



Guía de instalación de Git y Visual Studio Code en Raspberry Pi

Git en Raspberry Pi

1.Instalar Git.

Lo que aprenderá en esta guía, es a instalar y utilizar el software de control de versiones de Git para trabajar en tus propios proyectos dentro de Raspberry.

Lo que vas a aprender:

- Instalar Git en Raspberry
- Crear un Repositorio y subirlo a Github

Vamos a utilizar los conocimientos de SSH Raspberry (<https://github.com/fullmakeralchemist/raspberryssetupes>) en esta guía utilizando el Shell de Raspberry desde Putty y vamos a ingresar el siguiente comando:

```
sudo apt install git
```

```
pi@raspberrypi: ~
```

```
pi@raspberrypi:~ $ sudo apt install git
```

```
pi@raspberrypi: ~
```

```
pi@raspberrypi:~ $ sudo apt install git
Reading package lists... Done
Building dependency tree
Reading state information... Done
git is already the newest version (1:2.20.1-2+deb10u3).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
pi@raspberrypi:~ $
```

2.Configurando Git.

Lo primero que debemos hacer es ingresar a Git nuestra información. Esto es importante, ya que muchas personas pueden usar Git de forma colaborativa, por lo que es necesario saber quién realizó cambios en qué archivos. Puede usar su propio nombre de usuario y dirección de correo electrónico.

Así que ingresaremos los siguientes comandos:

```
git config --global user.name "usuariogit"
```

```
git config --global user.email "emailusuariogit@mail.com"
```

```
pi@raspberrypi: ~
```

```
pi@raspberrypi:~ $ git config --global user.name "USERGIT"
```

```
pi@raspberrypi: ~
```

```
pi@raspberrypi:~ $ git config --global user.email "user@mail.com"
```

A continuación, debes decirle a Git qué editor de texto desea usar. Si no tienes un editor favorito, simplemente puedes escribir:

```
git config --global core.editor nano
```

```
pi@raspberrypi: ~
```

```
pi@raspberrypi:~ $ git config --global core.editor nano
```

3.Creando nuestro primer proyecto

¿Quieres empezar un nuevo proyecto? Tal vez sea un bot como Jarvis que te cuide en todo momento. Necesitará de un directorio en su computadora para todos sus archivos, así que lo primero que debe hacer es crear ese directorio.

En la terminal, puedes usar el comando ls (en lista el contenido del directorio actual) para revisar las carpetas y archivos.

En la terminal, puedes usar el comando cd (cambio de directorio) para cambiar un nuevo directorio.

En la terminal, puedes usar el comando mkdir (crear directorio) para crear un nuevo directorio.

```
ls
```

```
cd Desktop
```

```
mkdir snitch-sniffer
cd snitch-sniffer
```

```
pi@raspberrypi: ~/Desktop/snitch-sniffer
pi@raspberrypi:~ $ ls
Bookshelf Documents Music Public Videos
Desktop Downloads Pictures Templates
pi@raspberrypi:~ $ cd Desktop
pi@raspberrypi:~/Desktop $ mkdir snitch-sniffer
pi@raspberrypi:~/Desktop $ cd snitch-sniffer
pi@raspberrypi:~/Desktop/snitch-sniffer $
```

A continuación, puedes crear un archivo que le dirá a las personas de qué se trata el proyecto. Puede utilizar cualquier editor de texto para hacer esto, como el Bloc de notas o mas adelante cuando tengamos el Visual Studio Code puedes hacer un segundo archivo para practicar. Crea un archivo llamado README.md, La extensión .md significa Markdown, que es un lenguaje de marcado. Puede obtener más información sobre Markdown aquí (<https://daringfireball.net/projects/markdown/>). Para crear un archivo desde el Shell de Raspberry utilizaremos el comando:

```
nano README.md
```

