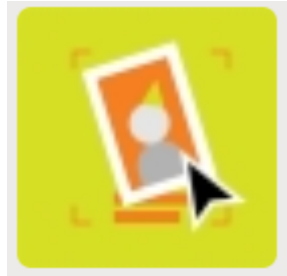


# DAW2-DWES



## TEMA 2

### DESARROLLO WEB EN ENTORNO SERVIDOR

Instalación, Configuración y Documentación  
del Entorno de Desarrollo y del Entorno de Explotación.

2021-2022

0. Instalación, configuración inicial y actualización de Windows 10. ....	3
1. PuTTY: administración remota del servidor.....	3
2. Notepad++.....	3
3. FileZilla: administración de archivos desde cliente.....	3
2.6.1 Instalación.....	3
2.6.2 Configuración.....	8
2.6.3 Ejemplo de uso.....	10
4. NetBeans: instalación y uso.....	11
4.1. Instalación y configuración inicial de NetBeans.....	11
4.2. Creación, modificación, transporte y borrado de un proyecto.....	12
4.3. Proyectos remotos. ....	12
4.4. Depuración de un programa.....	12
1. MySQL: administración de las bases de datos.....	12
2. GitHub: administración de repositorios. ....	17
6.1. GitHub y los tokens de acceso.....	17
6.2. Clonar un repositorio.....	19
6.3. Inicializar un repositorio.....	21
6.4. Guardar cambios y sincronizar el repositorio ( <i>commit</i> , <i>push</i> y <i>pull</i> ).....	22
6.4.1. Hacer <i>commit</i> . ....	22
Amend last commit. ....	23
6.4.2. Subir commits: <i>push</i> . ....	24
6.4.3. Bajar commits: <i>pull</i> . ....	26
6.4.4. Deshacer cambios y commits. ....	27
Deshacer cambios. ....	27
Deshacer commits.....	28
6.5. Creación y uso de ramas de trabajo ( <i>branches</i> ). ....	29
6.5.1. Crear ramas. ....	29
6.5.2. Cambiar entre ramas. ....	32
Pasar a una revisión concreta de una rama.....	33
Detached heads.....	34
6.5.3. Eliminar ramas.....	34

0. Instalación, configuración inicial y actualización de Windows 10.

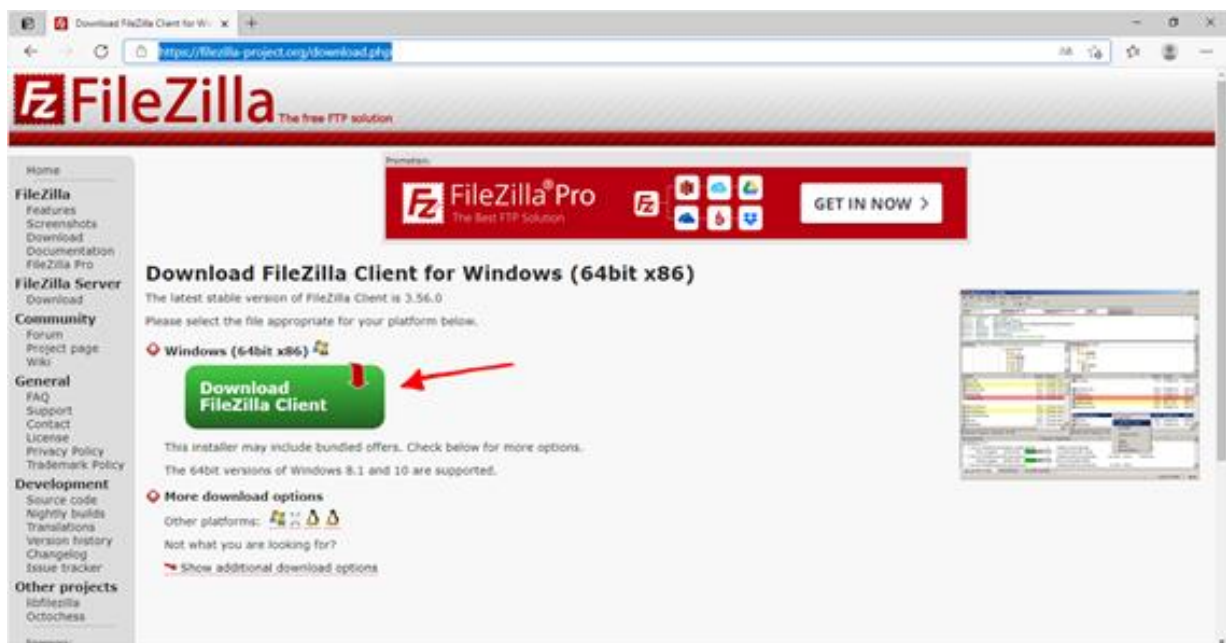
1. PuTTY: administración remota del servidor.

