

Visual Studio

Despliegue de aplicaciones web (DAW)

2º DESARROLLO DE APLICACIONES WEB
20/02/2021

NEREA ÁLVAREZ JUSTEL

Contenido

INTRODUCCIÓN	2
DESARROLLO DE LOS CONTENIDOS	3
1. instalación de Git portable y conexión con Visual Studio.....	3
2. Manejo de ficheros en Visual Studio	6
3. GitHub	11
CONCLUSIÓN.....	12
BIBLIOGRAFIA	13

INTRODUCCIÓN

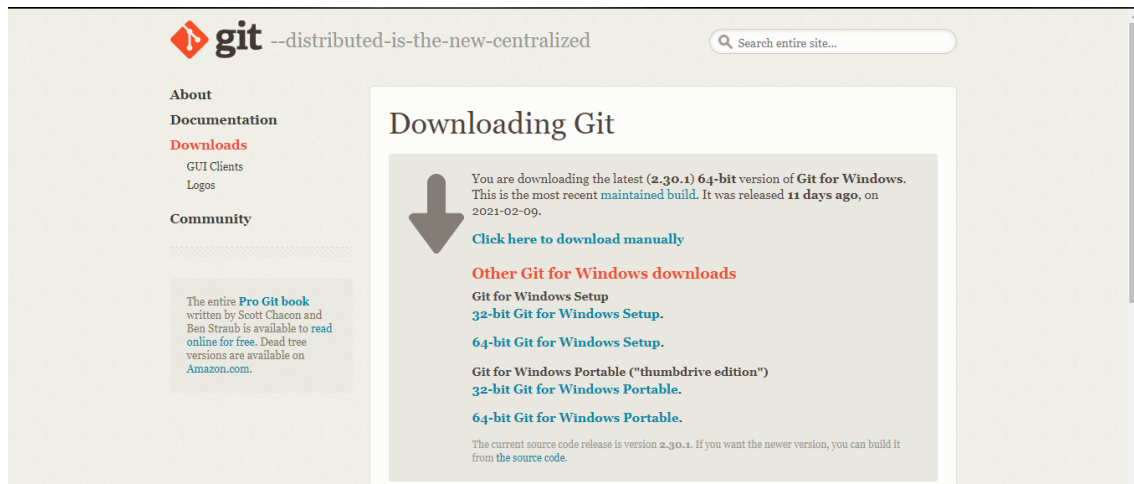
Visual Studio Code es un editor de código fuente desarrollado por Microsoft para Windows, Linux y macOS. Incluye soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código. También es personalizable, por lo que los usuarios pueden cambiar el tema del editor, los atajos de teclado y las preferencias. Es gratuito y de código abierto, aunque la descarga oficial está bajo software privativo e incluye características personalizadas por Microsoft.

Visual Studio Code se basa en Electron, un framework que se utiliza para implementar Chromium y Node.js como aplicaciones para escritorio, que se ejecuta en el motor de diseño Blink. Aunque utiliza el framework Electron, el software no usa Atom y en su lugar emplea el mismo componente editor (Monaco) utilizado en Visual Studio Team Services (anteriormente llamado Visual Studio Online).

DESARROLLO DE LOS CONTENIDOS

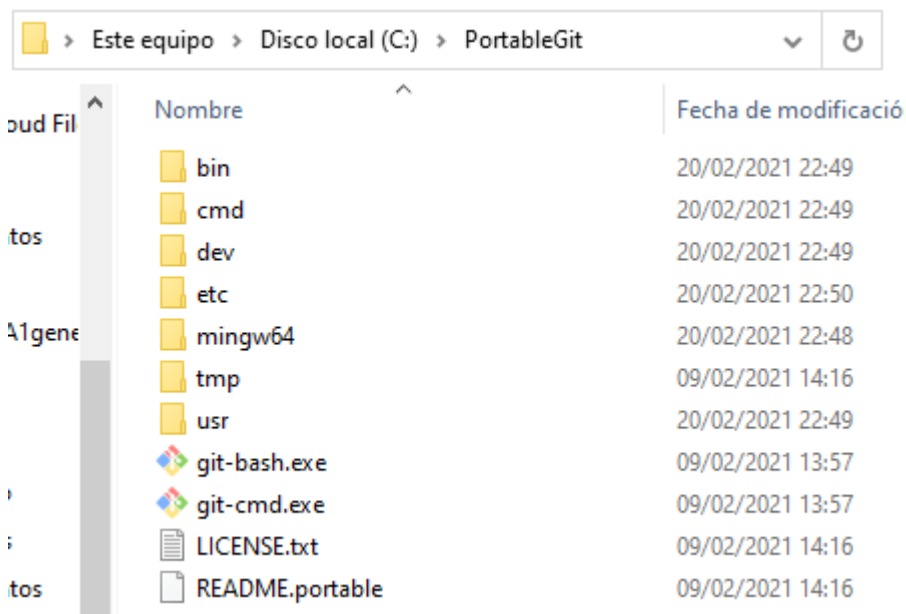
1. instalación de Git portable y conexión con Visual Studio.

Entramos en la página <https://git-scm.com/download/win> y seleccionamos la opción portable para Windows de 64-bits:



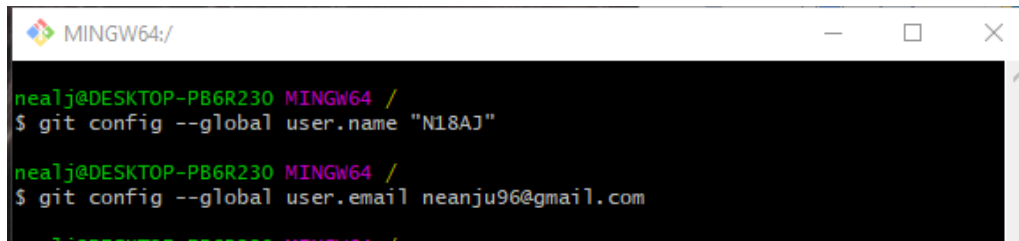
Ejecutaremos el .exe que se descarga, lo colocaremos en la ruta que creamos más optima y esperaremos a que termine.

