Bibiano montero

2º DAW

IFP Virgen de Gracia

Manual básico de git

desafío 1 despliegue aplicaciones web

Índice

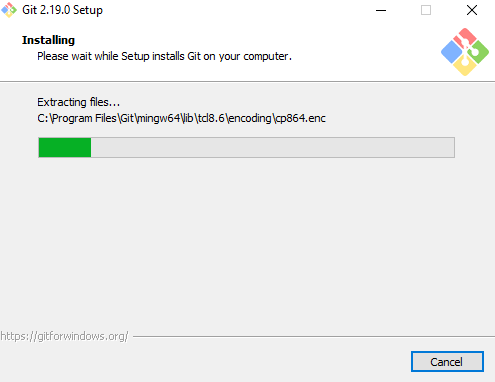
1. Preparar nuestro ordenador. Instalación del software necesario.
2. Creación del proyecto
3. Alta en GitHub y creación del repositorio
4. Subir el proyecto a la nube
5. Creando una segunda versión. Ramificación
6. Clonar un repositorio
7. Eliminar un repositorio

# Instalación del software necesario

Primero necesitamos instalar el cliente de Git, es un software que encontraremos en su web <https://git-scm.com/>.



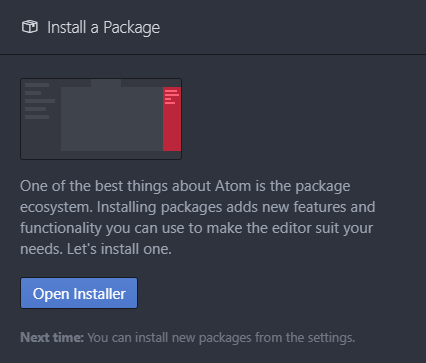
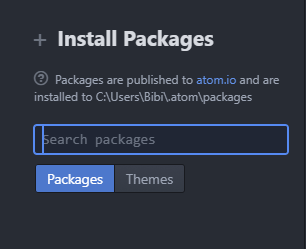
Hacemos clic en descargar y pasamos a la instalación.

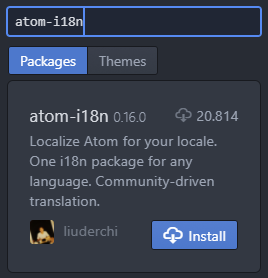
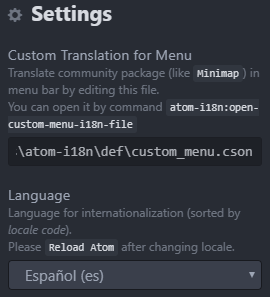
Durante la instalación dejamos todas las opciones predeterminadas. Una vez terminado, pasamos a la instalación del editor de código. En este caso utilizaremos Atom, para ello nos dirigimos a su web ( [https://atom.io/](https://atom.io/%20) ) y descargamos el instalador.



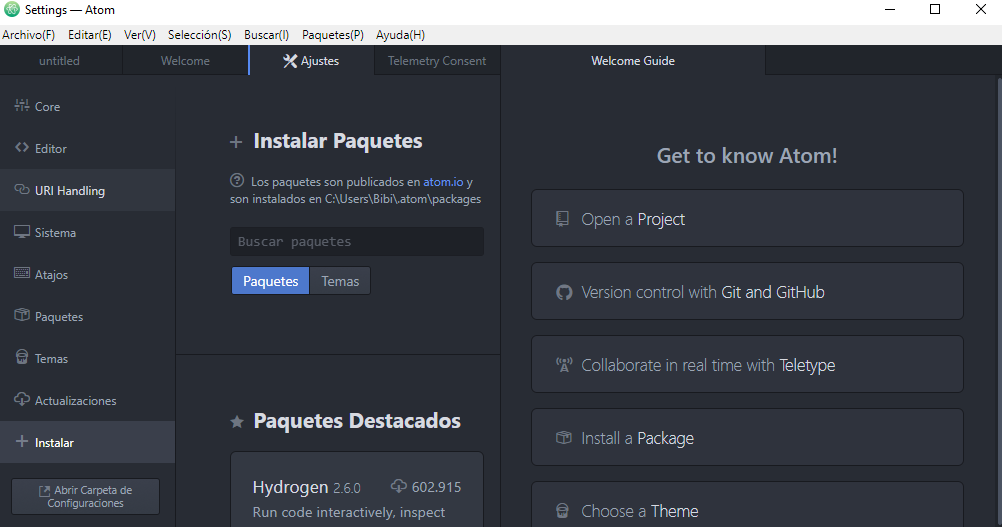
Una vez hacemos clic en el instalador, Atom se instala automáticamente sin necesidad de supervisar el proceso de instalación. A continuación, y esto es opcional, descargaremos un paquete de idiomas para facilitar el uso de la interfaz del programa. Para ello hacemos clic en Install a Package y seleccionamos Open Installer.