2. Notepad++

3. FileZilla: administración de archivos desde cliente.

#### 2.6.1 Instalación

Primero nos vamos a la [página web de descarga](https://filezilla-project.org/download.php) y nos bajamos el instalador para Windows de 64bit dando en **Download FileZilla Client**.



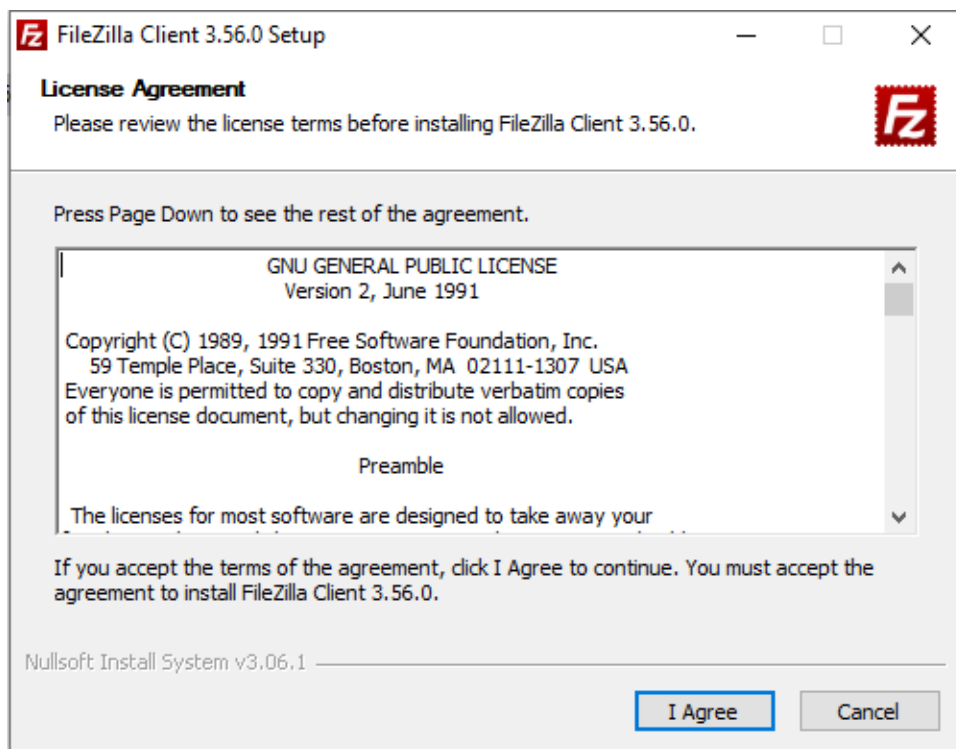
Cuando le damos a descargar, nos pregunta que edición de FileZilla, seleccionaremos la que se llama solo FileZilla y le damos a **Download**.

