

**Instituto Emiliani somascos**  
**Análisis de Sistemas**  
**Ciclo Escolar 2023**

# **MANUAL DE INSTALACION**

**Daniel Sebastián Hernández Valle**  
**5TO Computación “A”**  
**Clave: 33**

# ÍNDICE

<b>INTRODUCCIÓN .....</b>	<b>3</b>
<b>PROCESO DE INSTALACION.....</b>	<b>4</b>
<b>PROCESO DE CONFIGURACIÓN DE GIT HUB.....</b>	<b>7</b>
<b>RECOMENDACIONES .....</b>	<b>11</b>



## INTRODUCCIÓN

GitHub es un servicio de la nube que permite a los desarrolladores y desarrolladoras almacenar y administrar su código, así como a llevar el registro y control de cualquier cambio que se realice sobre este código.

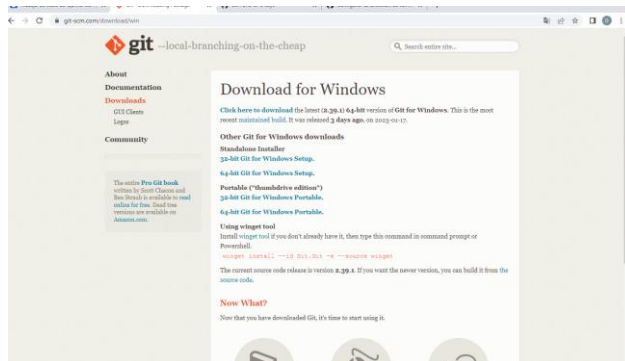
Una de las ventajas que ofrece esta plataforma es la colaboración que existe entre los usuarios, ya que pueden modificar el código para mejorarlo o agregar nuevas funciones.

A continuación, daremos una explicación de como instalar y configurar Git Hub en nuestro PC y así ya lo podremos utilizarlo para nuestras necesidades