En Search packages introducimos el nombre del paquete que buscamos, en este caso se trata de atom-i18n, lo escribimos y hacemos clic en Install. El proceso no tarda nada y en breves segundos tenemos acceso a la configuración del paquete. Seleccionamos Español (es) y nos pedirá que reiniciemos la aplicación.

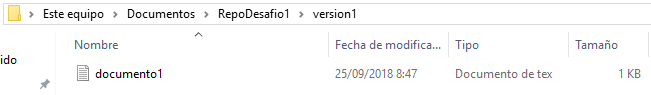
Al abrirse de nuevo la aplicación, veremos que los menús ya están en nuestro idioma.

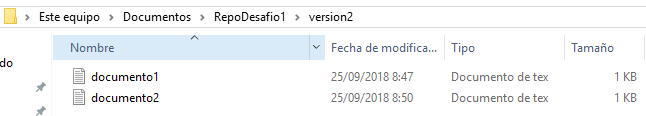


# Creación de nuestro proyecto

Trabajando en local, crearemos una carpeta para nuestro proyecto, yo la crearé en Mis Documentos con el nombre RepoDesafio1. Dentro de este, crearemos otra carpeta llamada version1 con un archivo de texto que llamaremos documento1, donde testearemos con los comandos de Git.

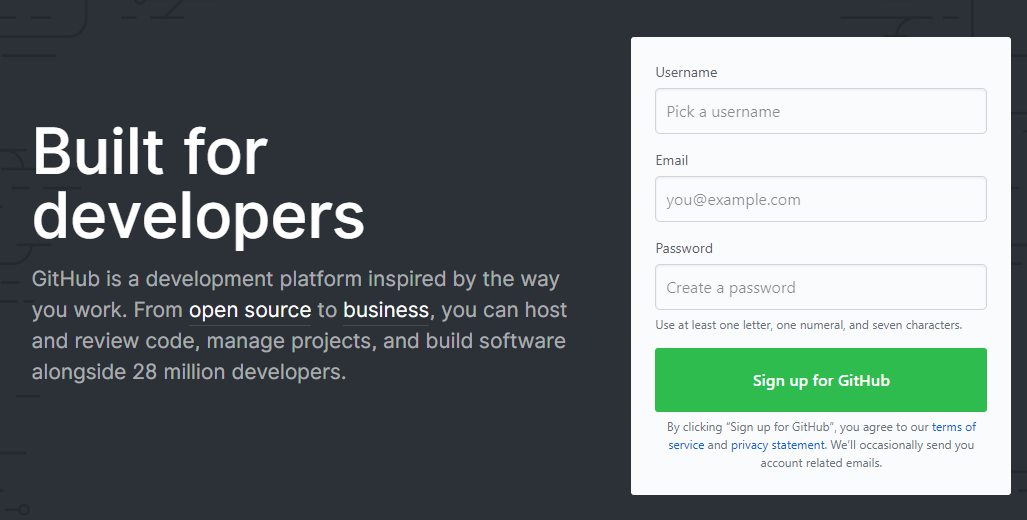
Una vez tengamos nuestro directorio listo, crearemos una copia donde realizaremos algún cambio, yo añadiré un segundo documento de texto.



