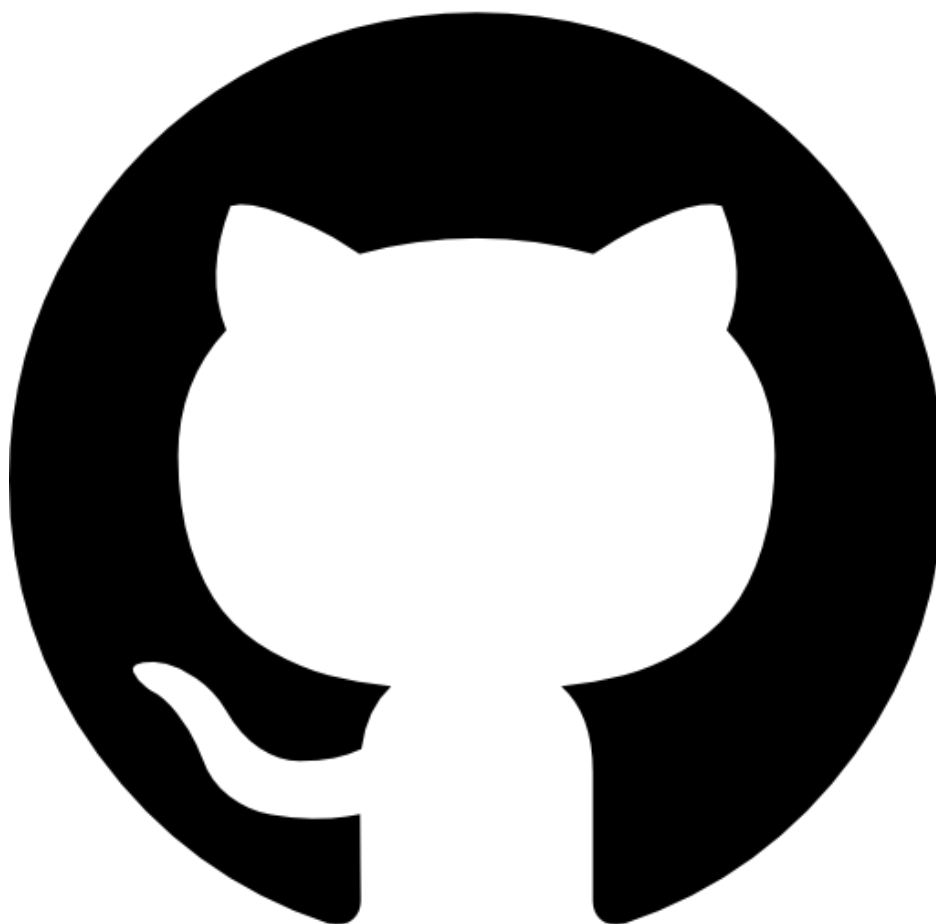


Diego Salazar.	15/09/2024.
Guía.	ASIR2.

GUÍA DE INSTALACIÓN DE GITHUB DESKTOP, VSC Y ATAJO PARA EL TERMINAL.



ÍNDICE.

ATAJO.....	3
INSTALACIÓN DE GITHUB.....	5
INSTALACIÓN DE VSC	8
WEBGRAFÍA.....	11

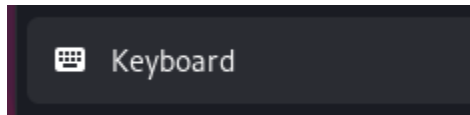
Hola, como epílogo menciono que les enseñe aquí como instalar y usar las herramientas mencionadas en el título, todo esto usando kali linux.

Primero como vamos a usar muchísimo el terminal lo mejor será crear un atajo para entrar rápido, dicho atajo existe en ubuntu pero no en kali.