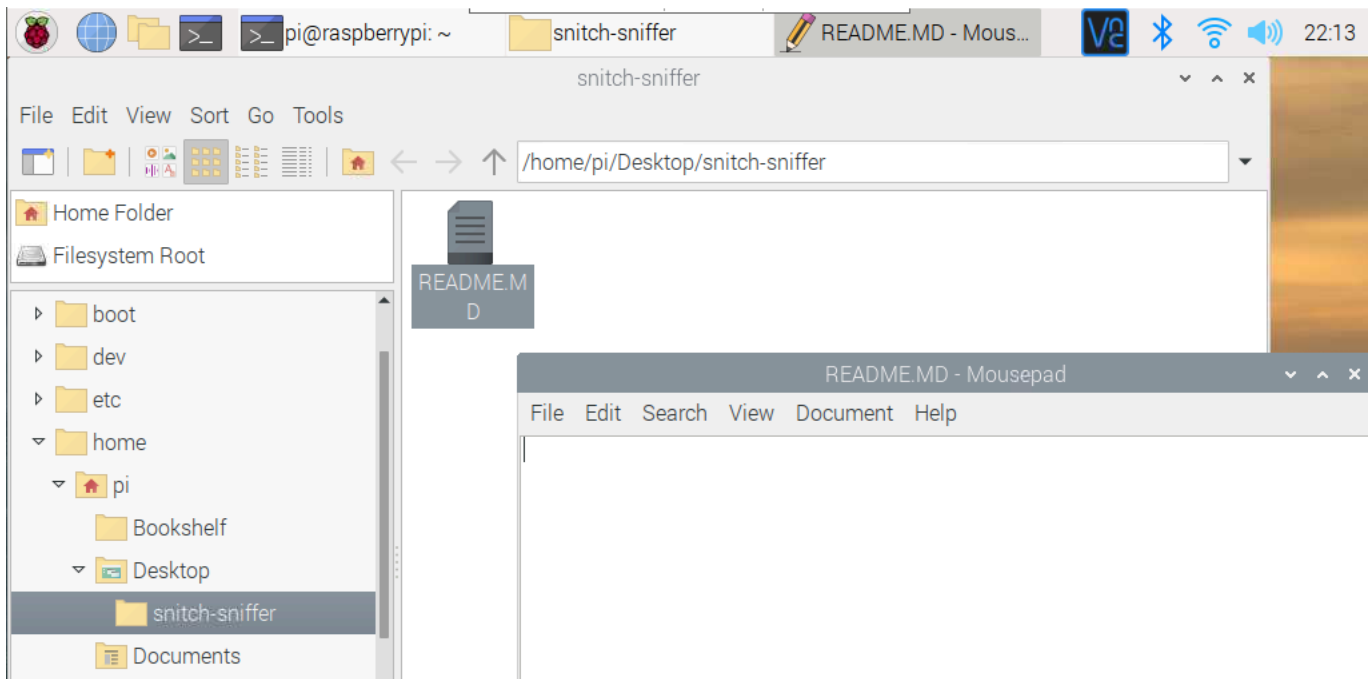
Ahora podremos ingresar texto en nuestro archivo, para guardar el archivo “Ctrl+O” y finalmente para salir del editor “Ctrl+X”.

