



# Github

**Configuración de la clave ssh en github**



# Conceptos básicos de SSH

*¿Qué es SSH?*

**Secure Shell** es un protocolo de conexión encriptado que permite operar de forma segura sobre insegura.

Nos permite conectarnos con otros sistemas, ejecutar comandos, descargar y subir archivos de forma segura.



# Conceptos básicos de SSH

## *Clave pública y clave privada*

Para asegurar la comunicación, ssh ocupa 2 claves (estas claves están en archivos):



**Clave pública**

*La podemos compartir*



**Clave privada**

*No la debemos compartir*

## Verificando la instalación

*¿Tenemos los archivos con las claves?*

```
ls -la ~/.ssh
```

Las claves ssh usualmente se guardan en la carpeta .ssh dentro de la carpeta home del usuario.

cd ~ te lleva automáticamente a esa carpeta

ls -la ~/.ssh te muestra todos los archivos de la carpeta .ssh dentro de tu home.

¿Y si la carpeta no existe o está vacía? Revisemos la siguiente diapositiva.

# Verificando la instalación

*3 opciones*

```
ls -la ~/.ssh
```

1. **Existe la carpeta .ssh** y hay dos archivos llamados id\_algo\_mas.pub y otro id\_algo\_mas -> Significa que estamos **ok**.
2. **No existe la carpeta .ssh**: Entonces tenemos que crearla con `mkdir ~/.ssh`
3. **Existe la carpeta .ssh y está vacía**

<< Si estamos en el caso 2, creamos la carpeta y luego estaríamos en el paso 3 >>

## Generando la clave ssh

*Dentro de la carpeta .ssh escribe:*

```
ssh-keygen -t rsa -b 4096 -C "tucorreo@email.com"
```

Nos pedirá que decidamos la carpeta donde dejar las llaves, presionaremos **Enter** para mantener la sugerencia.

Luego, nos pedirá que ingresemos una frase de contraseña, en nuestro caso solo presionaremos **Enter** para dejarlo sin contraseña.

Al terminar de ejecutar se creará el archivo id\_rsa.pub (clave pública) e id\_rsa (clave privada).

# Levantando el agente SSH

*En el terminal escribe:*

```
eval "$(ssh-agent -s)"
```

El agente ssh es el encargado de registrar las claves ssh y sus passphrases, debemos tenerlo activo para que funcione la conexión ssh. **Esto se hace una sola vez.**