Se generará una carpeta con todos los ficheros y aplicaciones necesarias del Git.



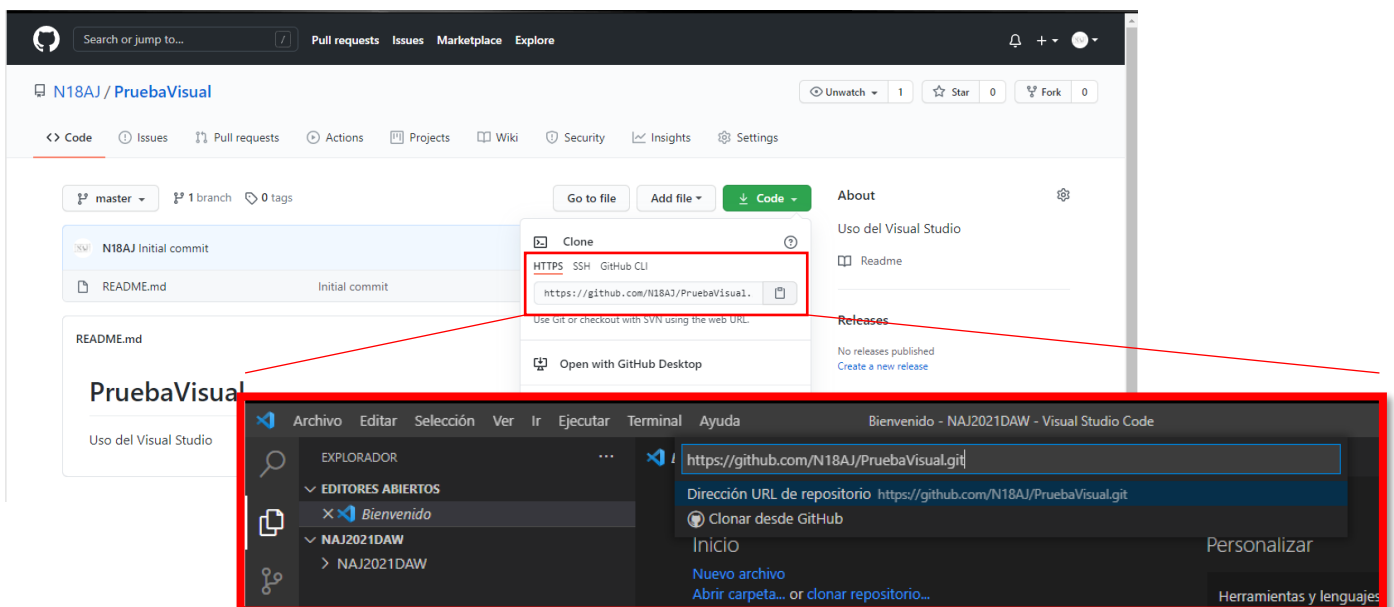
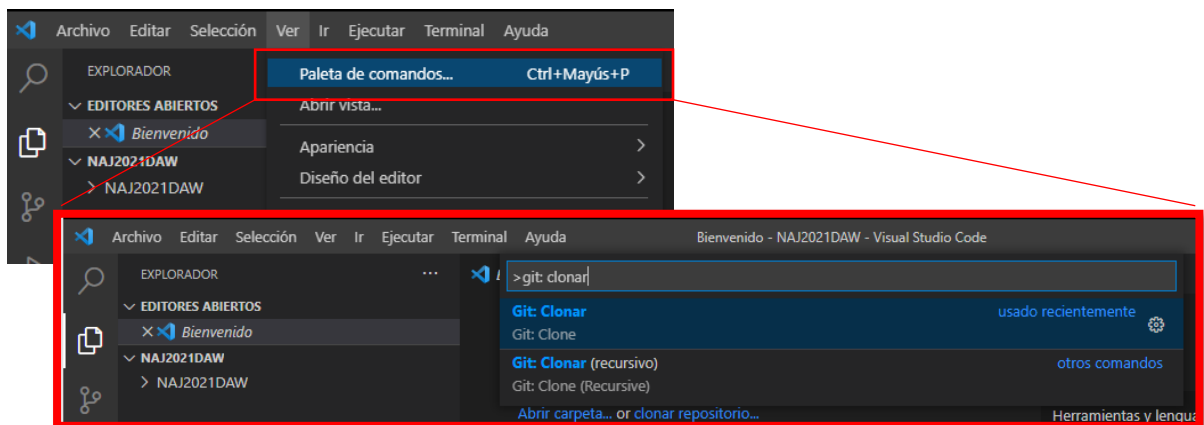
Dentro de Git debemos definir nuestro usuario y email de la siguiente manera:

```
$ git config --global user.name "eniun"  
$ git config --global user.email info@eniun.com
```



