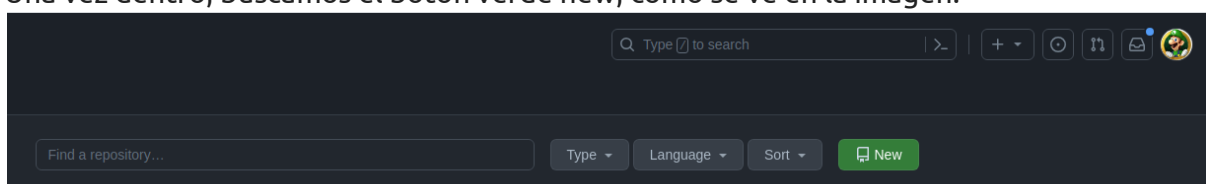
Opción 1: Buscamos el botón verde new en la pantalla principal, como se ve en la imagen.



Opción 2: Hacemos click en nuestro avatar y buscamos la opción "Your repositories"



Una vez dentro, buscamos el botón verde new, como se ve en la imagen.

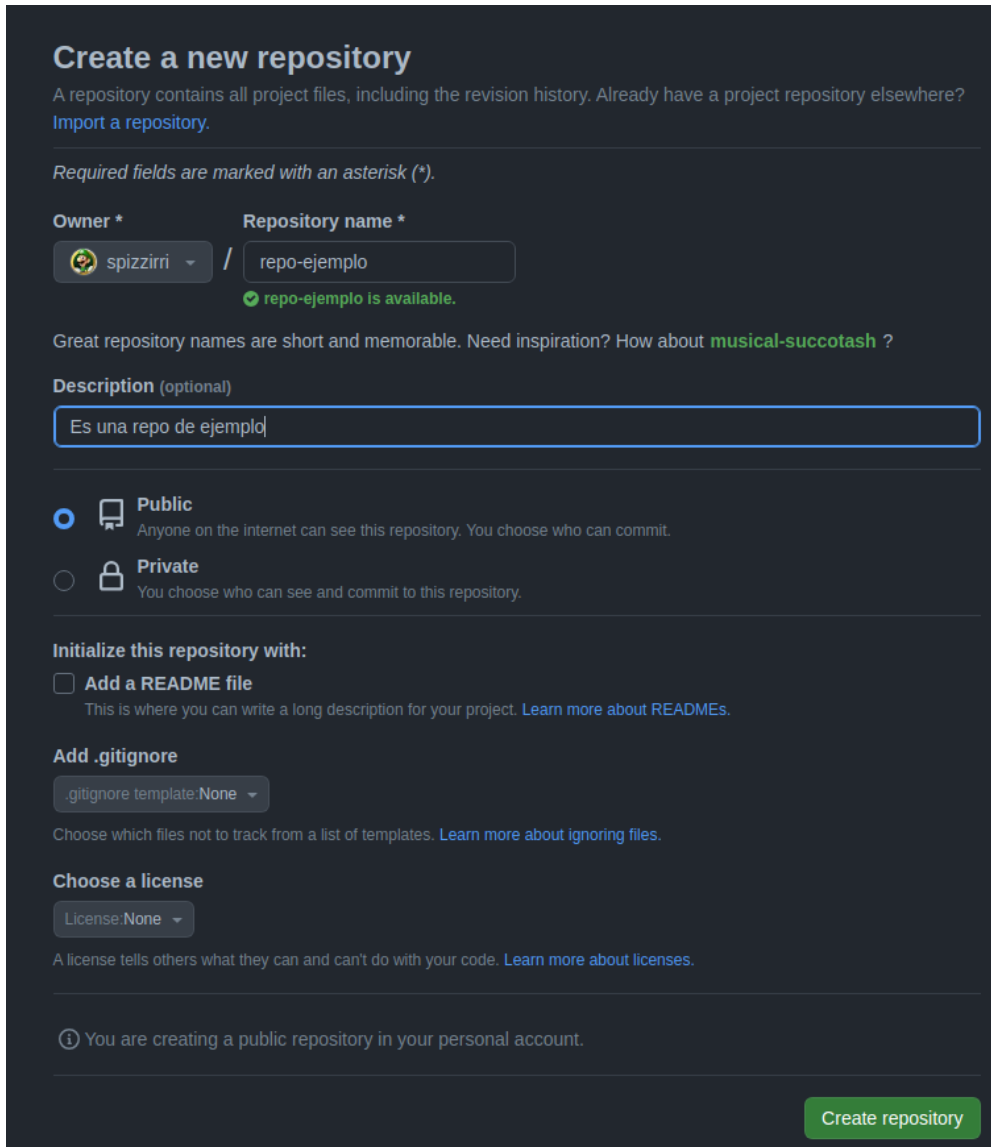


## Paso 3

Luego de dar click en el botón verde new, debemos completar los campos:

- Repository name: Elegir un nombre para nuestro repositorio.
- Description: Completar con una breve descripción

Finalmente damos click a Create repository




**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 \***

 spizzirri / repo-ejemplo

repo-ejemplo is available.

Great repository names are short and memorable. Need inspiration? How about **musical-succotash** ?