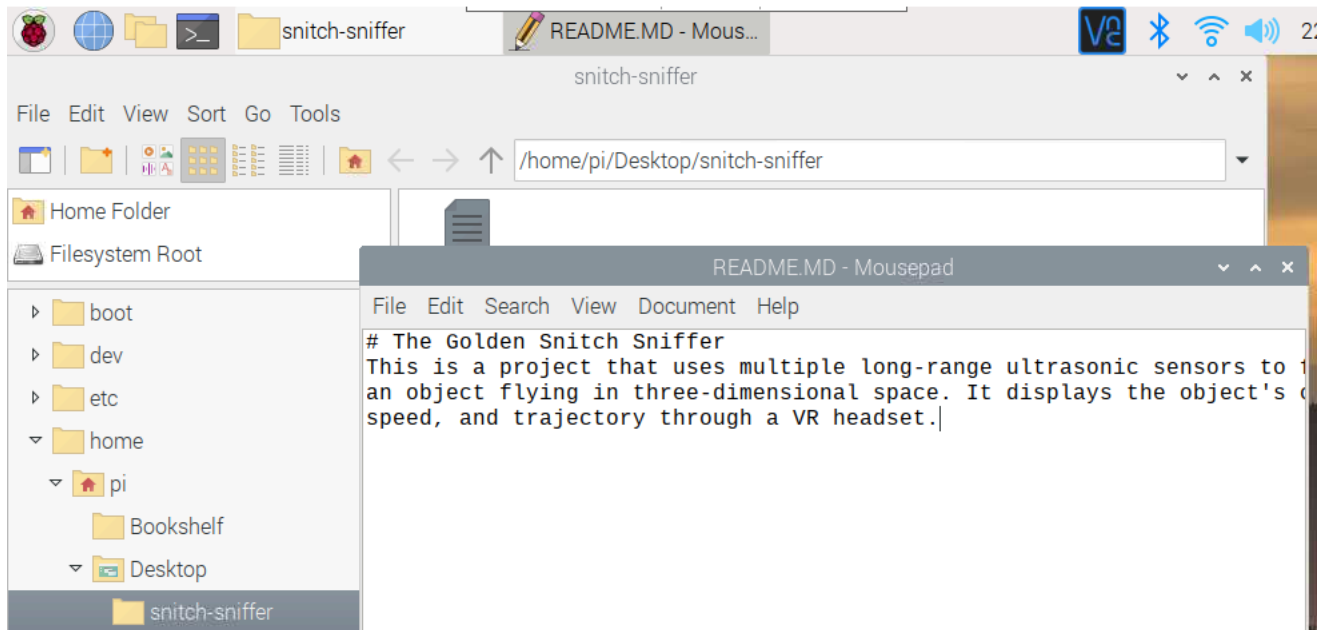
Una segunda opción es utilizar los conocimientos anteriores de Wireless Setup (<https://github.com/fullmakeralchemist/raspberryssetups>), para ingresar por VNC y crear el archivo. Como en cualquier ordenador usaremos clic derecho, opción nuevo y le damos un nombre y extensión al archivo.

Yo ingrese el siguiente texto en el archivo README.md

```
# The Golden Snitch Sniffer
```

```
This is a project that uses multiple long-range ultrasonic sensors to find and track an object flying in three-dimensional space. It displays the object's coor
```





Su archivo debería haber sido creado y ahora estará ubicado en su directorio. Puede escribir en la terminal para ver una lista de archivos el siguiente comando.

```
ls
```

Por el momento, el directorio es como cualquier otro de su sistema. Ahora necesitas hacer la parte de el control de versiones. Esto se conoce como repositorio de Git y toma la forma de un directorio oculto que realiza un seguimiento de todos los cambios en el directorio de trabajo. Escriba lo siguiente para crear el repositorio, que de ahora en adelante solo se llamará repositorio:

```
git init
```

```
pi@raspberrypi: ~/Desktop/snitch-sniffer
pi@raspberrypi:~/Desktop/snitch-sniffer $ git init
Initialized empty Git repository in /home/pi/Desktop/snitch-sniffer/.git/
pi@raspberrypi:~/Desktop/snitch-sniffer $
```

Si vuelve a escribir ls, no parecerá que haya cambiado nada. Sin embargo, puede usar `ls -a` para ver todos los archivos y directorios ocultos.

Ahora debería ver algo como esto en la ventana de su terminal:

```
. . . .git README.md
```

Ese directorio .git es el esqueleto del repositorio. Puede echar un vistazo al interior escribiendo lo siguiente.

```
ls -a .git
```

Así que ahora tienes la parte de la mochila mágica, pero aún no le has agregado nada. Ese archivo README.md aún no se ha colocado en la bolsa. Debe decirle a Git que desea agregar el archivo README.md al repositorio. Para hacer esto, simplemente escriba:

```
git add README.md
```

Sin embargo, a veces es más fácil agregar todo al repositorio, en lugar de agregar archivos individuales. Para hacer esto, puede escribir:

```
git add --all
```

Ahora Git sabe que necesita realizar un seguimiento de todos los cambios que suceden en el archivo README.md. Puede ver el estado de su repositorio en cualquier momento escribiendo lo siguiente:

```
git status
```

La respuesta anterior le indica que el archivo README.md aún no se ha confirmado. Esto significa que, aunque Git conoce el archivo, todavía no tiene almacenado ninguno de sus contenidos. La forma más sencilla de realizar una confirmación es escribiendo:

```
git commit -am "add README.md"
```

