

# **DOCUMENTACIÓN CONTROL DE VERSIONES**

## **Índice**

Introducción .....	2
Proceso para hacer los comandos desde Visual Studio .....	3
Registro de commits.....	5

## Introducción

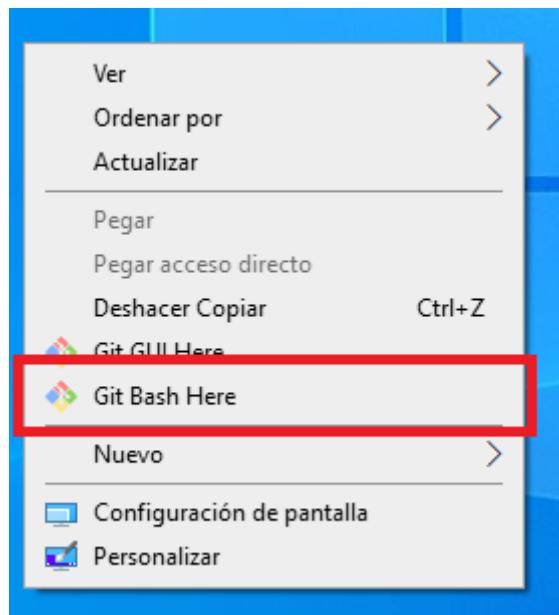
Para el control de versiones se ha creado en GitHub un repositorio nuevo.

Para acceder al repositorio remoto se debe usar la siguiente URL:

<https://github.com/Victorsg1999/protecniproject>

Para poder usar todos los comandos para el sistema de control de versiones cabe recalcar que para hacer todos los comandos a partir de ahora tenemos que tener instalado en nuestro sistema operativo el sistema de control de versiones denominado GIT <https://git-scm.com/>

Para poder ver el proyecto remoto a local procederemos a su descarga para ello se tendrá que abrir una terminal de GIT, normalmente en Windows una vez instalado el GIT, click derecho y hacemos click en Git Bash Here.



Acto seguido: git clone <https://github.com/Victorsg1999/protecniproject>

A screenshot of a terminal window titled 'MINGW64:/c/Users/Usuario/Desktop'. The window shows the command '\$ git clone https://github.com/Victorsg1999/protecniproject' being typed. The terminal is black with white text, and the window has standard window controls at the top.

Yo personalmente he creado una vez clonado el repositorio el archivo de configuración .gitignore, para evitar subir archivos irrelevantes hasta la finalización del proyecto, para tener una mejor visión en el control de versiones.

Para poder usar los comandos de una forma mas liviana, en vez de usar la terminal Git Bash Here y tener que posicionarme en la carpeta del proyecto para aplicar cambios, he optado por realizar todos los cambios desde el id Visual Studio Code, ya que incorpora muy bien las funcionalidades básicas

para el control de versiones y pose una terminal que se puede abrir presionando control + n para realizar comandos manualmente.

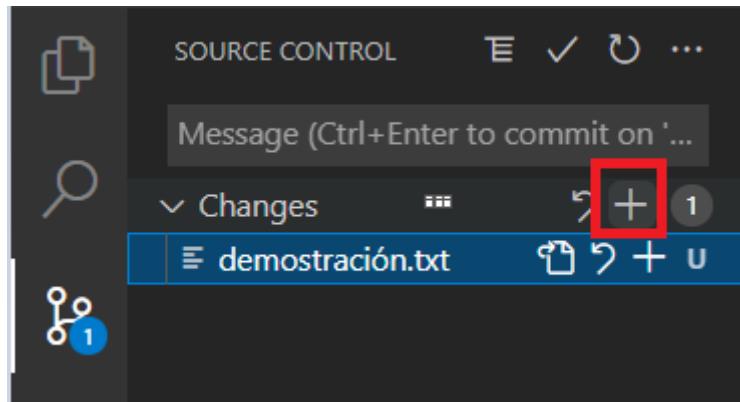
Ahora citare los comandos que nos facilita el Id Visual Studio Code para hacerlo de forma gráfica.

- **git add .** → Nos permite guardar todos los archivos en estado preparación, para poder realizar una subida posteriormente.
- **git commit -m ‘Mensaje del commit’** → Nos permite crear una breve descripción sobre los cambios que hemos hecho y actualizar el estado para una posterior subida.
- **git push** → Nos permite subir todos los cambios a la rama remota.

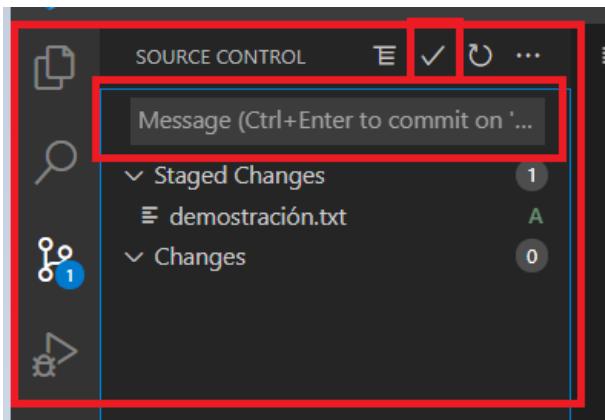
## Proceso para hacer los comandos desde Visual Studio

Para poder trabajar con los comandos, una vez iniciado el proyecto o cuando hagamos el primer commit nos pedirá nuestro usuario y contraseña de github y nuestra llave de github, se nos abrirá Chrome y se aceptará.