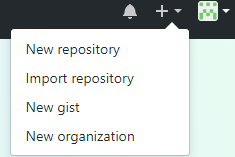


# Primeros pasos en GitHub

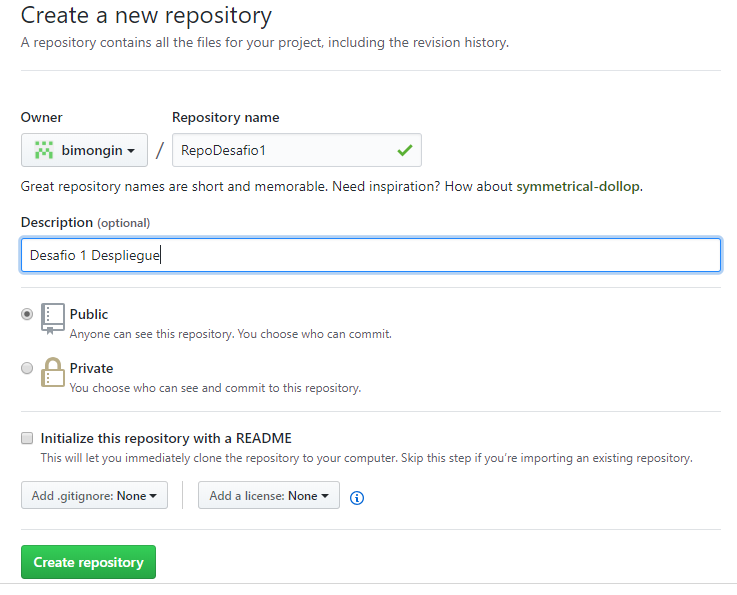
Entramos en la web de GitHub ( <https://github.com/> ) y nos registramos con una cuenta de correo.



Una vez nos demos de alta nos mostrará nuestro perfil de usuario de GitHub, así que pasaremos a crear un nuevo repositorio para introducir nuestro proyecto. Para ello hacemos clic en el botón “+” y después a New repository.



Le ponemos un nombre al repositorio, en este caso lo llamaré igual (RepoDesafio1), y dejamos marcada la opción de repositorio público. Después hacemos clic en Create repository.



Una vez lo creemos nos proporcionará una URL donde podremos subir nuestro proyecto.

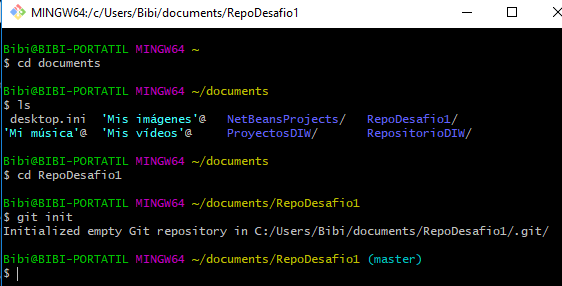


Terminado esto, ya podemos conectarnos a nuestro repositorio de Git y trabajar los comandos.

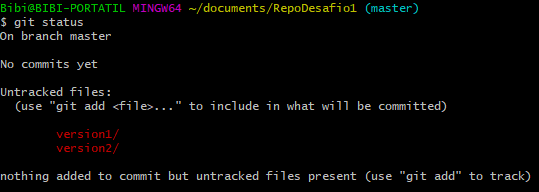
# Subiendo nuestro proyecto

Puesto que la idea es aprender a usar los comandos, buscamos en nuestro ordenador la aplicación Git Bash, la consola de comandos de Git, y trabajaremos desde ahí.

Lo siguiente es buscar mediante comandos el directorio de nuestro proyecto en local, así que lo buscamos mediante comandos y, una vez en la carpeta del proyecto, iniciamos Git con el comando Git init.

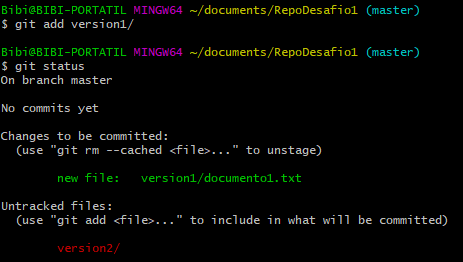


Ahora comprobaremos el estado de nuestro repositorio, con el comando Git status

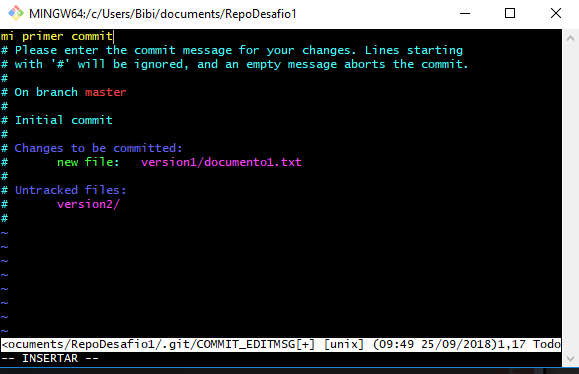


Como podréis comprobar, las dos carpetas de versiones salen en rojo; eso significa que aún no se ha hecho ningún commit. Así que lo siguiente es subir una parte del proyecto.

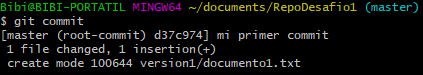
Usamos el comando git add, seguido de version1/, así Git sabrá que queremos trabajar con lo que esté en esa carpeta.