Esto confirma todos los cambios que ha realizado en el directorio en el repositorio de Git y agrega un mensaje que dice lo que hizo. El mensaje puede ser cualquier cosa en realidad, pero es mejor que sea bastante breve pero descriptivo de lo que ha cambiado.

```
pi@raspberrypi: ~/Desktop/snitch-sniffer
pi@raspberrypi:~/Desktop/snitch-sniffer $ git init
Initialized empty Git repository in /home/pi/Desktop/snitch-sniffer/.git/
pi@raspberrypi:~/Desktop/snitch-sniffer $ ls -a
. .. .git README.md
pi@raspberrypi:~/Desktop/snitch-sniffer $ ls -a .git
. .. branches config description HEAD hooks info objects refs
pi@raspberrypi:~/Desktop/snitch-sniffer $ git add README.md
pi@raspberrypi:~/Desktop/snitch-sniffer $ git add --all
pi@raspberrypi:~/Desktop/snitch-sniffer $ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   README.md

pi@raspberrypi:~/Desktop/snitch-sniffer $ git commit -am "add README.md"
[master (root-commit) 66eb018] add README.md
 1 file changed, 4 insertions(+)
 create mode 100644 README.md
pi@raspberrypi:~/Desktop/snitch-sniffer $
```

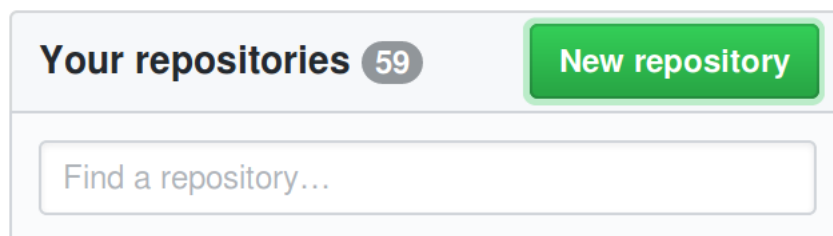
4.Trabajando con un Github

Ahora que sabe cómo hacer lo básico en Git, es hora de aprender a usarlo en todo su potencial: úselo para compartir su trabajo y colaborar con otros.

Hay muchos servicios que alojarán su repositorio de Git de forma gratuita. GitLab es uno de esos servicios y BitBucket es otro. En este recurso, utilizará GitHub, que es uno de los servicios más populares.


Lo primero que debe hacer es registrarse para obtener una cuenta en GitHub [<https://github.com/join?source=header-home>] y simplemente elegir el plan gratuito.

Ahora que tiene una cuenta, puede crear un repositorio de snitch-sniffer en GitHub. Busque el botón Nuevo repositorio y haga clic en él.




Asigne un nombre y una descripción al repositorio y haga clic en el botón Crear repositorio.

Owner**Repository name**

 **HarryPotter** ▼


 /


snitch-sniffer 

Great repository names are short and memorable. Need inspiration? How about **proba**

Description (optional)

3d tracking of objects in space

☒  **Public**
Anyone can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

☐ **Initialize this repository with a README**

This will let you immediately clone the repository to your computer. Skip this step if you're importing

Add .gitignore: **None** ▼

Add a license: **None** ▼



Create repository

A continuación, debería aparecer una página de instrucciones.

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

or HTTPS SSH `git@github.com:MarcScott/snitch-sniffer.git`

We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# snitch-sniffer" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin git@github.com:HarryPotter/snitch-sniff
git push -u origin master
```

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

```
git remote add origin git@github.com:HarryPotter/snitch-sniff
git push -u origin master
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Import code

Como ya tiene un repositorio listo para enviar a GitHub, entonces todo lo que necesita hacer es asegurarse de estar en el directorio de su proyecto y escribir:

```
git remote add origin git@github.com:HarryPotter/snitch-sniffer.git
```

y luego:

```
git push -u origin master
```

Si busca en GitHub, ahora debería poder ver su repositorio, junto con el archivo README.md que se muestra y que escribió.