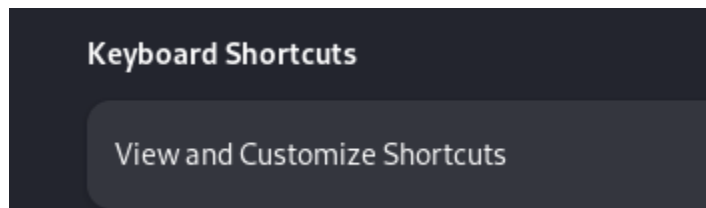
ATAJO

Paso a paso:

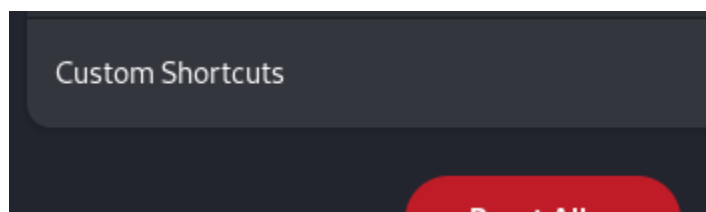
Vamos a los “Display Setting” o “ajustes”, “teclado” o “keyboard”



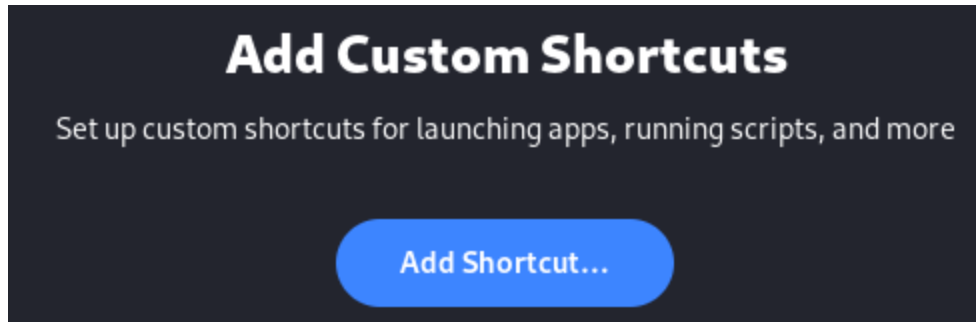
Al fondo estará esta opción, la pinchamos.



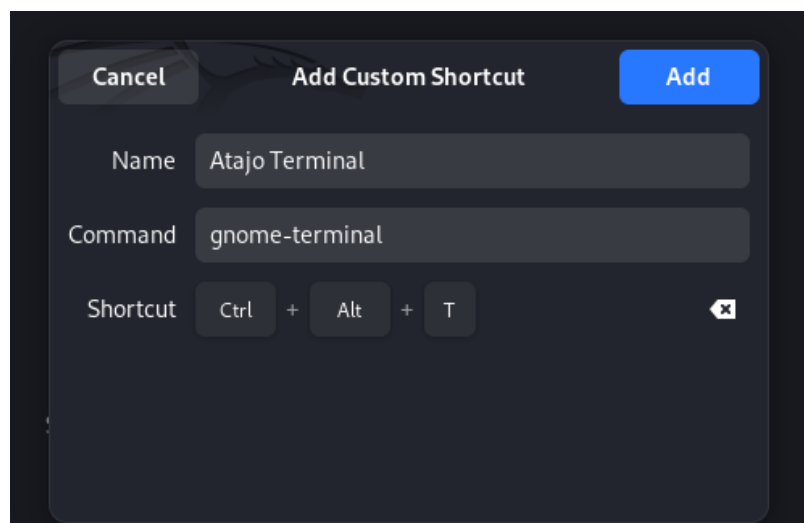
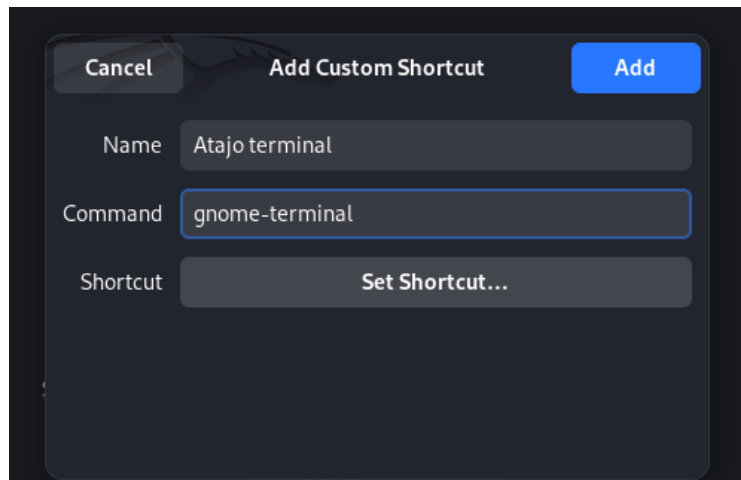
Se abrirá un menú y también al fondo estará la opción que hay que pinchar:

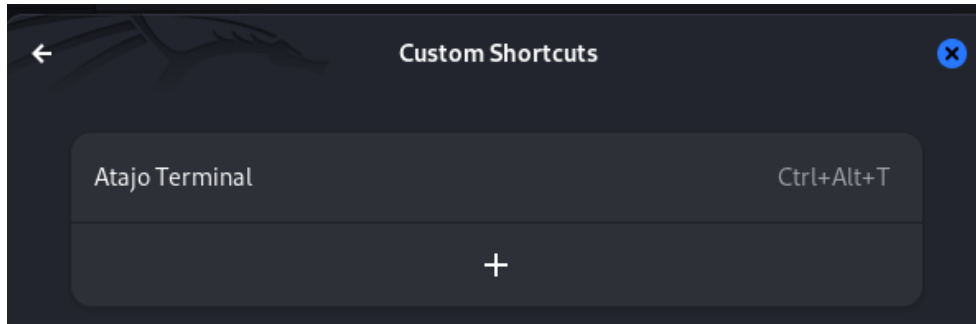


Agregamos



“gnome-terminal” es el comando necesario para que funcione el atajo, luego clic en “set shortcut” y elegimos las teclas que creamos correctas para este atajo, no obstante por comodidad y memoria muscular ya que he usado ubuntu mucho tiempo, usé el clásico **“control + alt + T”**





Y listo, atajo agregado.

INSTALACIÓN DE GITHUB

El **primer paso** es agregar la clave gpg:

```
wget -qO - https://apt.packages.shiftkey.dev/gpg.key | gpg --dearmor | sudo tee  
/usr/share/keyrings/shiftkey-packages.gpg > /dev/null
```

```
(asir@asir)-[~]  
$ sudo su  
[sudo] password for asir:  
(root@asir)-[/home/asir]  
# wget -qO - https://apt.packages.shiftkey.dev/gpg.key | gpg --dearmor | sudo tee /usr/share/keyrings/shiftkey-packages.gpg > /dev/null
```

El **segundo paso** es agregar el repositorio de **GITHUB DESKTOP**:

```
sudo sh -c 'echo "deb [arch=amd64 signed-by=/usr/share/keyrings/shiftkey-packages.gpg]  
https://apt.packages.shiftkey.dev/ubuntu/ any main" > /etc/apt/sources.list.d/shiftkey-  
packages.list'
```

```
(root@asir)-[/home/asir]  
# sudo sh -c 'echo "deb [arch=amd64 signed-by=/usr/share/keyrings/shiftkey-packages.gpg] https://apt.packages.shiftkey.dev/ubuntu/ any main" > /etc/apt/sour  
ces.list.d/shiftkey-packages.list'
```

Nos podemos dar cuenta de que esta instalación es para Ubuntu, sin embargo como kali y ubuntu están basadas en Debian, nos funcionará igual, sin embargo esto no asegura 100% la compatibilidad con todos los paquetes/repositorios de otras desktop apps, esta en específico sí que es compatible.