**Description (optional)**

Es una repo de ejemplo

☒ **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.

**Create repository**

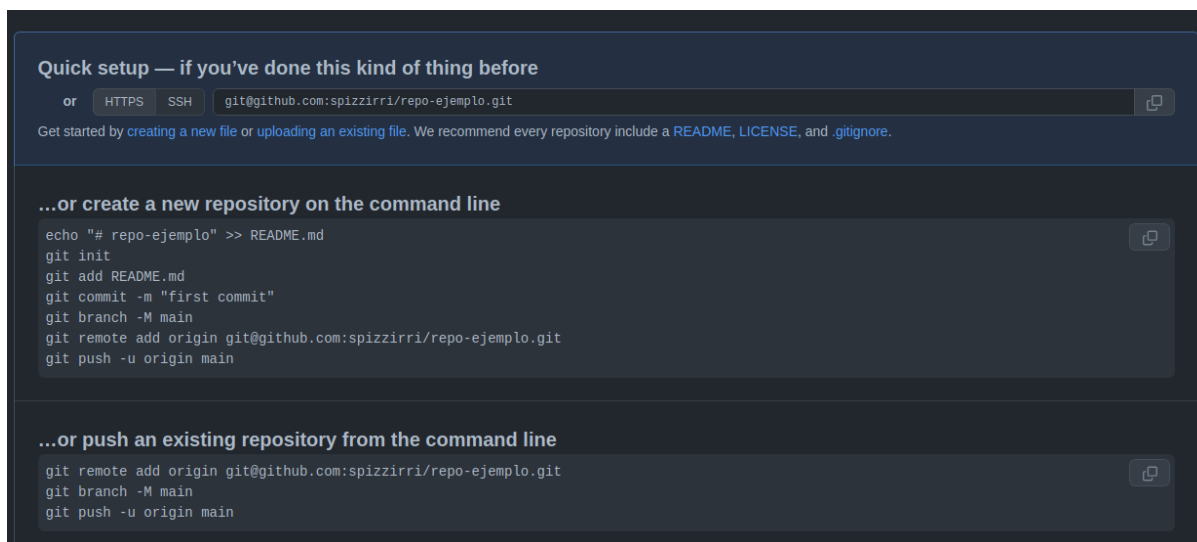
## Paso 4

Listo, nuestro repositorio está creado en github, ahora nos toca conectar el código que tenemos en nuestra computadora para poder subirlo al repo.

En la web, nos encontraremos con una vista en la que veremos las instrucciones para configurar y subir el código.

Tenemos dos opción para configurar el repositorio remoto contra el repositorio local

- **Opción 1:** instalar git y configurar una clave SSH. (Recomendado, pero no lo vamos a ver en este tutorial. El alumno es libre de investigarlo y utilizar esa manera)
- **Opción 2:** instalar github desktop.