Si todo funciona bien obtendremos una respuesta como esta: Agent pid 59566

# Agregando la clave generada al agente SSH

*En el terminal escribe:*

En Windows o Linux:

```
ssh-add ~/.ssh/id_rsa
```

En OSX

```
ssh-add -K ~/.ssh/id_rsa
```



# Agregando la clave en github

*Obtener la clave desde el terminal*

## Mac:

```
pbcopy <  
~/.ssh/id_rsa.pub
```

## Linux:

```
sudo apt install xclip  
xclip -sel clip <  
~/.ssh/id_rsa.pub
```

## Windows:

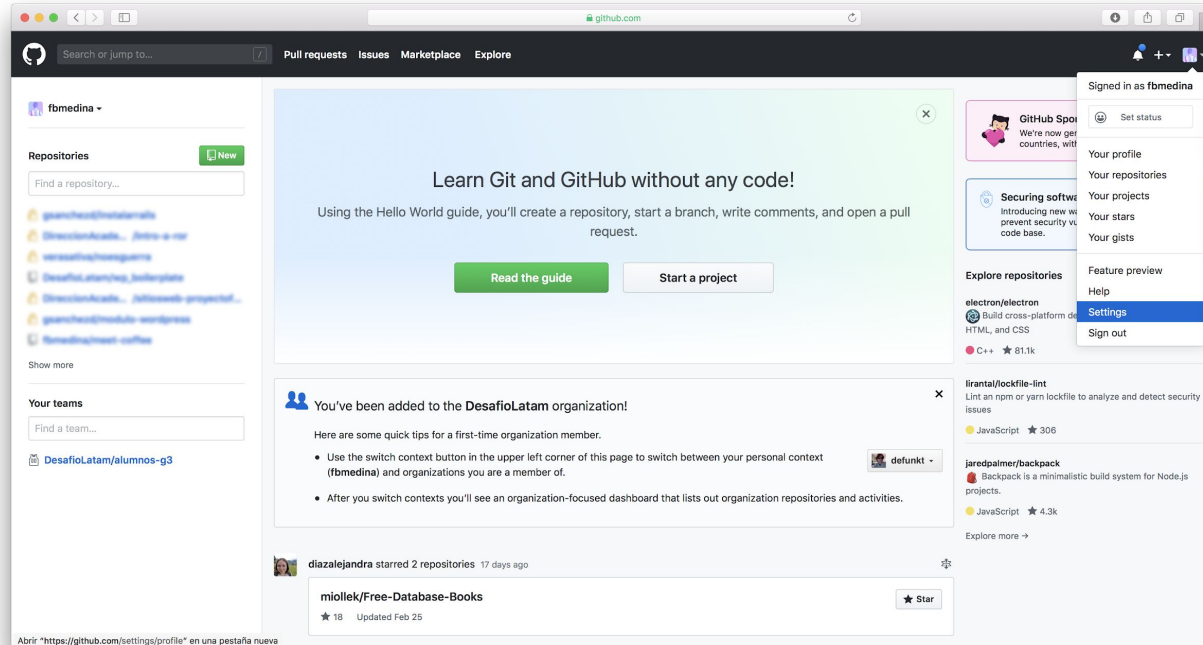
```
clip < ~/.ssh/id_rsa.pub
```

Con esto, la clave quedará copiada en el portapapeles lista para pegarla dentro de la configuración de GitHub.

# Agregando la clave en github

*Buscando la configuración en github*

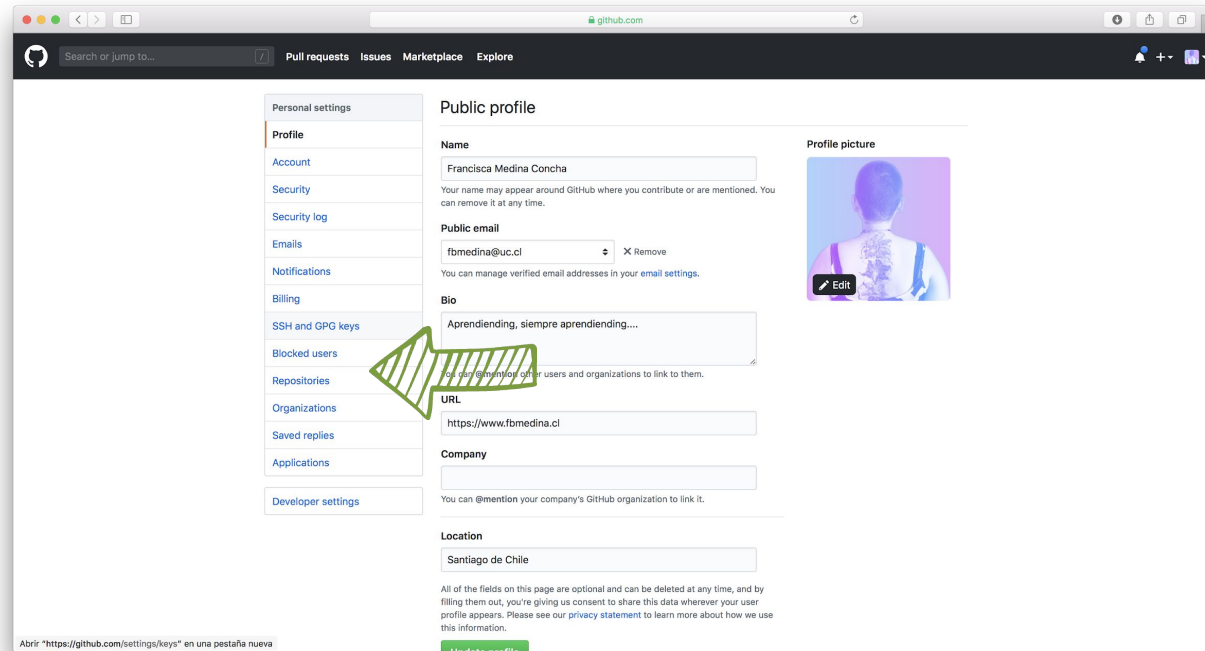
Iniciamos sesión en GitHub, presionamos nuestro avatar y luego elegimos la opción **Settings**.