### Please select your edition of FileZilla Client

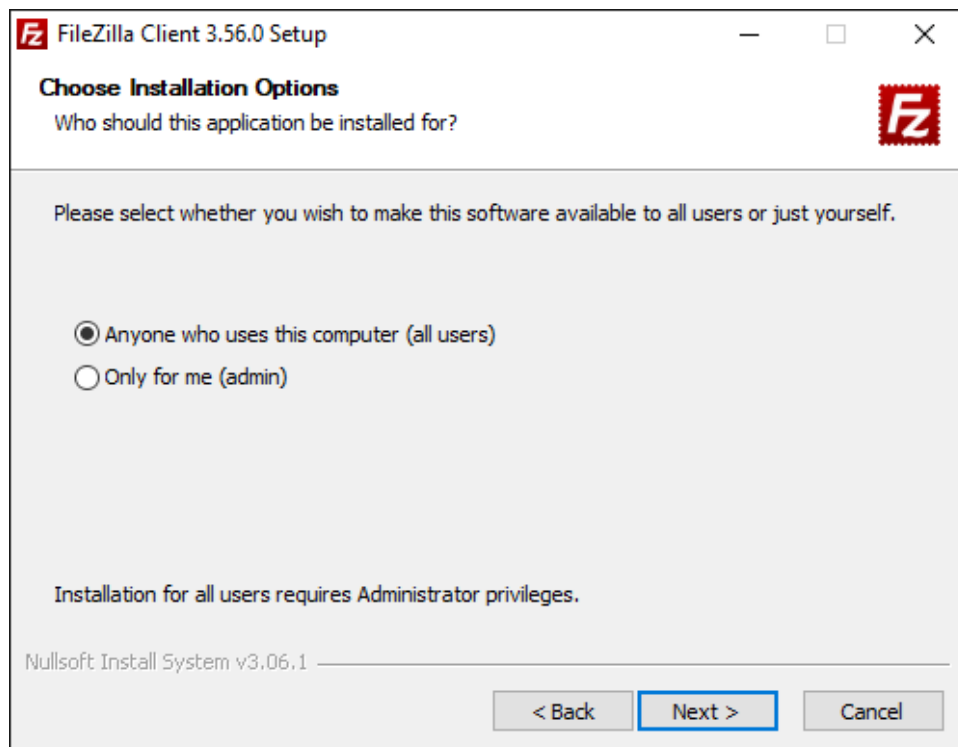
	FileZilla	FileZilla with manual	FileZilla Pro	FileZilla Pro + CLI
Standard FTP	Yes	Yes	Yes	Yes
FTP over TLS	Yes	Yes	Yes	Yes
SFTP	Yes	Yes	Yes	Yes
Comprehensive PDF manual	-	Yes	Yes	Yes
Amazon S3	-	-	Yes	Yes
Backblaze B2	-	-	Yes	Yes
Dropbox	-	-	Yes	Yes
Microsoft OneDrive	-	-	Yes	Yes
Google Drive	-	-	Yes	Yes
Google Cloud Storage	-	-	Yes	Yes
Microsoft Azure Blob + File Storage	-	-	Yes	Yes
WebDAV	-	-	Yes	Yes
OpenStack Swift	-	-	Yes	Yes
Box	-	-	Yes	Yes
Site Manager synchronization	-	-	Yes	Yes
Command-line interface	-	-	-	Yes
Batch transfers	-	-	-	Yes

Download
Select
Select
Select

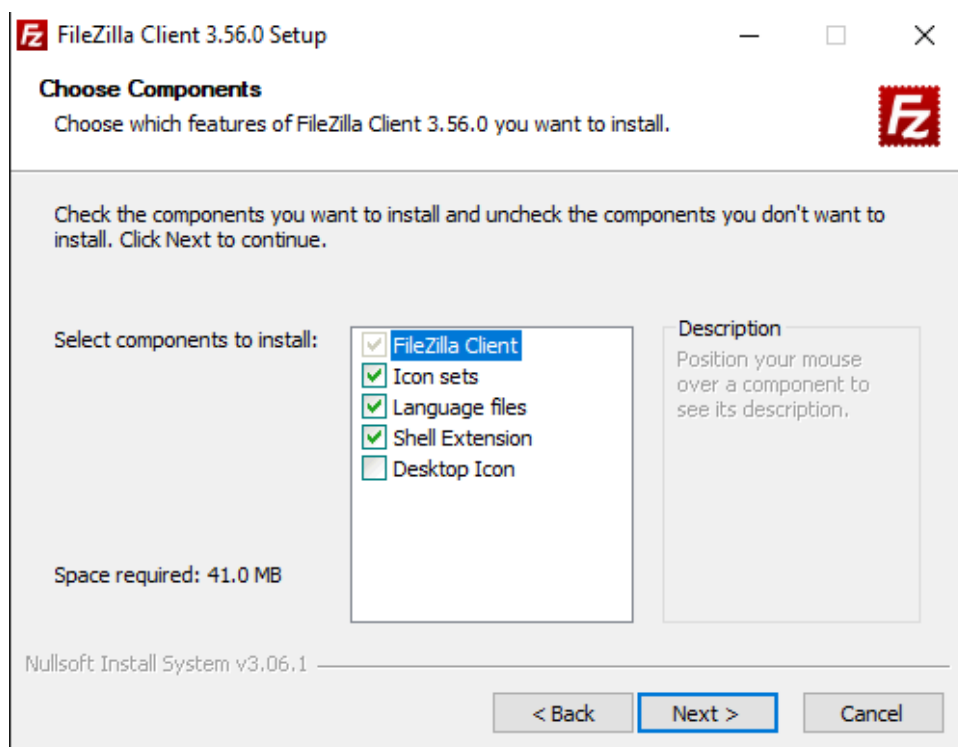
Una vez bajado el instalador lo ejecutaremos, lo primero será aceptar los términos de la licencia. Le daremos a **I Agree**.