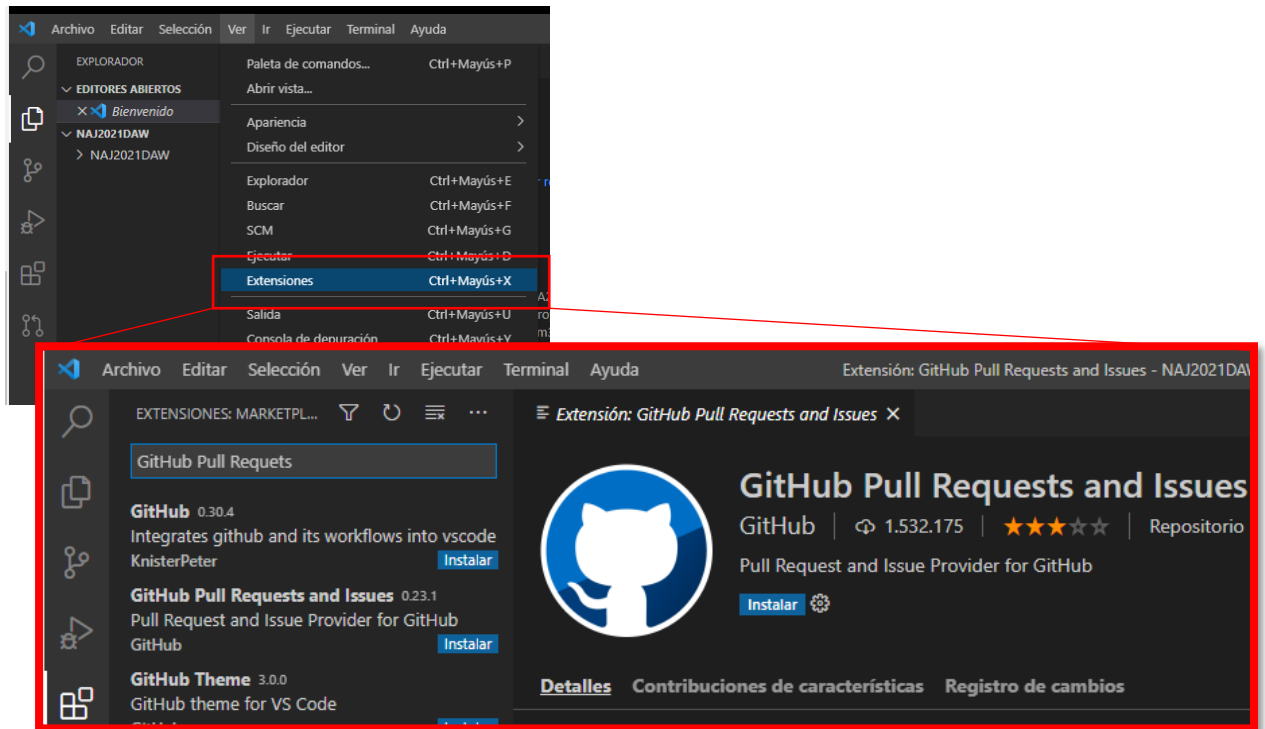
```
MINGW64:/  
nealj@DESKTOP-PB6R230 MINGW64 /  
$ git config --global user.name "N18AJ"  
  
nealj@DESKTOP-PB6R230 MINGW64 /  
$ git config --global user.email neanju96@gmail.com
```

Ahora vamos a clonar el proyecto creado en Github desde Visual Studio Code. Para ello, nos vamos al menú “Ver” seleccionamos “Paleta de comandos ...”. En ese punto buscamos “git: Clone”. La herramienta nos pedirá la ruta y ahí es donde tenemos que pegar la URL del repositorio que hemos creado en el punto anterior en Github.

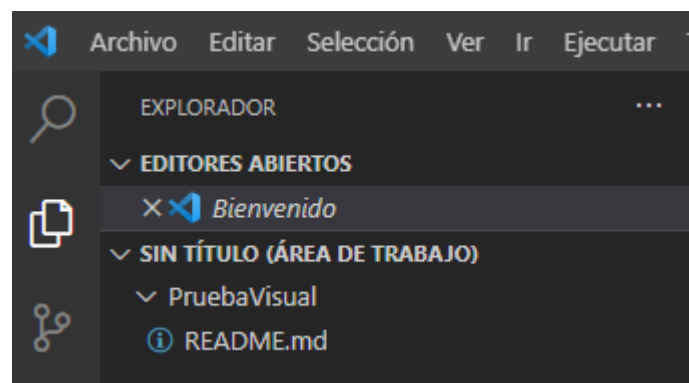
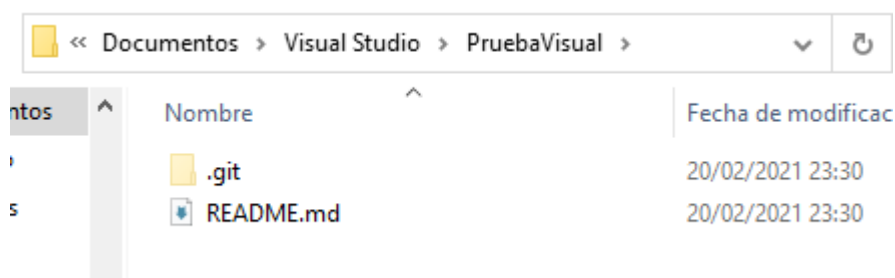


A continuación, tendremos que aceptar el uso del Visual Studio y el GitHub e iniciar sesión en nuestra cuenta GitHub. Crearemos la carpeta en la que trabajaremos con nuestro proyecto.

Antes de hacer un commit debemos instalar la extensión “GitHub pull request” desde el menú Ver > Extensiones.