Por último actualizamos repositorios e instalamos.

```
Sudo apt update && sudo apt install github-desktop
```

```
(root@asir)-[/home/asir]  
# sudo apt update && sudo apt install github-desktop
```

Así lo iniciamos:

```
(asir@asir)-[~]  
$ github
```

Sí no abre la app, lo mejor es que lo ejecutes sin ser root.

Iniciamos sesión

Welcome to GitHub Desktop

GitHub Desktop is a seamless way to contribute to projects on GitHub and GitHub Enterprise. Sign in below to get started with your existing projects.

Sign in to GitHub.com

Sign in to GitHub Enterprise

Let's get started!

Add a repository to GitHub Desktop to start collaborating

Filter your repositories



Your repositories

Diego0rrill0/LMSGI

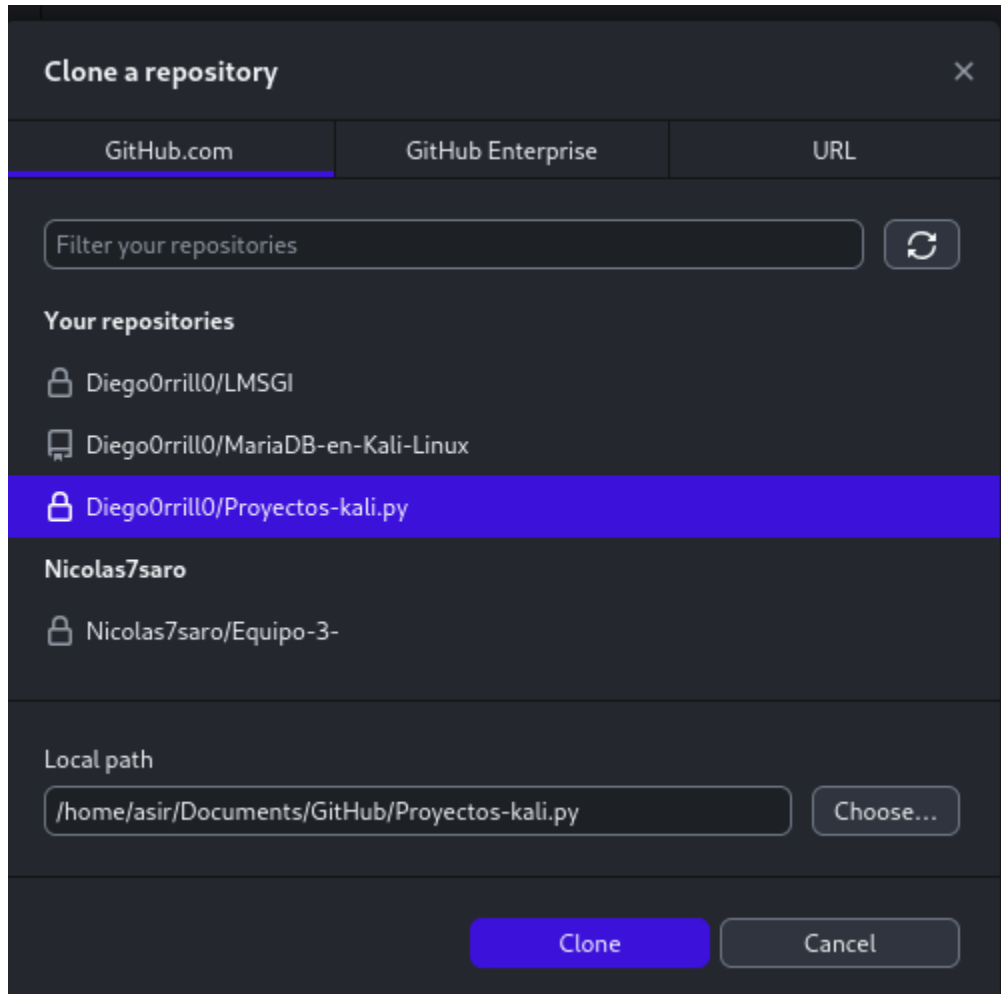
Diego0rrill0/MariaDB-en-Kali-Linux

Diego0rrill0/Proyectos-kali.py

Ahora como necesito clonar un repositorio de paso les enseñaré:

Presionamos Control + shift + O y se nos abrirá un menú para clonar repositorios existentes.