The screenshot shows the GitHub 'Quick setup' interface. It has a dark theme. At the top, it says 'Quick setup — if you've done this kind of thing before'. Below this, there are three tabs: 'or', 'HTTPS', and 'SSH'. The 'SSH' tab is selected, and it shows the URL 'git@github.com:spizzirri/repo-ejemplo.git'. Below the tabs, there is a note: 'Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.' Below this, there are two sections. The first section is titled '...or create a new repository on the command line' and contains a list of commands: 'echo "# repo-ejemplo" >> README.md', 'git init', 'git add README.md', 'git commit -m "first commit"', 'git branch -M main', 'git remote add origin git@github.com:spizzirri/repo-ejemplo.git', and 'git push -u origin main'. The second section is titled '...or push an existing repository from the command line' and contains the commands: 'git remote add origin git@github.com:spizzirri/repo-ejemplo.git', 'git branch -M main', and 'git push -u origin main'. Each section has a copy icon to its right.

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

or  HTTPS  SSH  git@github.com:spizzirri/repo-ejemplo.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 "# repo-ejemplo" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin git@github.com:spizzirri/repo-ejemplo.git
git push -u origin main

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

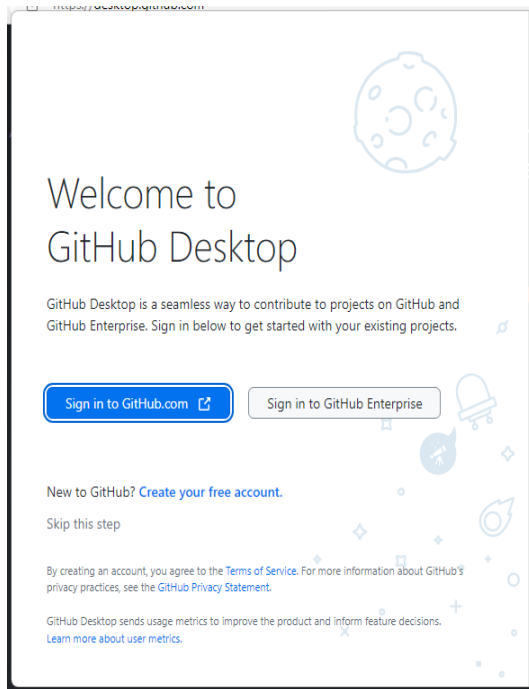
git remote add origin git@github.com:spizzirri/repo-ejemplo.git
git branch -M main
git push -u origin main
```

## Paso 5

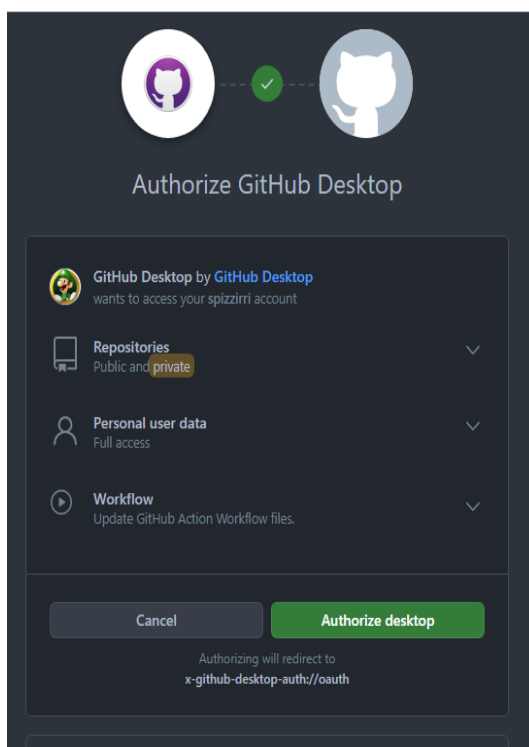
Descargar e instalar github desktop

<https://desktop.github.com/>

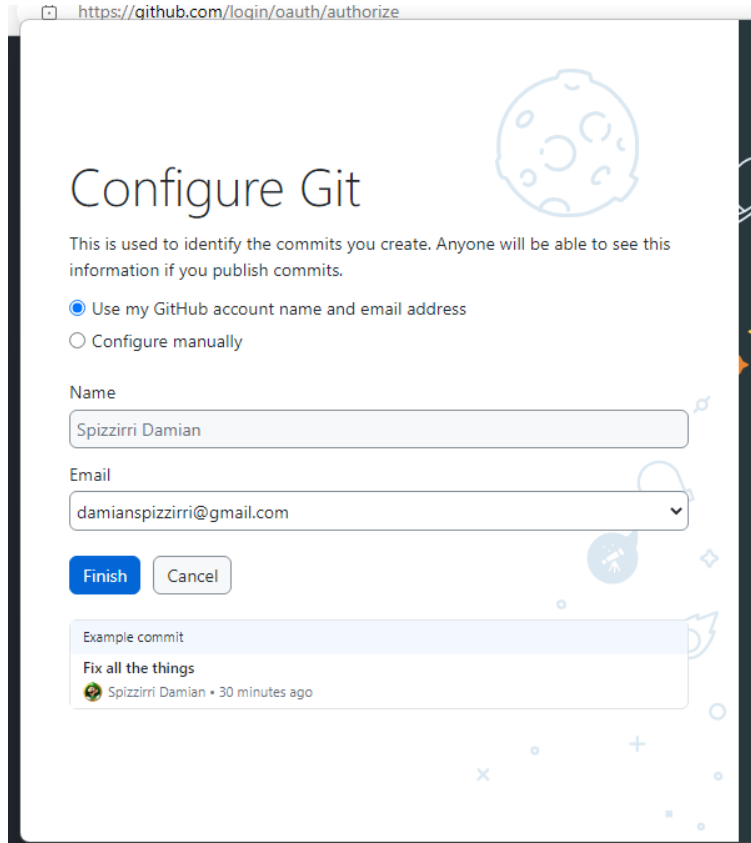
Al instalar nos pedirá iniciar sesión



Al iniciar sesión nos pedirá permiso para acceder a nuestros repositorios



Al aceptar los permisos nos mostrará la configuración para nuestros futuros commits.



The screenshot shows the GitHub 'Configure Git' page in a web browser. The URL in the address bar is <https://github.com/login/oauth/authorize>. The page has a light blue header with the GitHub logo. Below the header, the title 'Configure Git' is displayed in a large, dark blue font. A sub-header explains: 'This