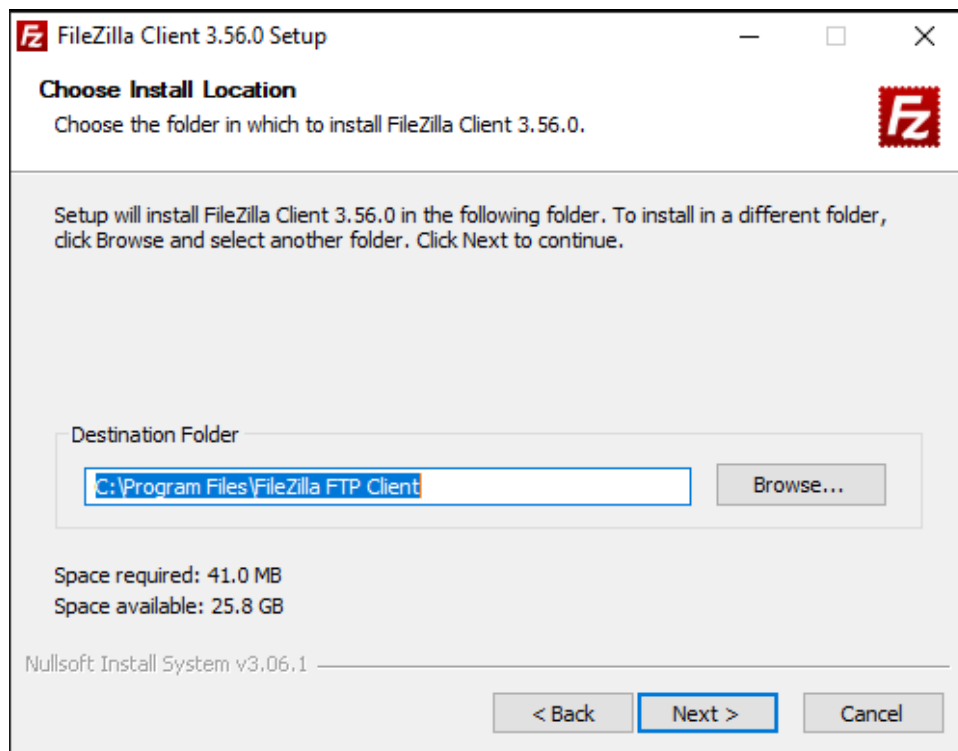
Ahora nos preguntara en que usuarios haremos la instalación, en este caso lo voy a hacer para todos los usuarios. Y le damos a **Next**.