# PROCESO DE INSTALACION

1. DESCARGAR Git Hub en la página oficial

<https://git-scm.com/download/win>



2. Luego de descárgalo ejecutamos el launcher

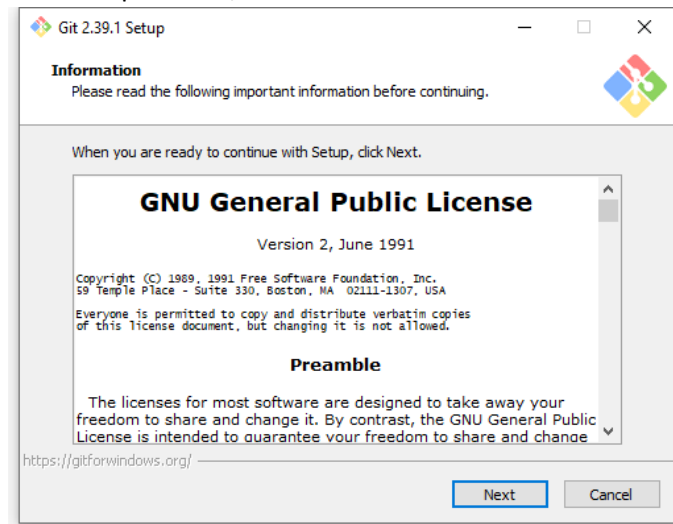
 Git-2.39.1-64-bit

20/01/2023 15:29

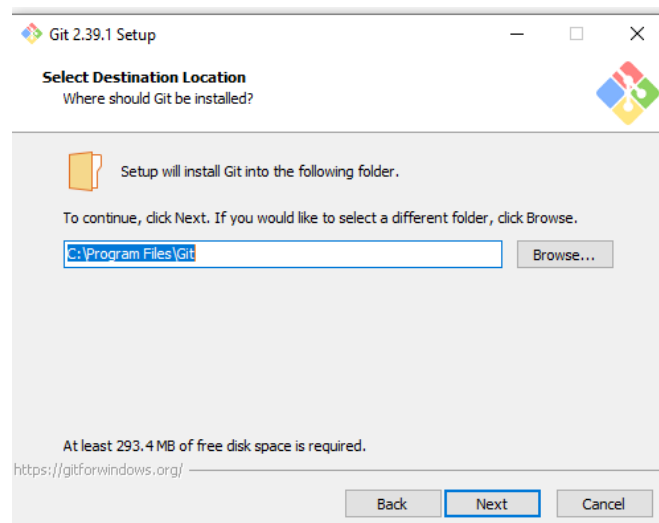
Aplicación

51.757 KB

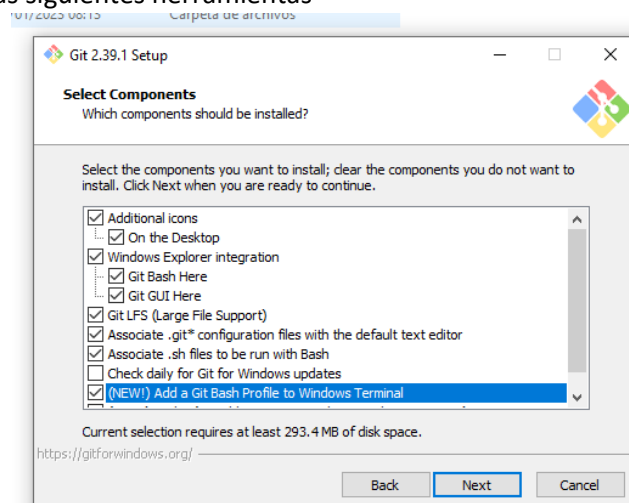
3. Nos saldrá la licencia de producto, daremos clic en NEXT



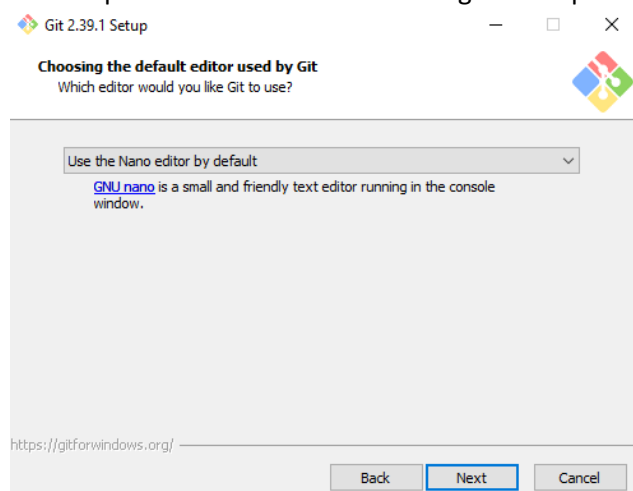
4. Nos indicara la carpeta de instalación, la seleccionamos y le damos a NEXT



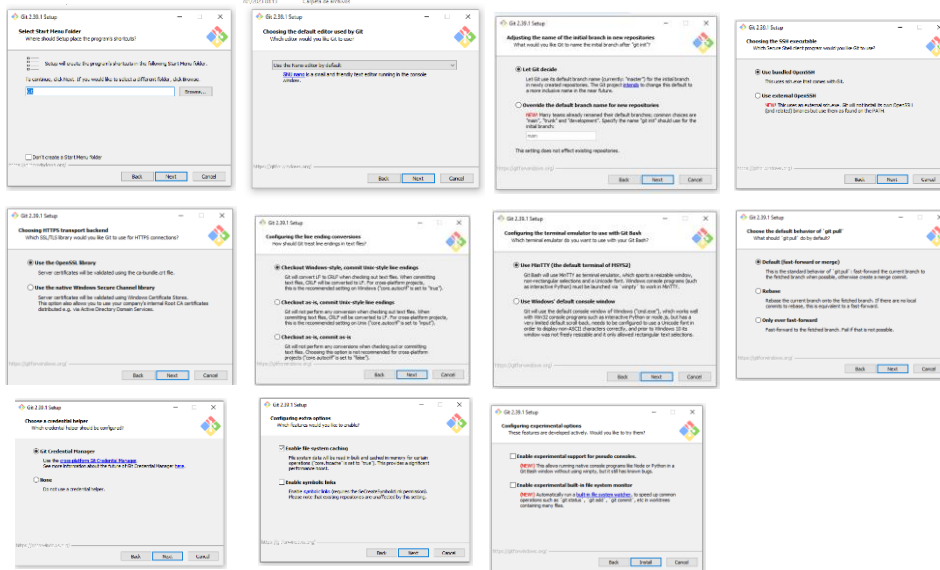
5. Chequeamos las siguientes herramientas



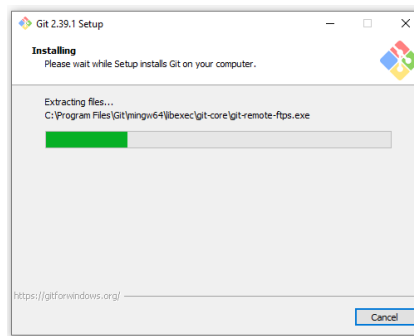
6. En esta pantalla seleccionaremos la siguiente opción