is used to identify the commits you create. Anyone will be able to see this information if you publish commits.' There are two radio buttons: 'Use my GitHub account name and email address' (selected) and 'Configure manually'. Below these are input fields for 'Name' (containing 'Spizzirri Damian') and 'Email' (containing 'damianspizzirri@gmail.com'). At the bottom left are 'Finish' and 'Cancel' buttons. At the bottom right is an 'Example commit' section showing a commit titled 'Fix all the things' by 'Spizzirri Damian' from '30 minutes ago'.

Configure Git

This is used to identify the commits you create. Anyone will be able to see this information if you publish commits.

☒ Use my GitHub account name and email address  
☐ Configure manually

Name  
Spizzirri Damian

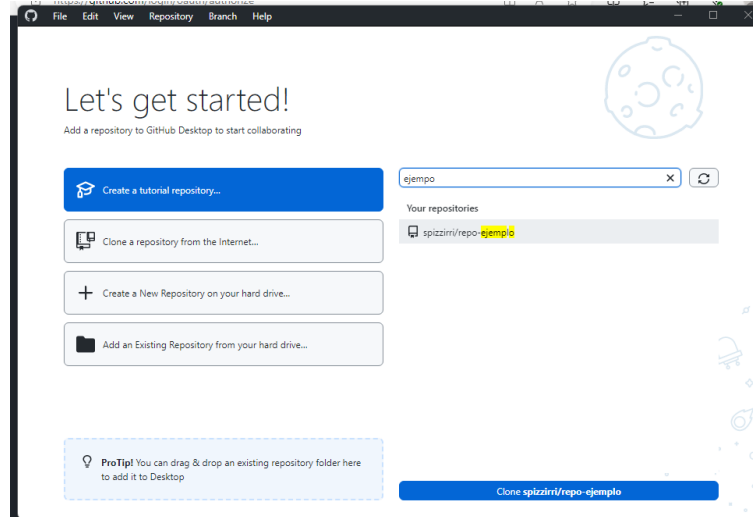
Email  
damianspizzirri@gmail.com

Finish Cancel

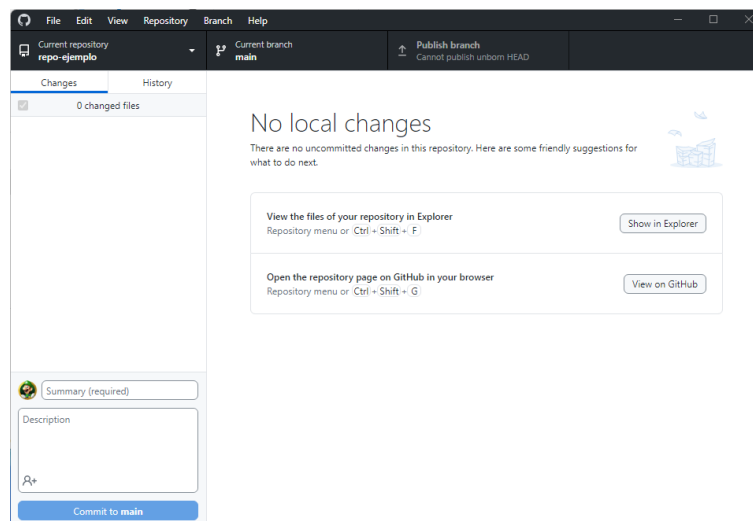
Example commit  
Fix all the things  
Spizzirri Damian • 30 minutes ago

## Paso 6

Buscamos el repositorio que habíamos creado y lo clonamos presionando el botón Clone. Al clonarlo, estaremos creando una copia del repositorio remoto en nuestra computadora.