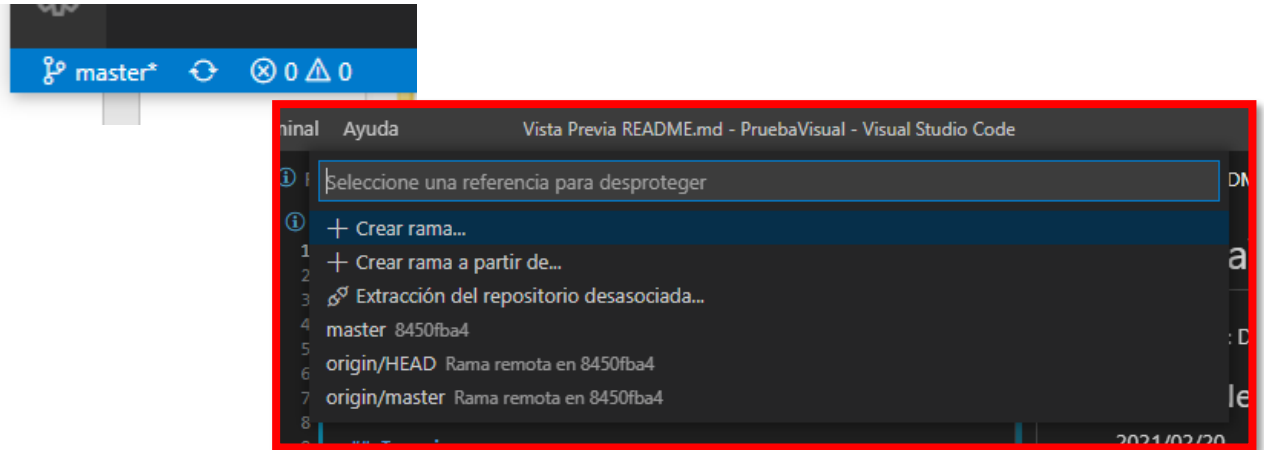


Aquí podemos ver la carpeta creada y la vista en el Visual Studio.

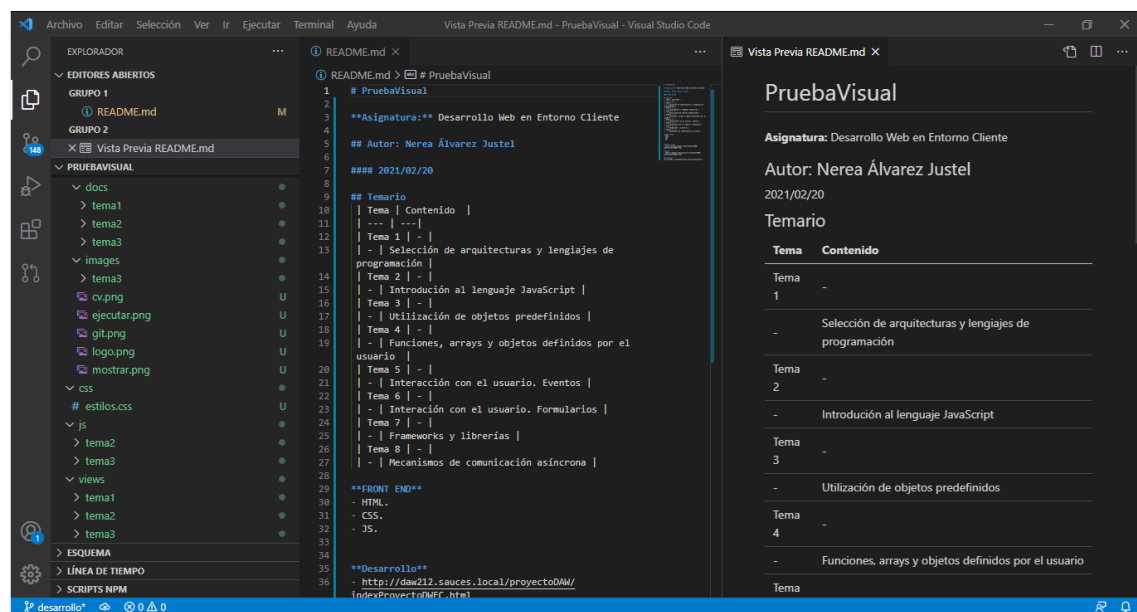


2. Manejo de ficheros en Visual Studio

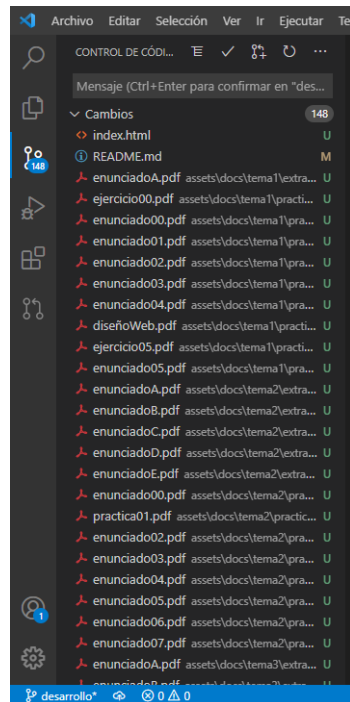
Primeramente, crearemos una rama “desarrollo”. Haremos clic en la parte inferior izquierda, donde se desplegarán las opciones.