6 commits

1 branch

Branch: master ▼

New pull request



Harry Potter new foo function



README.md

add README.md



quidditch-rules.json

add json rules and python program



snitch-sniffer.py

new foo function



README.md

The Golden Snitch Sniffer

This is a project that uses multiple long range ultrasonic sensors to find and display the objects coordinates, speed and trajectory through a VI

Cada vez que realiza cambios en su proyecto y desea enviarlos a GitHub, puede escribir:

```
git push origin master
```

Si está trabajando en una rama diferente, escribiría:

```
git push origin <branch-name>
```

Para saber mas acerca de Git en Raspberry te recomiendo ir a [Getting started with Git \[https://projects.raspberrypi.org/en/projects/getting-started-with-git\]](https://projects.raspberrypi.org/en/projects/getting-started-with-git).

Instalar Visual Studio Code en Raspberry.

1.Instalación.

Thonny IDE (entorno de desarrollo integrado) incluido con la última versión del sistema operativo Raspberry Pi OS. Thonny viene con Python 3.7 integrado, por lo que no es necesario instalar nada.

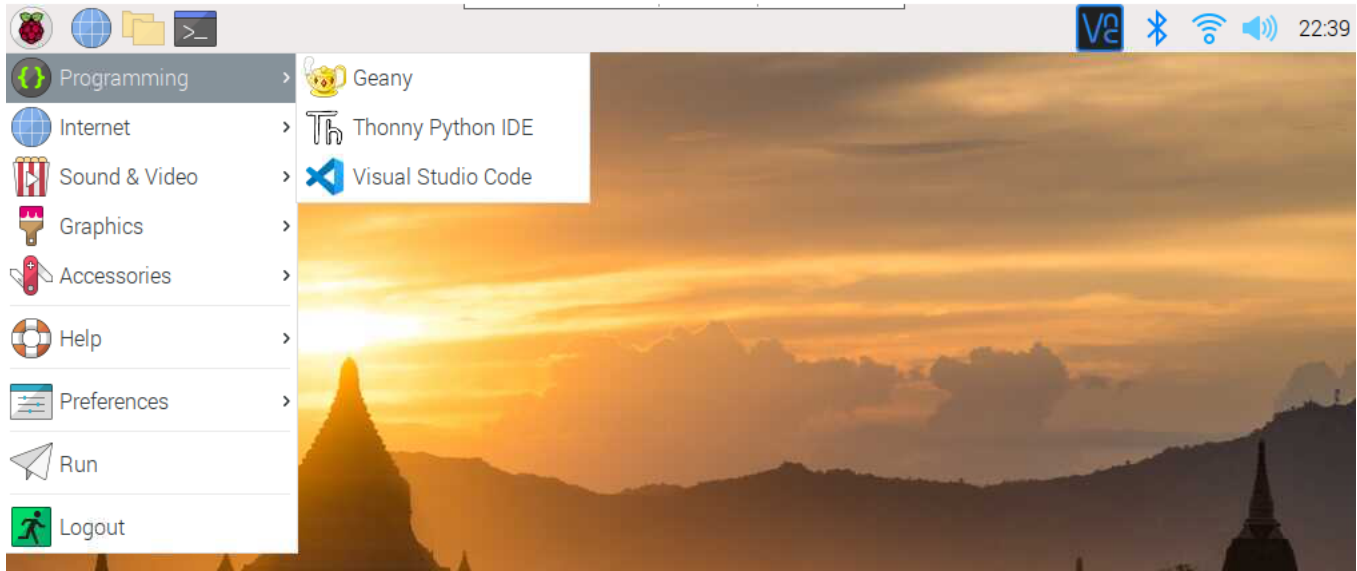
Como preferencia personal mi IDE favorito es VS CODE, para algunos proyectos como Apps en Flask, Bots y Tensorflow, me siento mas cómodo trabajando en la interfaz de VS. La gran noticia es que VS Code ahora está disponible como parte de los paquetes apt de Raspberry Pi OS. Inicie la terminal Raspberry Pi y ejecute los siguientes comandos:

```
sudo apt update
```

```
sudo apt full-upgrade
```

```
sudo apt install code -y
```

Tardara un poco la instalación, después de terminar, puedes acceder por VNC y revisar que la instalación fue exitosa.



Listo con esto, podemos empezar a trabajar con VS CODE y Git, trata de utilizar lo aprendió en esta guía, para crear un segundo repositorio con VS.

También puedes revisar el repositorio de este post en mi Github.