Nos aparecerá un mensaje en el que nos dice donde se clonará ese repositorio. Más tarde veremos una pantalla en la dirá 0 changed files. Esto nos indica que entre nuestro repositorio local y el repositorio remoto hay 0 diferencias.





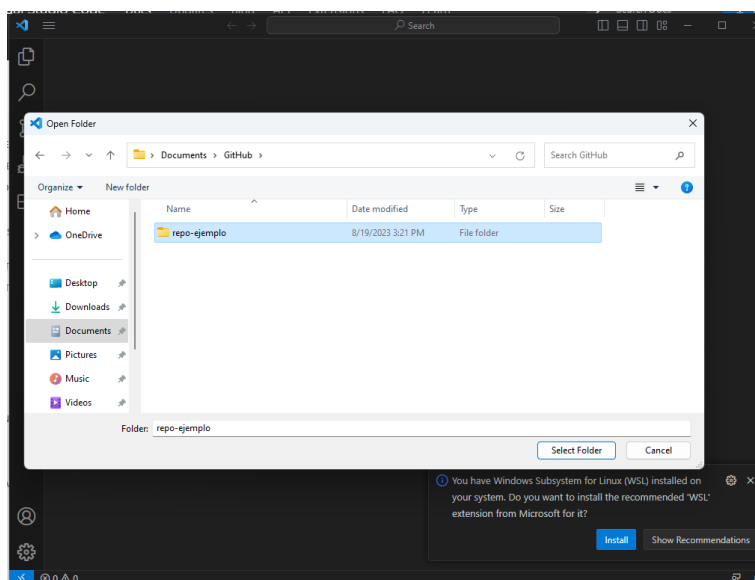
## Paso 7

Ahora vamos a abrir nuestro repositorio con el editor de código Visual Studio code. El editor que usemos no es relevante, pero usaremos ese a modo de ejemplo.

<https://code.visualstudio.com/>

Buscamos la carpeta en donde se clonó el repositorio y lo abrimos.

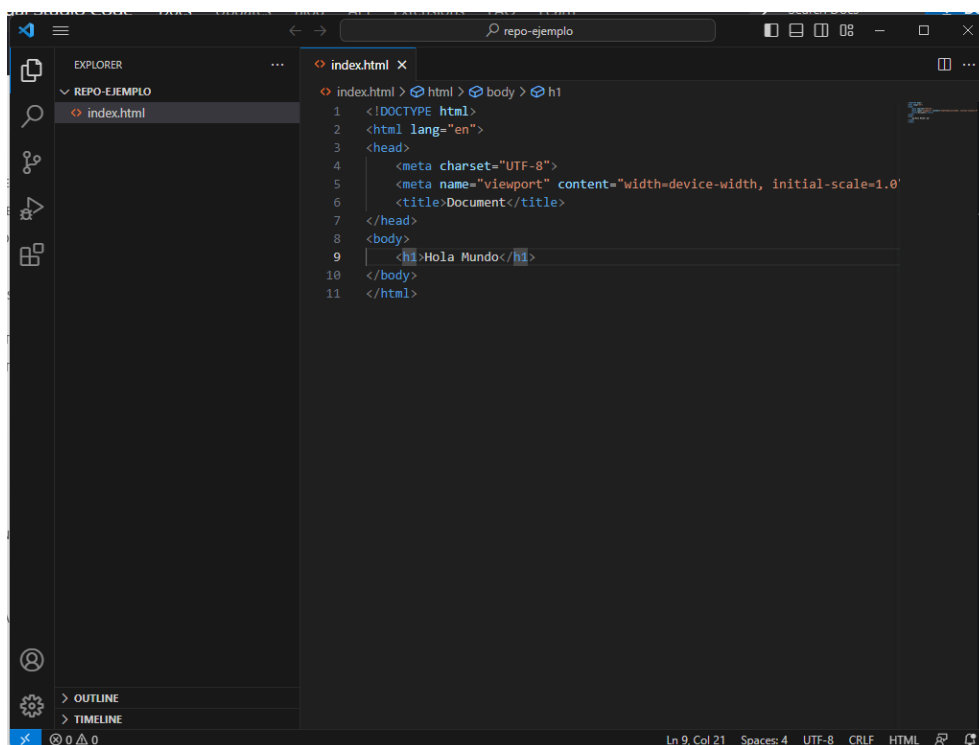
En caso de que no nos acordemos dónde está el código podemos fijarnos en github desktop en la opción repository → show in explorer.



Una vez que abrimos la carpeta veremos que está vacía porque el repositorio es totalmente nuevo.

## Paso 8

Vamos a agregar un archivo index.html a modo de ejemplo para ver como subir los archivos



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Document</title>
7 </head>
8 <body>
9   <h1>Hola Mundo</h1>
10 </body>
11 </html>
```