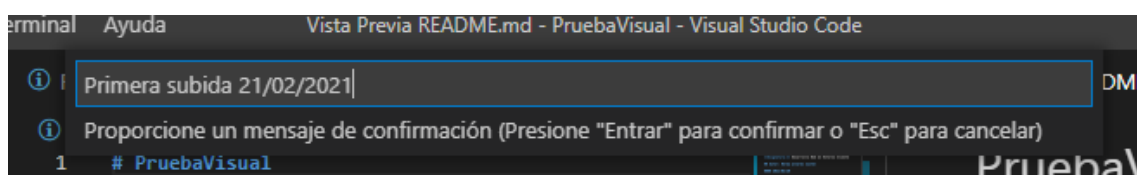
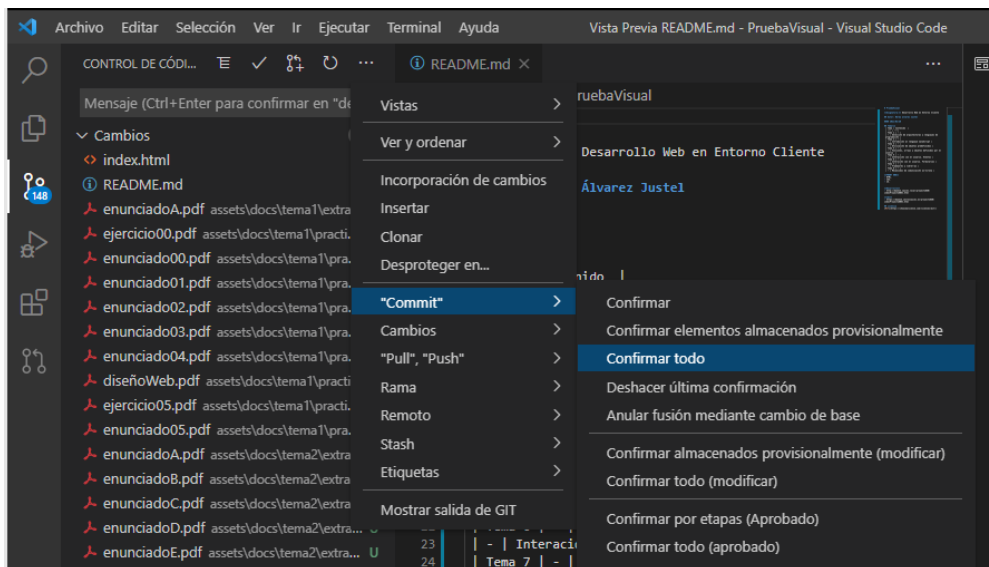
Ahora realizaremos cambios en el README y añadiremos algún fichero en y rutas de carpetas.



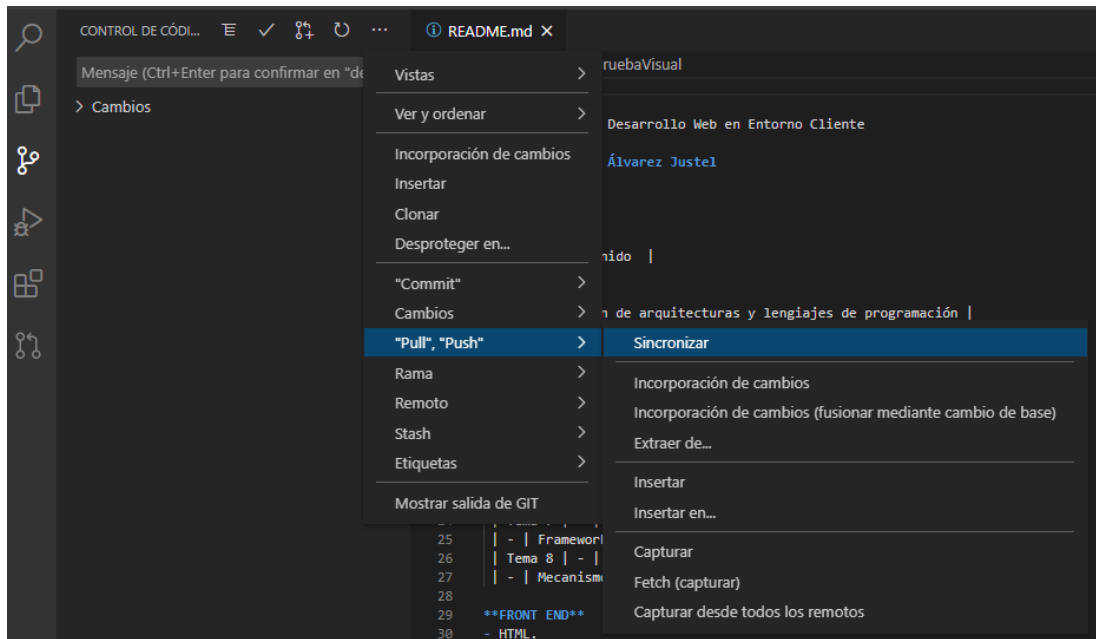
Seleccionando la pestaña de Control de Código fuente, veremos todos los cambios o elementos nuevos para la subida.



En este punto, realizaremos el primer commit y subida al repositorio. Haremos clic en los 3 puntos, se desplegará un montón de opciones, seleccionamos “Commit” y de las opciones la que mas se adapte a nuestras necesidades. Una vez seleccionado nos pedirá un mensaje de muestra.