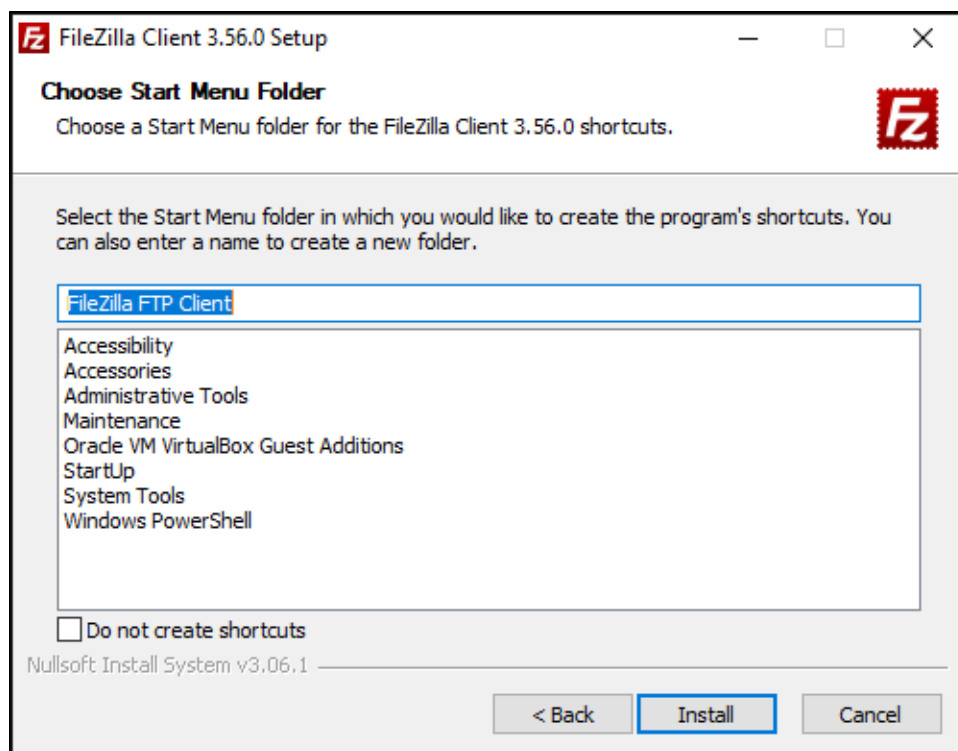
Lo siguiente será elegir los componentes que nos va a instalar, dejaremos marcado lo que viene por defecto, ya que es lo que necesitamos. Le damos a **Next**.



Ahora nos pedirá la ubicación de donde va a realizar la instalación en nuestro equipo, en mi caso lo he dejado por defecto, pero podemos elegir la ruta que queramos. Y le damos a **Next**.



Por último, nos pedirá elegir el menú de inicio, en este caso lo dejamos por defecto. Y ya le damos a **Install**.



Una vez finalice la instalación nos mostrara esta ventana, ya estaría instalado el cliente de FileZilla. Dejaremos marcada la opción de Start FileZilla now para que se nos abra al darle a **Finish**.



## Completing FileZilla Client 3.56.0 Setup

FileZilla Client 3.56.0 has been installed on your computer.

Click Finish to close Setup.