## Paso 9

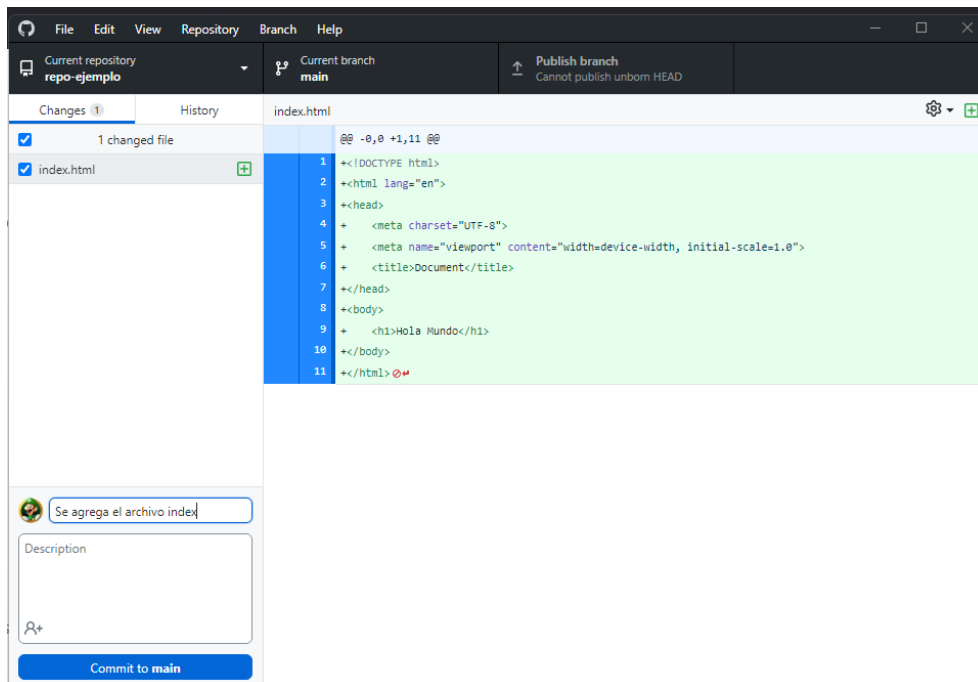
Luego de guardar el archivo vemos que en github desktop nos aparece que hay cambios para subir.

En este caso es uno solo. (1 changed file).

Lo que tenemos que hacer ahora es:

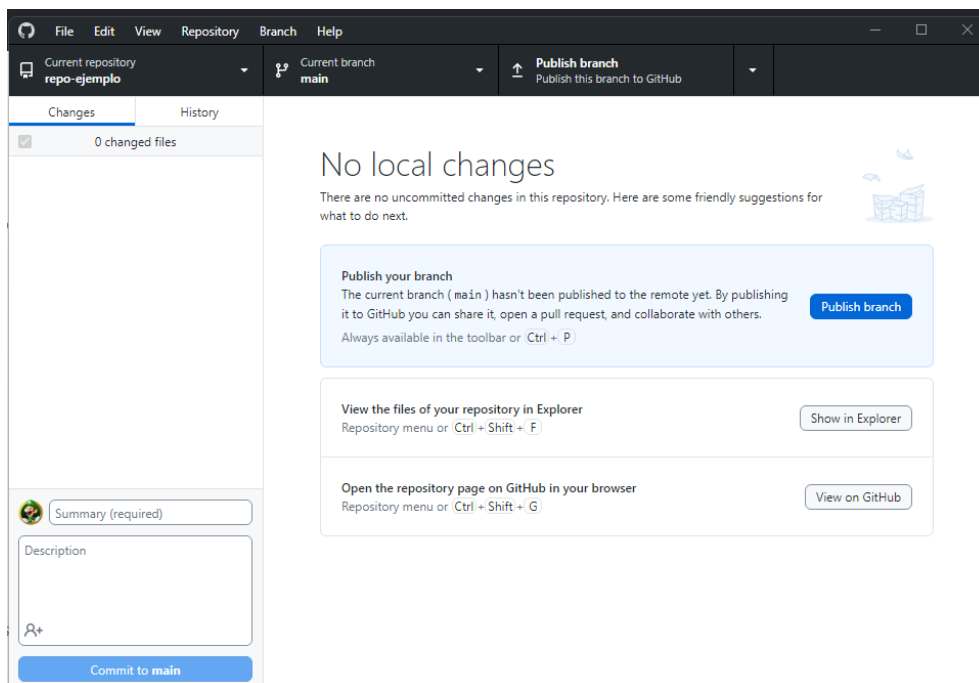
1. Confirmar los cambios (commit)
2. Subir los cambios al repositorio de github (push)

Para hacer commit, escribimos un comentario como se ve en la imagen y hacemos click en el botón commit. Esto confirmará los cambios, pero todavía tenemos el código en nuestra computadora.