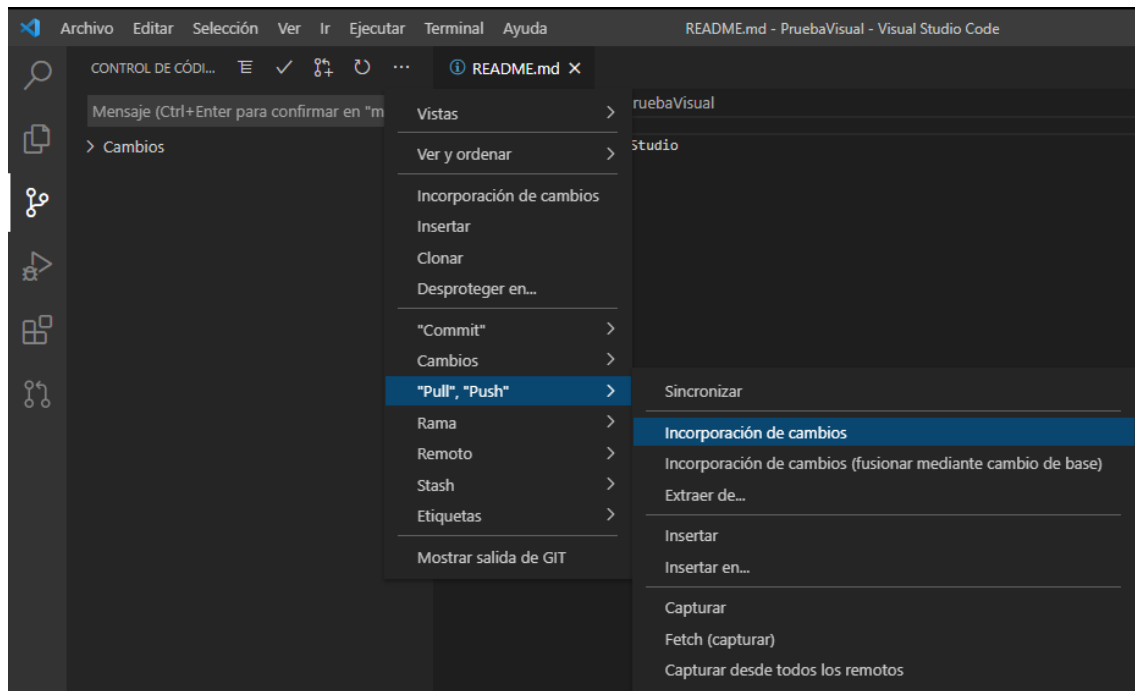


Realizaremos la misma selección a través de los 3 punto para hacer el “Push”.



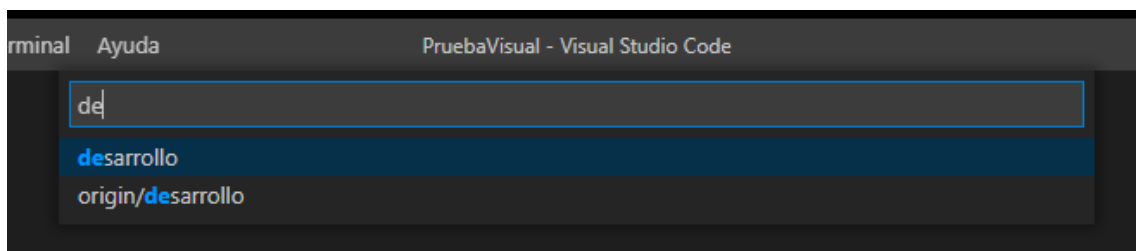
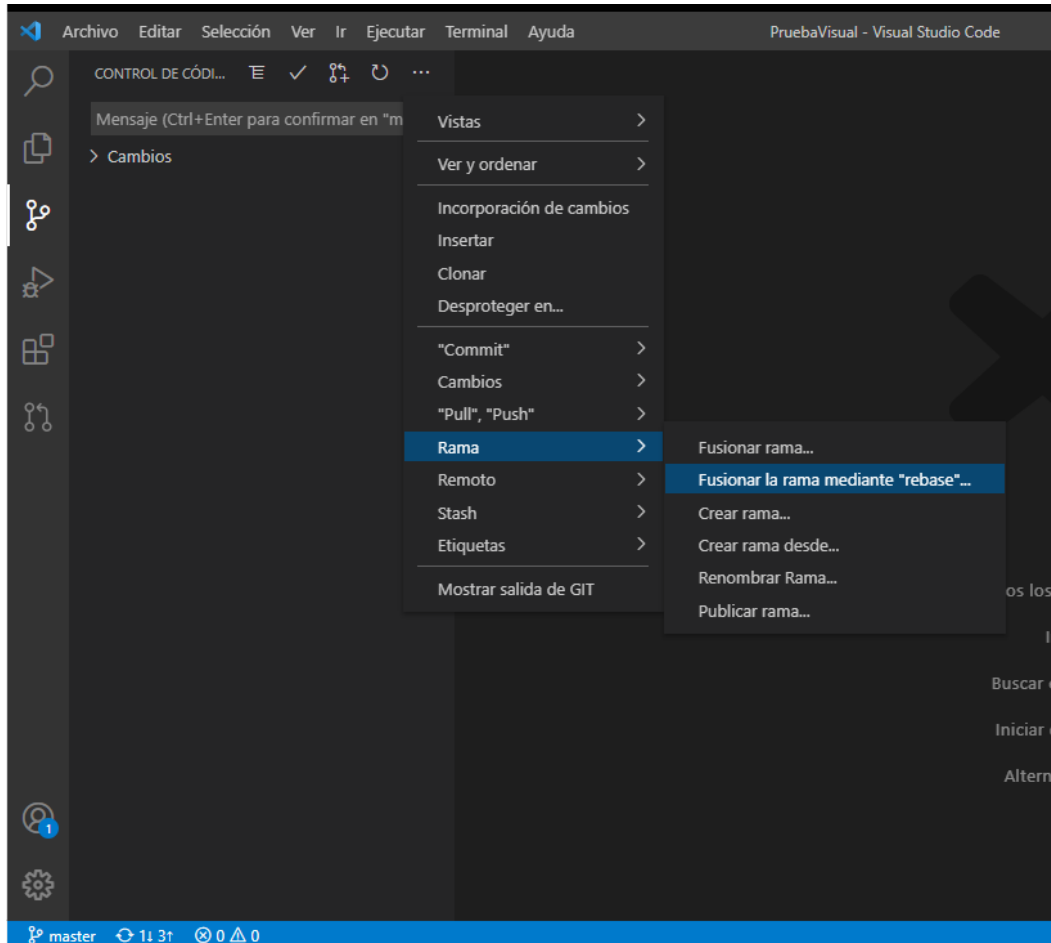
Ahora realizaremos un “Pull” de la rama desarrollo a la máster en el git, y nuevamente un “Pull” a la rama máster de nuestro Visual Studio.

Primero cambiaremos a la rama máster y después realizaremos la misma selección a través de los 3 punto para hacer el “Pull”.

