

Github es un sistema de control de versiones, que nos ayuda a llevar un orden completo de las modificaciones y avances de un trabajo. Sirve para poder trabajar un proyecto en diferentes ordenadores y poder acceder a lo que has hecho anteriormente con muchas más opciones.

En Github podemos hacer muchas cosas pero las más significativas y importantes son las siguientes:

Podemos hacer que podamos tirar hacia atrás en un trabajo subiendo commits por si ocurre algún problema y no sabemos qué hacer volvemos a como lo teníamos antes.

Trabaja con ramas para poder editar cosas sin tener que tocar el trabajo principal.

Se pueden crear clones de los repositorios para poder hacer trabajo simultáneo en diferentes ordenadores y después juntarlo todo.

Puedes trabajar sin conexión a la red

Trabaja con plataformas de desarrollo colaborativo para poder tener los repositorios en la nube.

Una vez instalado Github tendremos que configurar nuestro nombre de usuario y email con estos comandos en la consola.

```
git config --global user.name "Tu nombre"
git config --global user.email tugmail@gmail.com
```


Para crear un repositorio solo tendremos que darle al botón de Nuevo en Crear un nuevo repositorio

Y cuando lo configuremos tendremos que ponerlo público si queremos que más gente lo pueda ver y podemos subirlo a github pages por ejemplo y añadir README para que venga inicializado ya por defecto.

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 *  SamuelMartin3 /

Great repository names are short and memorable. Need inspiration? How about [urban-computing-machine](#) ?

Description (optional)

- ☒  **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

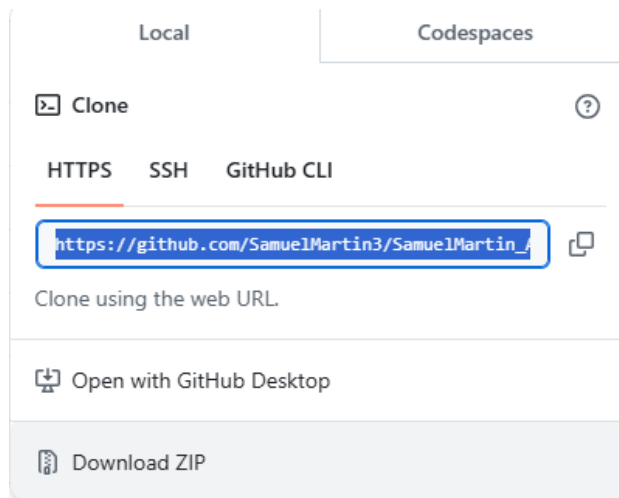
License: **None**

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

Una vez tenemos el repositorio creado tenemos que clonarlo en nuestra máquina local para poder trabajar y después subirlo a github para hacer las pruebas fuera por si ocurre algo. Primero de todo tendremos que copiar la URL del repositorio y después en nuestra consola en la raíz del repositorio tendremos que poner los siguientes comandos.



- git clone URL del repositorio
- git init (para iniciarlo)
- git add "archivo" o git add . (para subir el archivo o todo)
- git commit -m "nombre del commit" (para saber que hemos subido por si tenemos que ir hacía atrás saber que es lo último que hemos hecho)
- git push origin main (para subirlo a la rama principal del repositorio)

Todo esto si ya tenemos el repositorio inicializado por el README si no hemos marcado esta opción tendremos que poner antes de lo anterior estos comandos

- git init
- git add README.md (para iniciar el repositorio en el local)
- git branch -M main (para asociarlo a una rama)
- git commit -m "first commit"
- git remote add origin https://github.com/TuNombre/Prueba.git. (para asociar el local con el remoto)
- git push -u origin main

Y para sincronizarlo tendremos que poner los comandos de diferente forma git init

```
-git remote add origin https://github.com/TuNombre/Prueba.git.  
-git branch -M main  
-git add .  
-git commit -m "first commit"  
-git push -u origin main
```

Y por último para importar un repositorio ya creado solo tenemos que copiar la URL y a la hora de crear el nuevo repositorio nos dará la opción de importar un repositorio y tendremos que copiar la URL allí.

Import your project to GitHub

Import all the files, including revision history, from another version control system.

Required fields are marked with an asterisk ().*

Support for importing Mercurial, Subversion and Team Foundation Version Control (TFVC) repositories ended on April 12, 2024. For more details, see the [changelog](#).

Your source repository details

The URL for your source repository *

Learn more about [importing git repositories](#).

Si queremos utilizar Github Pages tendremos que ir al repositorio y en sus opciones de ajustes tendremos que ir al apartado de Pages y poner la rama principal y la carpeta root cuando hayamos hecho eso nos saldrá la URL del Pages y ya. Es muy importante que tengamos el index.html para que lo reconozca y podamos acceder.

Code and automation

Branches

Tags

Rules

Actions

Webhooks

Environments

Codespaces

Pages

Last deployed by SamuelMartin2 last week

Build and deployment

Source

Deploy from a branch

Branch

Your GitHub Pages site is currently being built from the `main` branch. [Learn more about configuring the publishing source for your site.](#)

main

/ (root)

Save