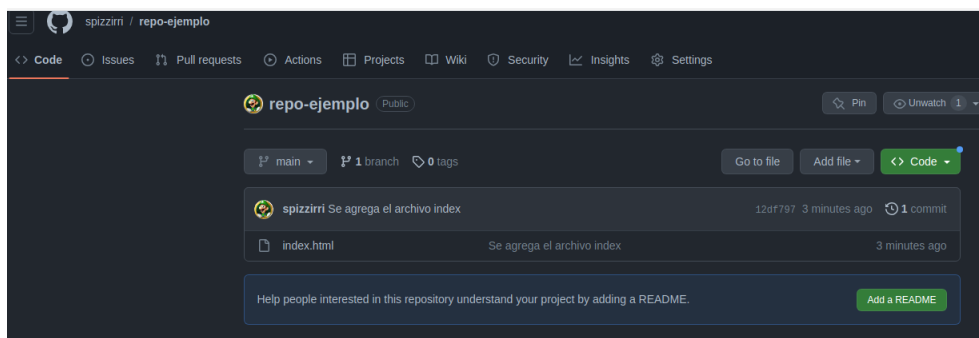


Por ser la primera vez, luego de hacer el primer commit veremos este mensaje.  
Hagamos click push para subir nuestros cambios a github.

## Paso 10

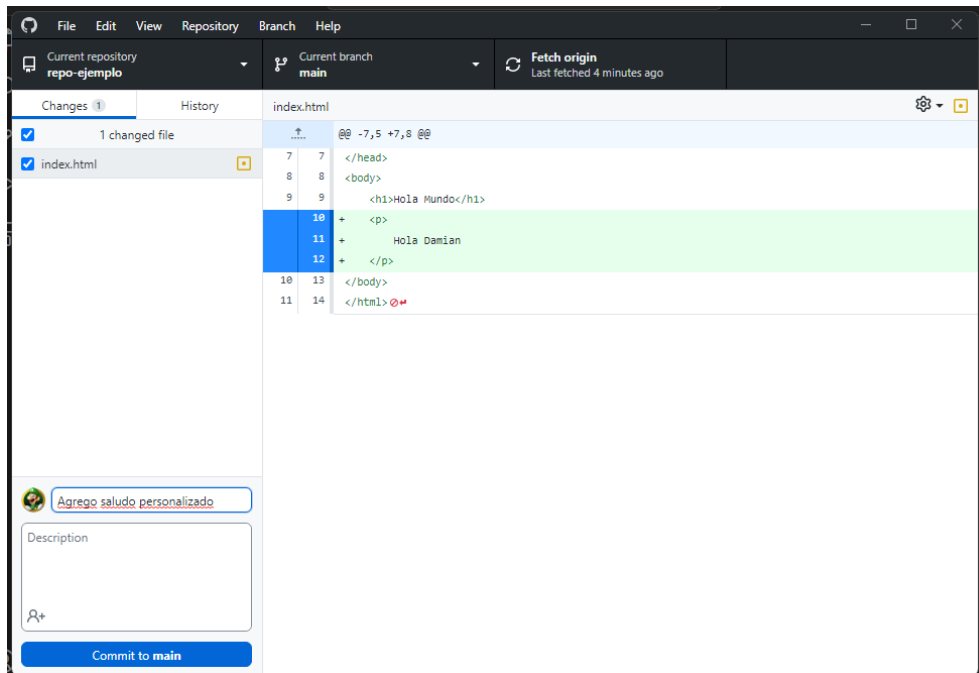


Luego de haber hecho el push, vayamos a github para confirmar que están nuestros cambios.



## Paso 11

Hagamos otro cambio, escribimos un comentario y confirmamos haciendo click en commit



Y luego subimos el código a github haciendo click en push origin