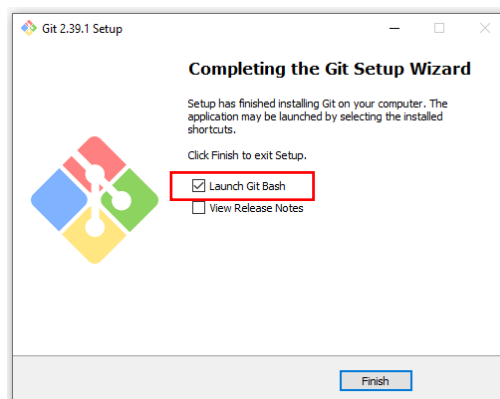
## 7. En las siguientes pantallas dejaremos todo por defecto



## 8. Luego de saltar todas las pantallas anteriores esperaremos a que termine la instalación

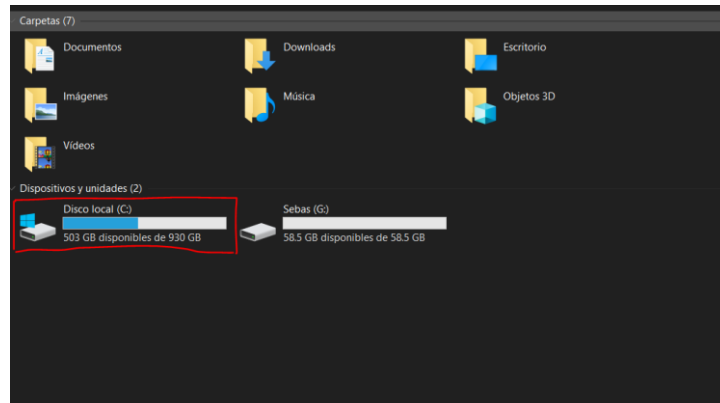


## 9. Para finalizar la instalación desmarcaremos el check de abajo y marcaremos el de arriba y daremos a finalizar la instalación



# Proceso de configuración de Git Hub

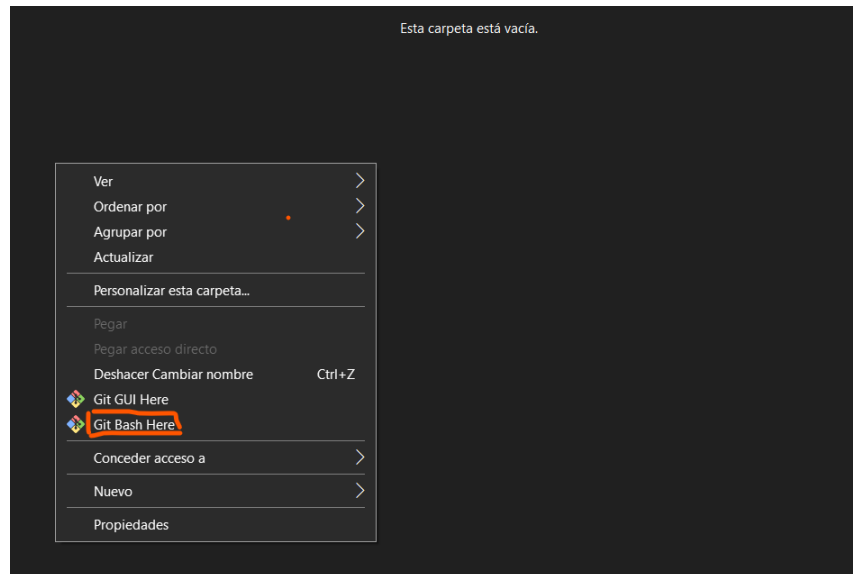
## 1. Nos dirigimos a nuestro disco local, entramos a la carpeta usuario, donde crearemos una carpeta para la configuración



Nombre	Fecha de modificación	Tipo	Tamaño
Archivos de programa	20/01/2023 21:35	Carpeta de archivos	
Archivos de programa (x86)	3/01/2023 19:20	Carpeta de archivos	
DB2	10/06/2022 20:21	Carpeta de archivos	
Games	14/03/2022 18:43	Carpeta de archivos	
Intel	14/01/2023 18:51	Carpeta de archivos	
PerfLogs	29/07/2022 17:31	Carpeta de archivos	
Usuarios	5/09/2022 14:07	Carpeta de archivos	
Windows	14/01/2023 21:04	Carpeta de archivos	