# Agregando la clave en github

## Buscando la configuración en github

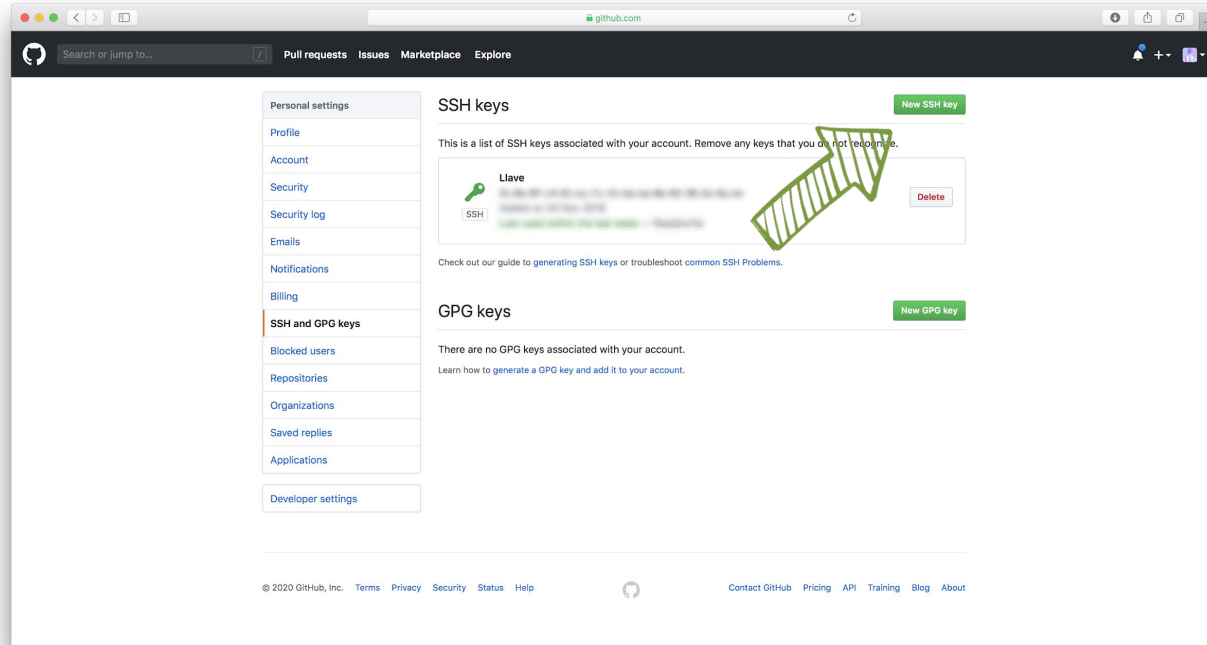
Dentro de la configuración personal de tu cuenta, presiona la opción **GPG and SSH keys**.