Presionamos en el que queremos clonar y “clone”.



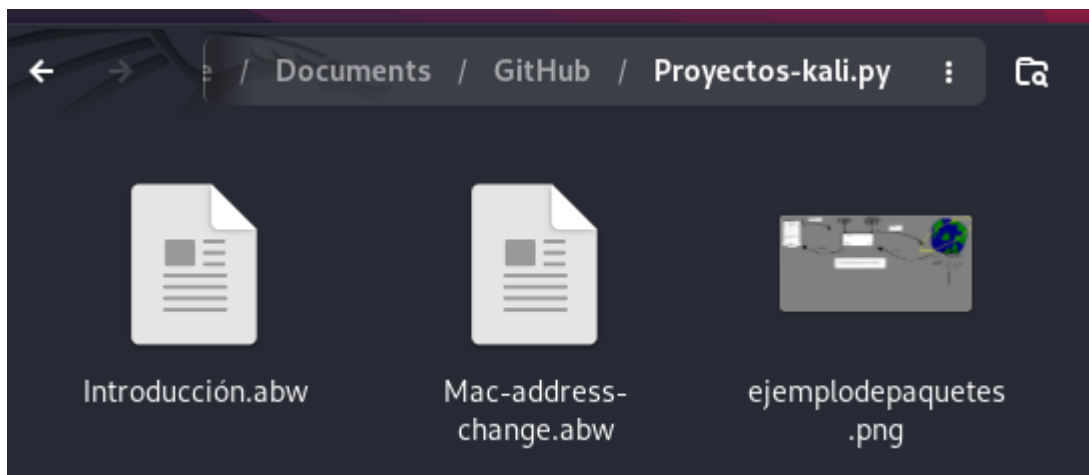
Y si te preguntas donde se clonó el repositorio pues ahí mismo lo dice carajo presta atención, abajo de “Local path” con lo cual si vamos a esa ruta en una pc ahí estará

```
(asir@asir)-[~/Documents]
$ cd GitHub

(asir@asir)-[~/Documents/GitHub]
$ ls
MariaDB-en-Kali-Linux  Proyectos-kali.py

(asir@asir)-[~/Documents/GitHub]
$
```

O de manera gráfica



INSTALACIÓN DE VSC

Descargamos la clave pública:

```
wget -qO- https://packages.microsoft.com/keys/microsoft.asc | gpg --dearmor > packages.microsoft.gpg
```

```
(asir@asir)-[~]
$ wget -qO- https://packages.microsoft.com/keys/microsoft.asc | gpg --dearmor > packages.microsoft.gpg
```

Instalación de la clave y nos aseguramos de darnos permisos de escritura y lectura:

```
sudo install -o root -g root -m 644 packages.microsoft.gpg /usr/share/keyrings/
```



```
(asir@asir)-[~]  
$ sudo install -o root -g root -m 644 packages.microsoft.gpg /usr/share/keyrings/  
[sudo] password for asir:
```

Agregamos el repo de vsc:

```
sudo sh -c 'echo "deb [arch=amd64 signed-  
by=/usr/share/keyrings/packages.microsoft.gpg]   
https://packages.microsoft.com/repos/vscode stable main" >  
/etc/apt/sources.list.d/vscode.list'
```

```
(asir@asir)-[~]  
$ sudo sh -c 'echo "deb [arch=amd64 signed-by=/usr/share/keyrings/packages.microsoft.gpg] https://packages  
.microsoft.com/repos/vscode stable main" > /etc/apt/sources.list.d/vscode.list'
```