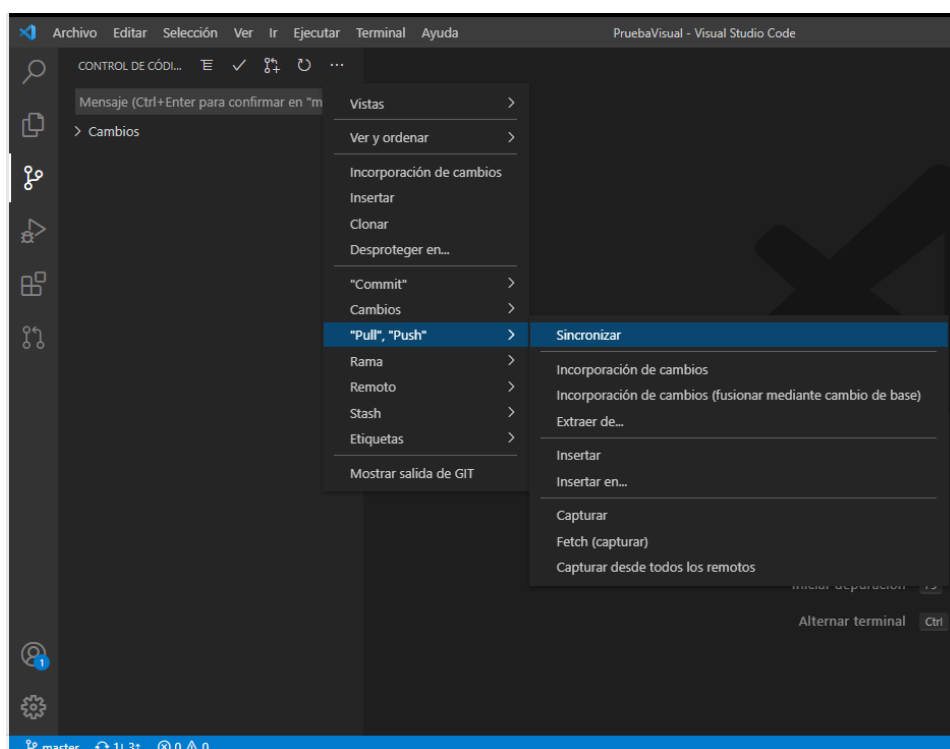
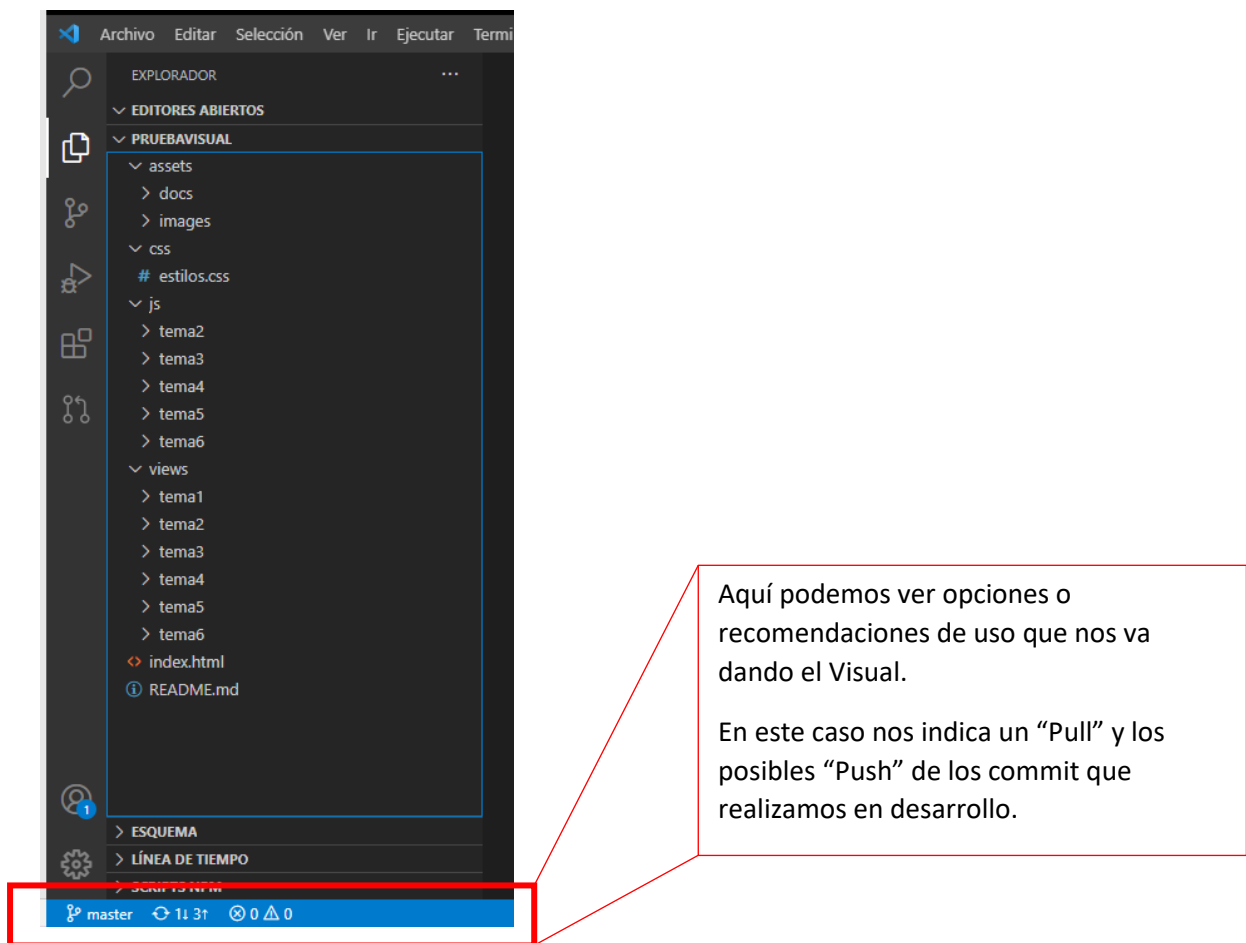


Mostrare alguna función más. Posicionados en “desarrollo” realizare varios “Commit”.