☒ Start FileZilla now

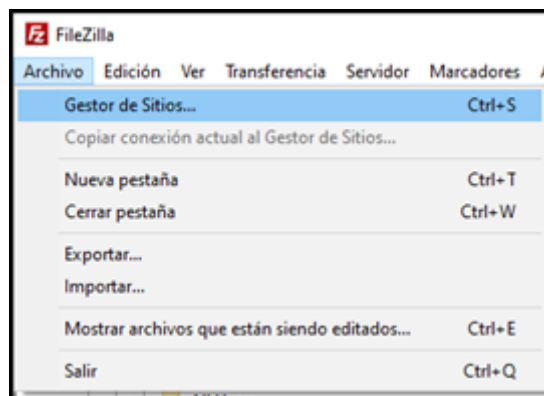
< Back

Finish

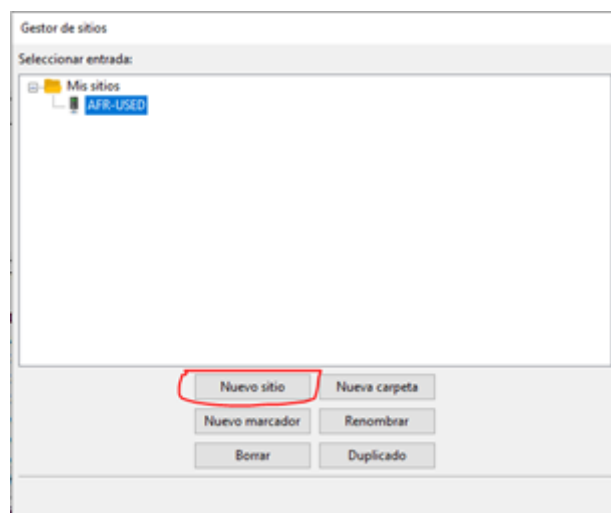
Cancel

## 2.6.2 Configuración

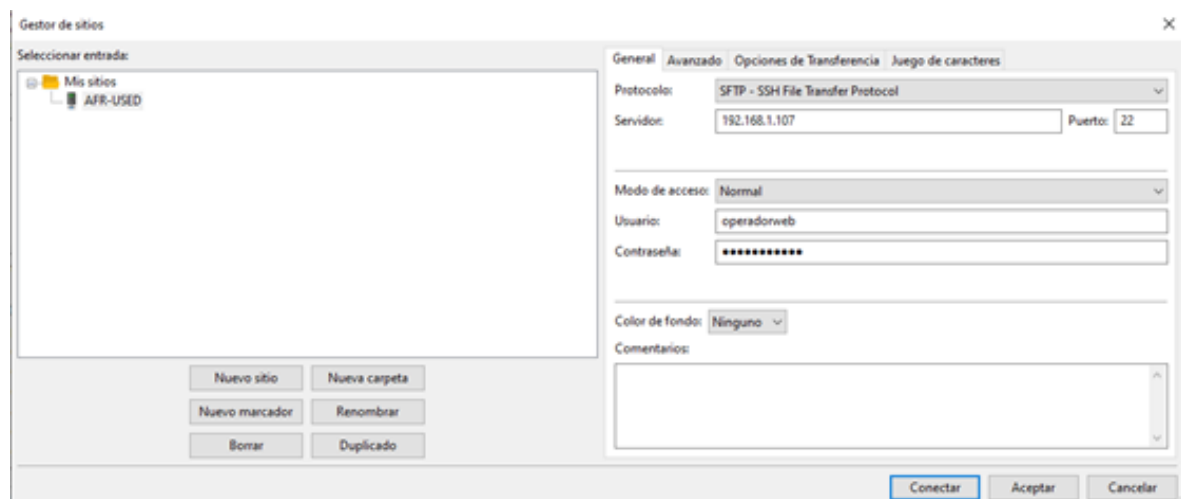
Una vez instalado el cliente de FileZilla, lo abrimos. Nos iremos a Archivo y Gestor de Sitios...



Una vez se nos abra el gestor de sitios, le daremos a **Nuevo Sitio** para configurar la conexión a nuestro servidor de Ubuntu y poder acceder más fácilmente.

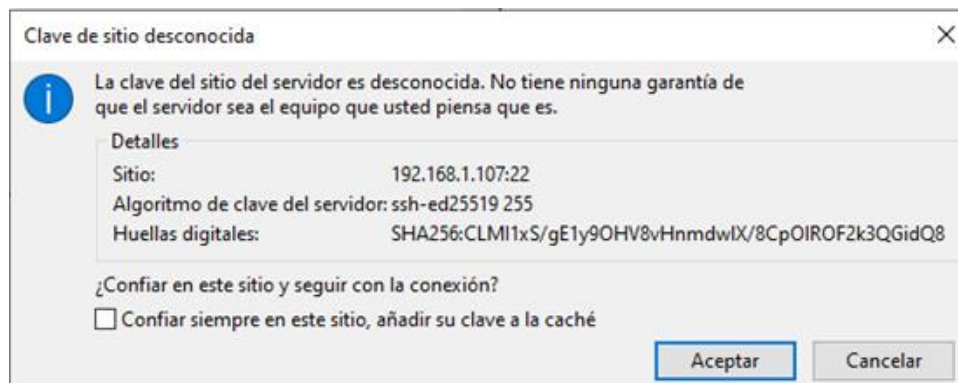


Una vez tengamos agregado un sitio con el nombre que queremos, configuraremos el apartado de General, el protocolo que usaremos para la conexión será SFTP, el servidor será la IP de nuestro Ubuntu Server, el puerto el 22, y el usuario y contraseña serán los de operadorweb que es el administrador del sitio. Una vez configurado le daremos a **Conectar**.