Borramos packages.microsoft.gpg ya que sólo era un archivo temporal

```
(asir@asir)-[~]  
$ rm -f packages.microsoft.gpg
```

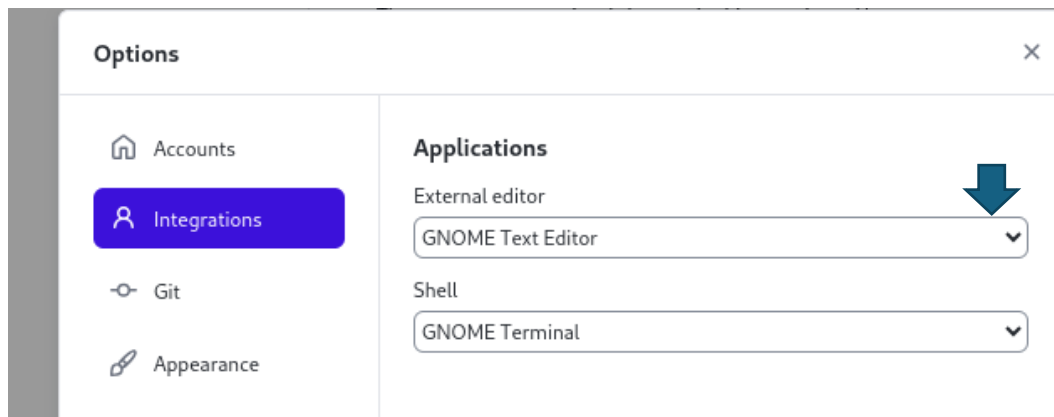
Instalación de la app

Actualizamos con “sudo apt update && sudo apt upgrade -y” y ejecutamos un simple **“sudo apt code -y”** (-y al final del install significa “yes” con lo cual no nos detendremos en algún momento de la descarga para hacerlo manualmente, podemos usarlo para ahorrarnos tiempo por ejemplo cuando estamos actualizando repositorios por primera vez en una máquina nueva podemos ejecutar un **“sudo apt update && sudo apt upgrade -y”**).

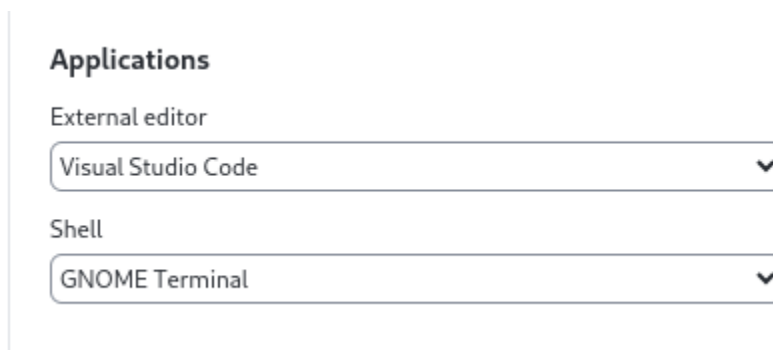
```
(asir@asir)-[~]  
$ sudo apt install code -y  
The following packages were automatically installed and are no longer required:  
  g++-13  g++-13-x86-64-linux-gnu  libstdc++-13-dev  
Use 'sudo apt autoremove' to remove them.  
  
Installing:  
  code
```

Sincronizamos VSC con GitHub Desktop

Sólo hace falta darle a “options”



Hacemos click en la flecha y cambiamos a vscode.



“Save” y listo.

WEBGRAFÍA

<https://code.visualstudio.com/docs/setup/linux>

<https://gist.github.com/berkorbay/6feda478a00b0432d13f1fc0a50467f1>