Ahora posicionados en la rama “master”, realizaremos un rebase de “desarrollo”. Y subiremos mediante un “Push” esos cambios al GitHub.



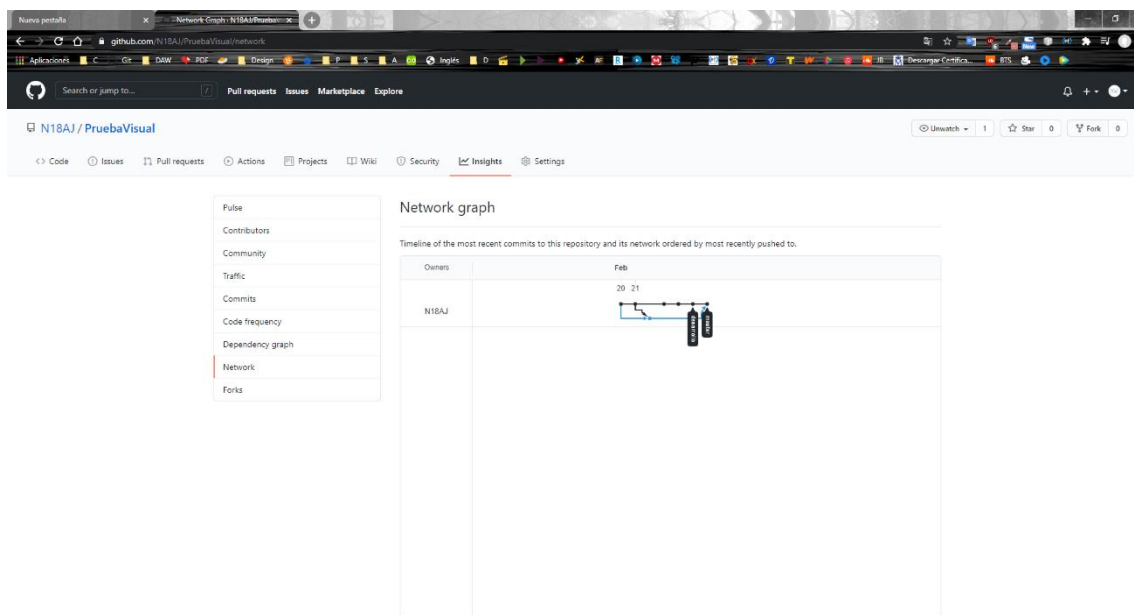
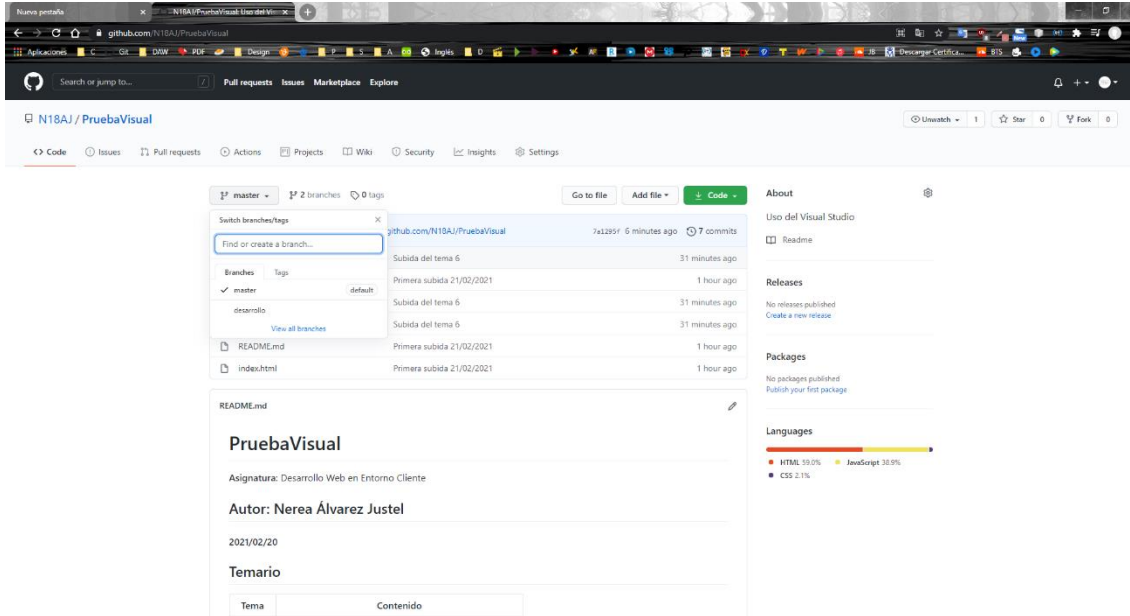
Haremos clic en la pestaña del “Explorador” y veremos todos los ficheros rebasados.



3. GitHub

Aquí se podrá ver todos los movimientos desde el GitHub:

<https://github.com/N18AJ/PruebaVisual>



CONCLUSIÓN

Mediante el Visual Studio Code considero que la manipulación de ficheros y el movimiento de ellos mediante un repositorio es mucho más fácil y automatizado. Tienes muchas opciones de sincronización o de realizar cambios en el GitHub desde el propio Visual, sin tener que realizarlos en la página del GitHub.

También considero que tienes mas desglosadas los tipos de movimientos, por lo que puedes saber mejor cual hacer o cual realizara mejor lo que necesitas en ese momento.

Todavía me queda mucho más que saber sobre este tema y el Visual Studio Code en general.

BIBLIOGRAFIA

https://es.wikipedia.org/wiki/Visual_Studio_Code

<https://www.eniun.com/repositorio-git-visual-studio-code-github/>

<https://github.com/N18AJ/PruebaVisual>