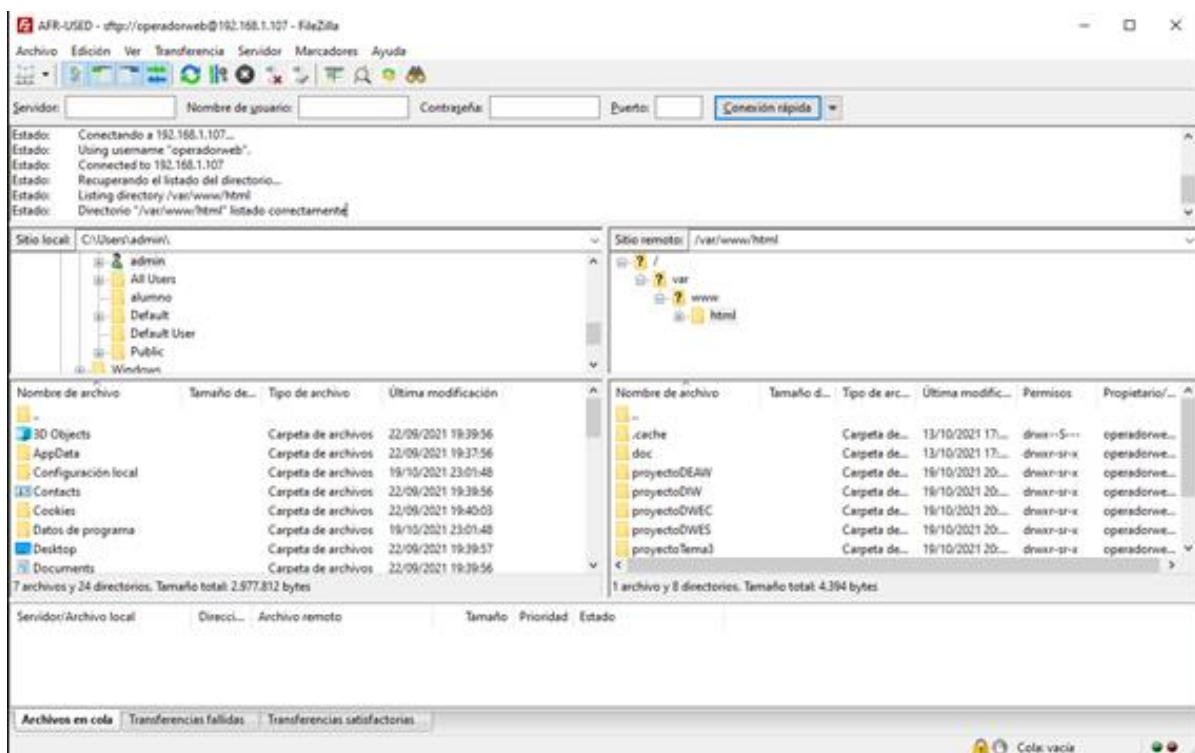




Nos dirá que la clave del servidor es desconocida, igualmente le daremos a **Aceptar** para conectarnos.

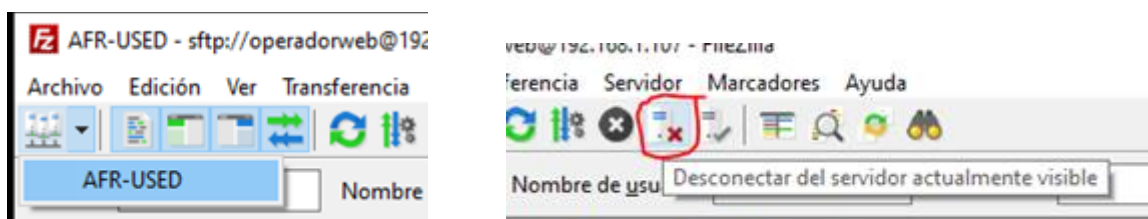


Y si todo está bien configurado, se realizara la conexión con nuestro servidor. Desde aquí ya podremos administrar los archivos y directorios de nuestro servidor web.



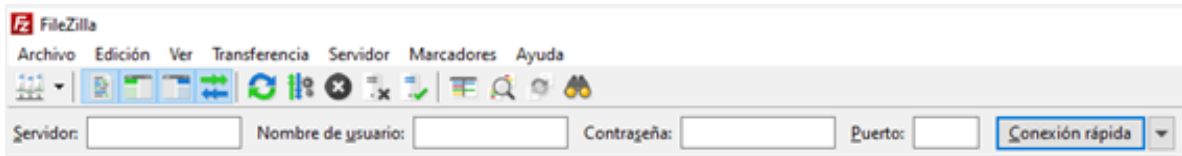
Para acceder a la conexión rápida que hemos configurado, solo tenemos que desplegar la pestaña de la izquierda superior y nos saldrá dicha conexión.

También podremos desconectarnos del servidor desde el botón de la imagen y realizar otra conexión.