# Agregando la clave en github

*Buscando la configuración en github*

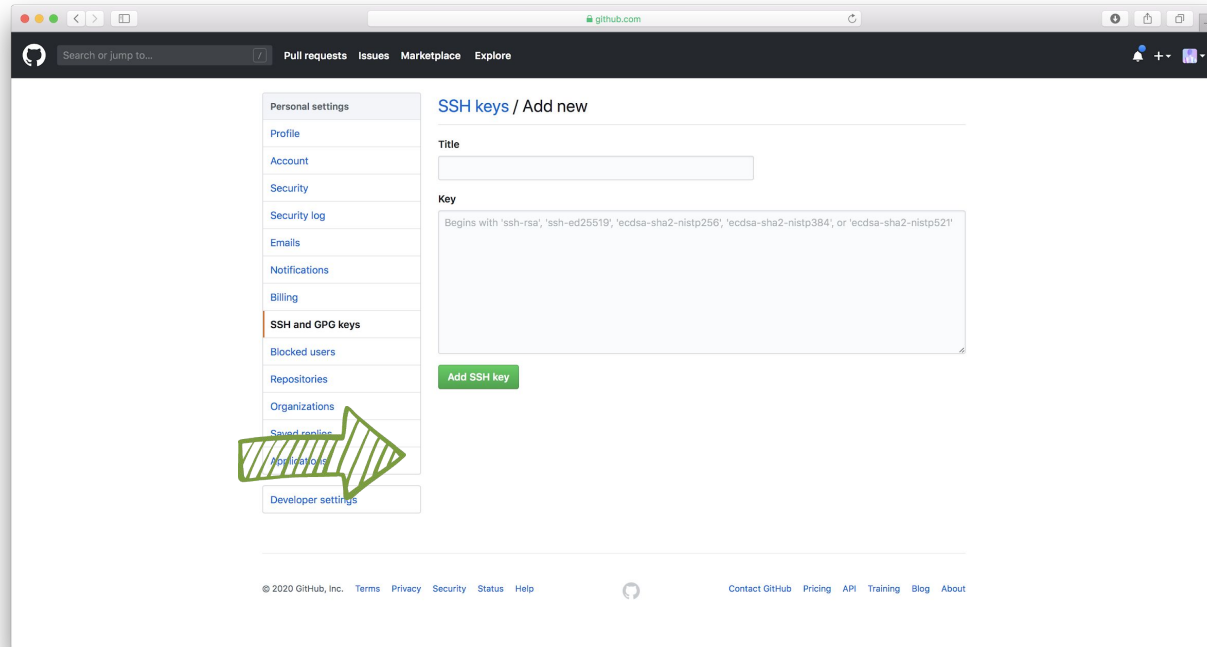
Dentro, presionamos **New SSH Key**.



# Agregando la clave en github

## Buscando la configuración en github

Colocamos un nombre al dispositivo, esto es solo una referencia, ya que podemos agregar el nombre que queramos, pero que nos ayude a recordar en qué dispositivo está la clave; y luego pegamos la clave SSH (copiada anteriormente).



# Agregando la clave en github

*Probando la configuración desde el terminal*

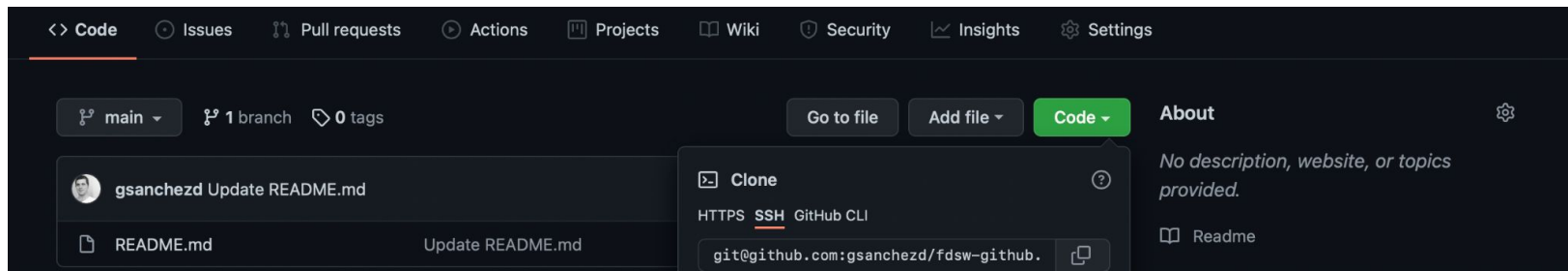
```
ssh -T git@github.com
```

Deberíamos obtener como respuesta el siguiente mensaje: <<Hi username! You've successfully authenticated, but GitHub does not provide shell access>>

# Probando Github

## Probemos descargar en un proyecto

1. Ingresar a <https://github.com/gsanchezd/fdsd-github> y hacer fork del proyecto.
2. Desde el fork, copiar la dirección para descargar el proyecto haciendo clic en code y luego ssh.



3. Escribir en el terminal `git clone` <https://github.com/gsanchezd/fdsd-github> proyecto-fdsd para descargar el proyecto.

# Probando Github

## *Probemos subir cambios*

4. Realizar un cambio cualquiera que puedas observar fácilmente en el archivo index.html utilizando el editor de código.
5. Utiliza `git add index.html` para agregar los cambios.
6. Utiliza `git commit -m "texto descriptivo"` para confirmar los cambios.
7. Utiliza `git push origin main` para subir los cambios.
8. Revisa en la página de github de tu fork si los cambios fueron subidos.
9. Felicidades, lo lograste.