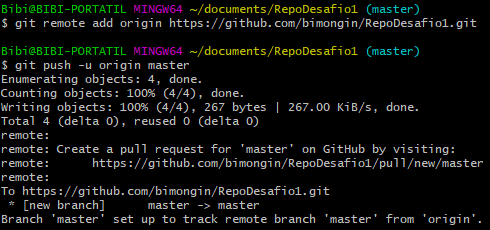
Ahora se muestra en verde, indicando que podemos hacer ya un commit de nuestro proyecto. Usamos el comando git commit y se nos notificará de los cambios.



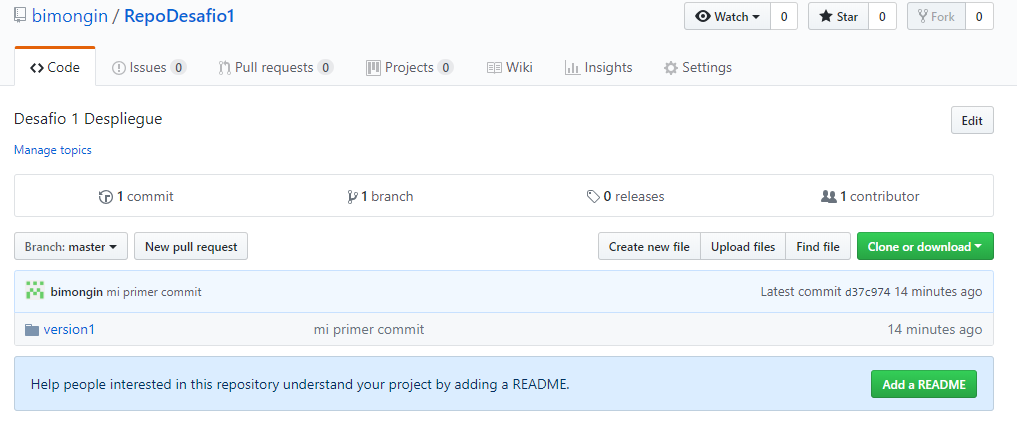
Le ponemos un nombre para identificarlo, yo en este caso lo he llamado “mi primer commit”. Y escribimos “:qw” para salir y guardar los cambios. Entonces nos informará de los cambios realizados.



Ahora ya sólo queda hacer el push a nuestro repositorio de GitHub, para ello usamos el comando “git remote add origin” seguido de la URL que nos proporcionó antes. Y después hacemos el push con el comando “git push –u origin master”.

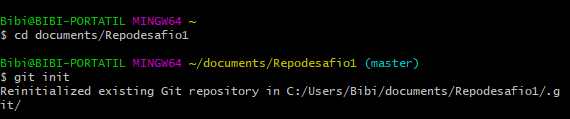


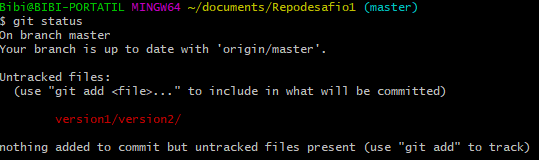
Nos notifica que todo bien y correcto, así que vamos de nuevo a nuestro panel en GitHub y veremos que ya está nuestro primer commit subido a la nube.



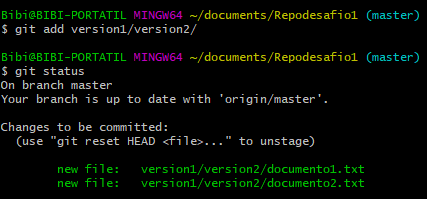
# Creando una segunda versión. Ramificación.

El siguiente paso va a ser modificar nuestro repositorio, añadiendo la carpeta version2 dentro de la version1. Para ello seguimos los mismos pasos que antes para conectarnos al repositorio (en caso de que hayamos desconectado).

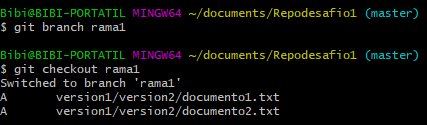




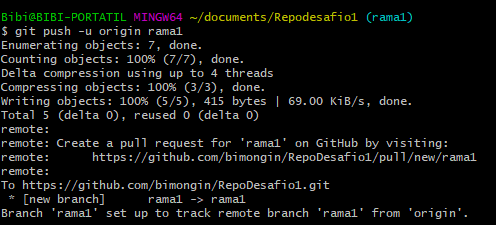
Lo siguiente es hacer el git add de version2.