Nombre	Fecha de modificación	Tipo	Tamaño
.dnx	16/03/2022 16:00	Carpeta de archivos	
.nuget	20/05/2022 15:23	Carpeta de archivos	
.Origin	24/11/2021 14:07	Carpeta de archivos	
.QtWebEngineProcess	24/11/2021 14:07	Carpeta de archivos	
ansel	24/11/2021 14:43	Carpeta de archivos	
Búsquedas	18/06/2022 22:35	Carpeta de archivos	
Configuración	20/01/2023 21:40	Carpeta de archivos	
Contactos	18/06/2022 22:35	Carpeta de archivos	
Documentos	14/01/2023 14:25	Carpeta de archivos	
Downloads	20/01/2023 21:17	Carpeta de archivos	
Escritorio	20/01/2023 19:54	Carpeta de archivos	
Favoritos	18/06/2022 22:35	Carpeta de archivos	
Imágenes	3/01/2023 16:46	Carpeta de archivos	
Juegos guardados	18/06/2022 22:35	Carpeta de archivos	
Música	18/06/2022 22:35	Carpeta de archivos	
Objetos 3D	18/06/2022 22:35	Carpeta de archivos	
OneDrive	14/07/2022 22:56	Carpeta de archivos	
pseint	2/03/2022 09:03	Carpeta de archivos	
TuneFab	22/05/2022 08:21	Carpeta de archivos	
Videos	20/01/2023 20:09	Carpeta de archivos	
Vínculos	18/06/2022 22:35	Carpeta de archivos	
estado-bateria	18/04/2022 09:47	Chrome HTML Do...	46 KB

**2. Luego de crear la carpeta de configuración, damos click derecho adentro de la carpeta, y le damos a la opción de Git Hub Bash**



3. Entramos y en la pestaña que se nos presenta, escribimos el siguiente comando para configurar nuestro nombre de usuario

**git config --global user.name "nombre"**

```
Usuario@DESKTOP-6LHA8DN MINGW64 ~/Configuracion
$ git config --global user.name "Sebss10"

Usuario@DESKTOP-6LHA8DN MINGW64 ~/Configuracion
$ |
```

4. Hacemos lo mismo, pero con la diferencia que ahora enlazaremos nuestro correo asociado a GIT HUB a nuestro pc, por medio de este comando

**git config --global user.email "YOUR\_EMAIL"**

```
Usuario@DESKTOP-6LHA8DN MINGW64 ~/Configuracion
$ git config --global user.email "danielhernandez@emilianisomascos.edu.gt"
```

5. Hacemos el mismo procedimiento pero con el comando para verificar si existe alguna clave SSH en el PC, el cual, el comando es

**ls -al ~/.ssh**

```
Usuario@DESKTOP-6LHA8DN MINGW64 ~/Configuracion
$ ls -al ~/.ssh
ls: cannot access '/c/Users/sebas/.ssh': No such file or directory
```



6. Por ultimo generamos una nueva clave de SSH PARA terminar la configuración  
Ponemos este comando para enlazar nuestro correo

```
ssh-keygen -t ed25519 -C "your_email@example.com"
```

```
Usuario@DESKTOP-6LHA8DN MINGW64 ~/Configuracion
$ ssh-keygen -t ed25519 -C "danielhernandez@emilianisomascos.edu.gt"
```

7. Luego de ingresar ese código, nos generara un archivo, que le pondremos un nombre de creación

```
Usuario@DESKTOP-6LHA8DN MINGW64 ~/Configuracion
$ ssh-keygen -t ed25519 -C "danielhernandez@emilianisomascos.edu.gt"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/c/Users/sebas/.ssh/id_ed25519): Prueba1
```

8. Luego de crear nuestro archivo, ponemos una clave de verificación y será valido nuestro archivo

```
Your public key has been saved in Prueba1.pub
The key fingerprint is:
SHA256:aCi5SCYXwQrTC6En/v8NYLSHjNzUJ0hwT4hk22uJ8Dc danielhernandez@emilianisomascos.edu.gt
The key's randomart image is:
+--[ED25519 256]--+
|..+o+oo..|
|= +o+.=|
|+=oo = + |
|oo=0B.=.o|
|.++=.Eo.S|
|+o.o+..+|
|.. .. .|
|. . . .|
|... ..|
+-----[SHA256]-----+
```

9. Ahora cerramos nuestro Git Hub y vamos a la carpeta de creación, ahí abriremos este archivo por bloc de notas y copiamos el serial que se nos muestra