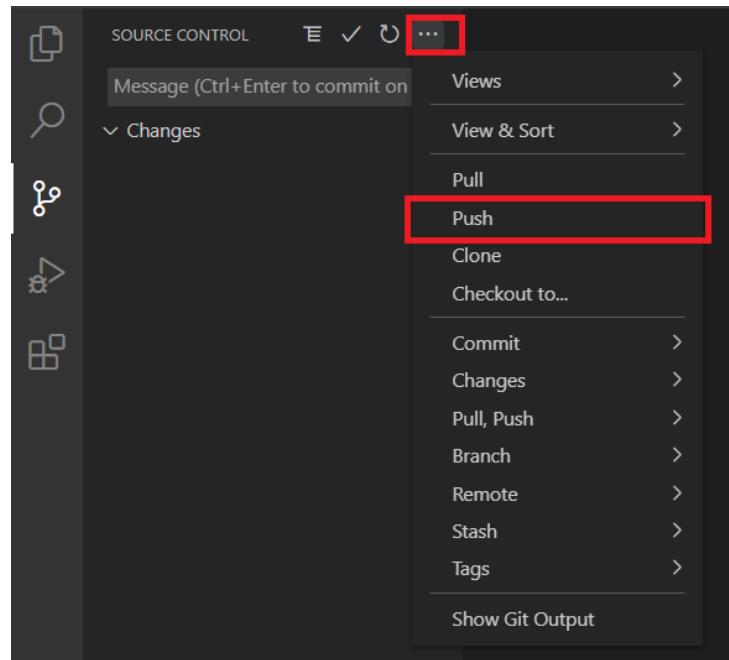
- Para hacer un guardado de datos usaremos git add . en Visual Studio nos ubicaríamos en el tercer ícono en el de la rama, y presionaríamos el +, nos pondría todos los cambios en staggit.



- Se nos han guardado los cambios, para hacer un commit escribiríamos en mensaje la descripción que consideremos oportuna y presionaremos sobre el check de confirmación.



- Por último, para subir los cambios a la rama remota, se hace git push.



Aclaración la demostración anterior es ficticia, lo único que nos quedaría hacer es ir al repositorio remoto <https://github.com/Victorsg1999/protecniproject> y visualizar que nuestro commit se haya procesado correctamente.

Para detallar gráficamente la evolución del proyecto, procederé a abrir un git bash here, y ejecutar el siguiente comando git log --graph.

```
Usuario@DESKTOP-VVI3H39 MINGW64 /c/xampp/htdocs/php/Ejercicios/protecnaproject/
protecnaproject (master)
$ git log --graph
```

## Registro de commits

1. Creo index.html, estilos.css, .gitignore

```
* commit b85cb356021fbb94e361f3bc5bb372418e29eff
| Author: Víctor Sempere Guilabert <v.semgui@gmail.com>
| Date: Tue Mar 16 20:31:50 2021 +0100
|
| Configuración inicial del proyecto
```

2. Añado librerías Boostrap y Font-awesome .

```
* commit cba4035858235d48020b38d979cfe55c98790127
| Author: Víctor Sempere Guilabert <v.semgui@gmail.com>
| Date: Tue Mar 16 20:48:02 2021 +0100
|
| Añado las librerías de bootstrap y font-awesome
```

3. Añadido header y slider Boostrap.

```
* commit 1a0d2f0ba7f87c913f280eef803b8c9c756c3dbc
| Author: Víctor Sempere Guilabert <v.semgui@gmail.com>
| Date: Tue Mar 16 21:19:22 2021 +0100
|
| Añado header y slider
```

4. Plantilla home boceto finalizado.

```
* commit a9edd0b38c961a55c1ebc3c09be20d099981e8b1
| Author: Victor Sempere Guilabert <v.semgui@gmail.com>
| Date: Sat Mar 20 11:04:48 2021 +0100
|
| Home idea finalizado.
```

5. Clase PHP y registro

```
* commit 0066a06e97651ceca7f97b6bfadce4ab79cad1a5
| Author: Víctor Sempere Guilabert <v.semgui@gmail.com>
| Date: Thu Apr 8 16:41:36 2021 +0200
|
| Se ha hecho una primera versión de la clase php
| y su registro.
```

6. Se ha incluido menú, contacto, inicio, y primera versión administrativa. Se ha corregido error librería Font awesome, ahora si pilla la indexada localmente,

```
* commit cfeeabceae02337254199a07864129a68c0fa082
| Author: Víctor Sempere Guilabert <v.semgui@gmail.com>
| Date:   Sun May 2 08:39:52 2021 +0200

| Added clase php. Added contacto. Added inicio.
| Added menú php. Added primera versión sección administrativa creación
| usuarios.
| Fix libreria font-awesome.
```

7. Added animaciones, perfil usuario, paginación php, boceto histórico, más restructuración documento.

```
* commit 356fac8cda2813e7ab8e7080a4a6bfc2fd7b58c0 (HEAD -> master, origin/master)
| Author: Víctor Sempere Guilabert <v.semgui@gmail.com>
| Date:   Wed May 12 01:05:57 2021 +0200

| Creación de animaciones, perfiles de usuario,
| paginación php, boceto historico, restructuración de codigo.
```