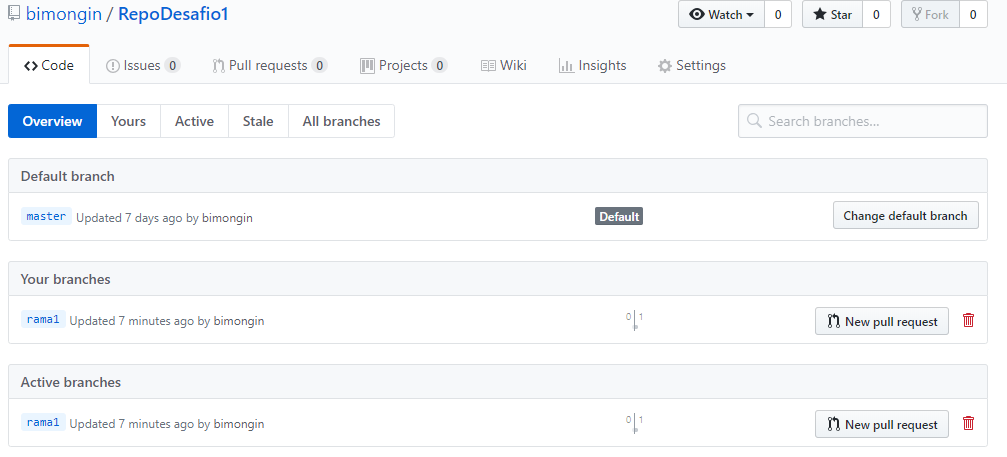
Ahora crearemos la rama1, para ello usamos el comando branch seguido del nombre que queramos darle, y después checkout para trabajar como rama1.



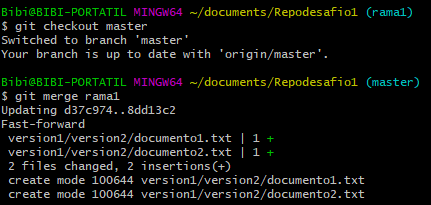
Todo lo que hagamos ahora se guardará en la rama1, para comprobarlo haremos un commit del estado de la rama y el push para subirla.



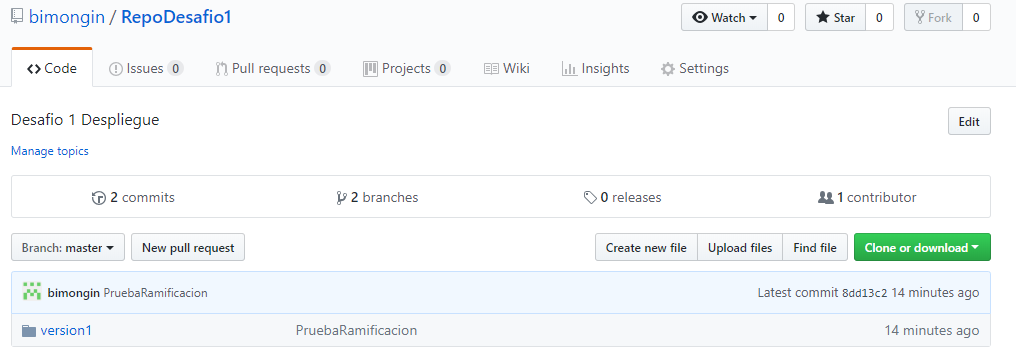
Para comprobar que todo está correctamente, vamos a nuestro repositorio en GitHub y comprobamos los cambios.



Como se puede ver, aparece rama1 como una rama. Lo siguiente es fusionar el contenido de rama1 con la rama master. Para ello hacemos uso del comando merge.



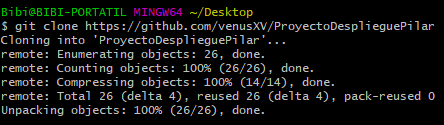
Ahora la rama Master también ve el contenido de la rama1.



# Clonar un repositorio

Por último, clonaremos el repositorio de un compañero en nuestro PC, para ello vamos al Git Bash y elegimos una carpeta donde clonarlo. Yo lo haré en el escritorio.

Para ello usamos el comando git clone seguido de la url del repositorio de nuestro compañero. En este caso clonaré el repositorio de Pilar.



Al hacer esto ya nos aparece en nuestro escritorio su repositorio, con el que podremos trabajar.

# Eliminar un repositorio

Para eliminar un repositorio se puede hacer mediante el comando rm –rf .git, o bien hacerlo desde la web de GitHub. En la zona de Settings, si bajamos al final de la página encontraremos la Danger Zone, haciendo clic en Delete this repository y confirmando se elimina por completo.