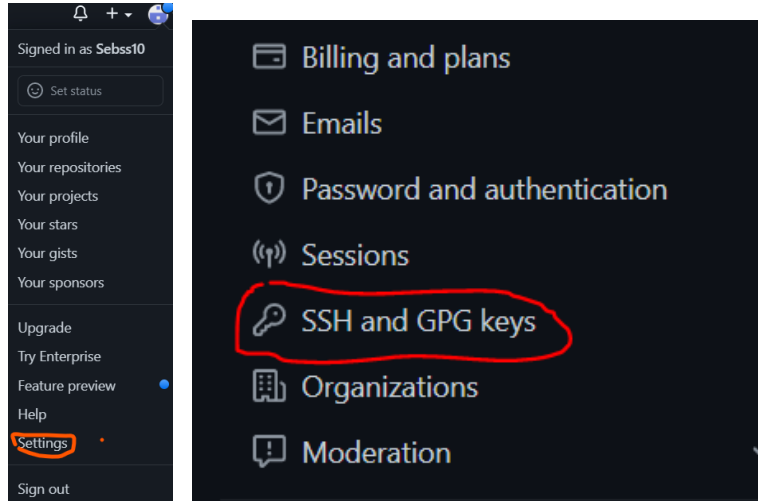
Prueba1	20/01/2023 22:19	Archivo	1 KB
Prueba1	20/01/2023 22:19	Microsoft Publishe...	1 KB

Prueba1: Bloc de notas

Archivo Edición Formato Ver Ayuda

```
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAILy5qeQNDcfTaIRt9FuUyx9hm6Mp1uxuvOCzfV+szTR danielhernandez@em
```

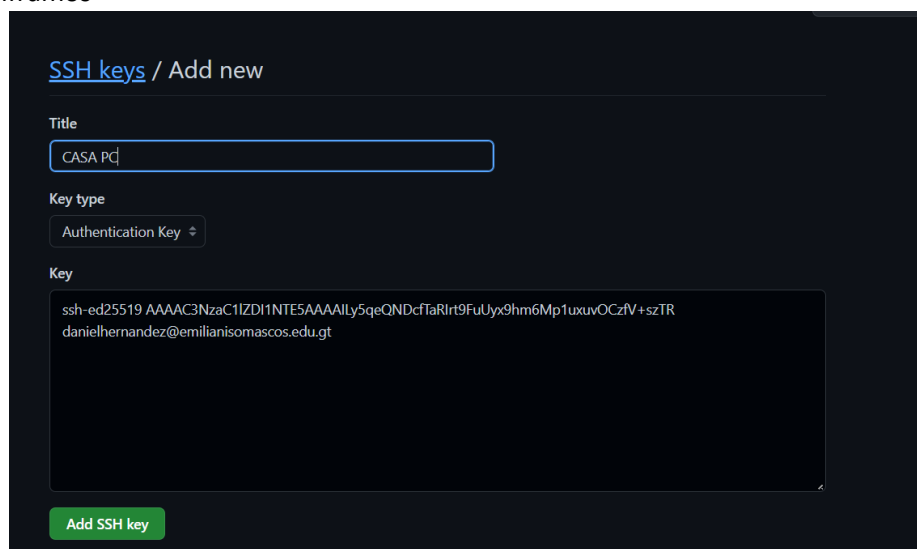
10. Ahora nos vamos a nuestra cuenta de Git Hub, y seleccionamos las siguientes opciones



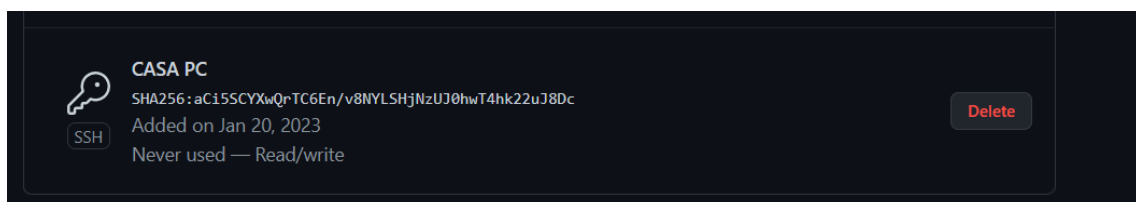
11. Ahora, daremos click en esta opción, para enlazar nuestro archivo



12. Por ultimo. Le ponemos un nombre y copiamos y pegamos la clave que anteriormente desciframos



13. Y con esto ya tendríamos configurado totalmete nuestro entorno de Git Hub



# RECOMENDACIONES

- Tener creada una cuenta de Git Hub para poder configurarlo
- Saber configurar Git Hub en tu pc para tener un rendimiento excelente
- Actualiza el código constantemente
- Revisa los cambios antes de agregarlos a staging