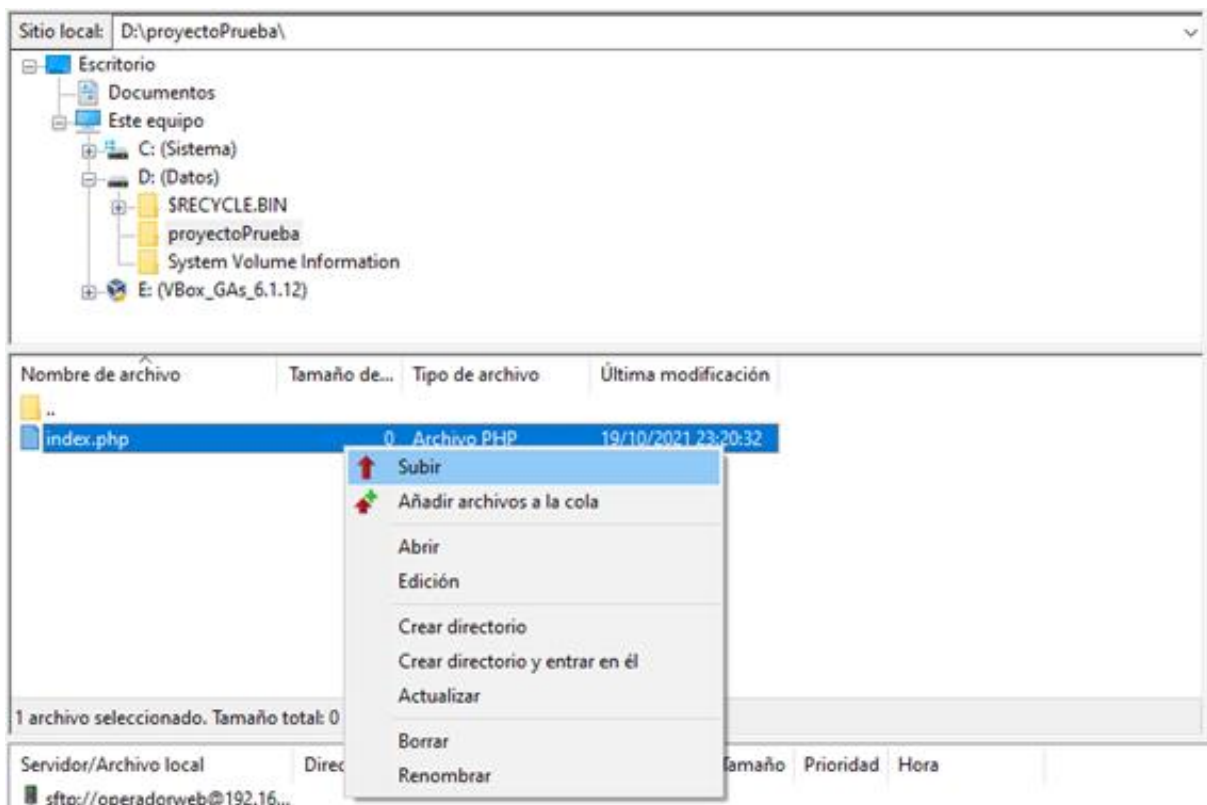


### 2.6.3 Ejemplo de uso

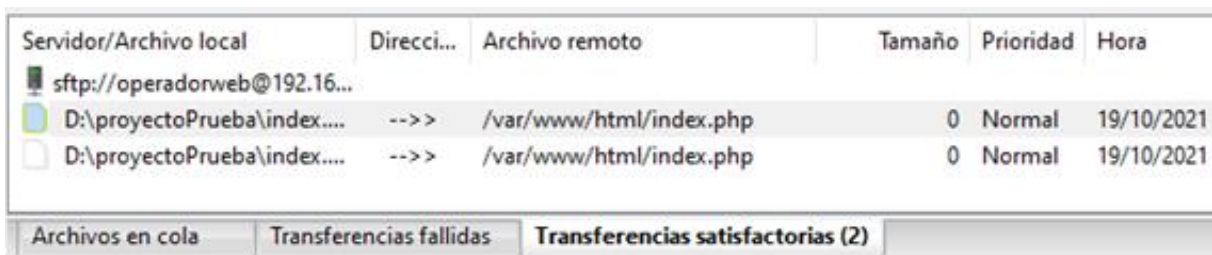
Nos conectamos a nuestro servidor, mediante la conexión que creamos anteriormente o a otra específica desde la parte superior metiendo los datos.



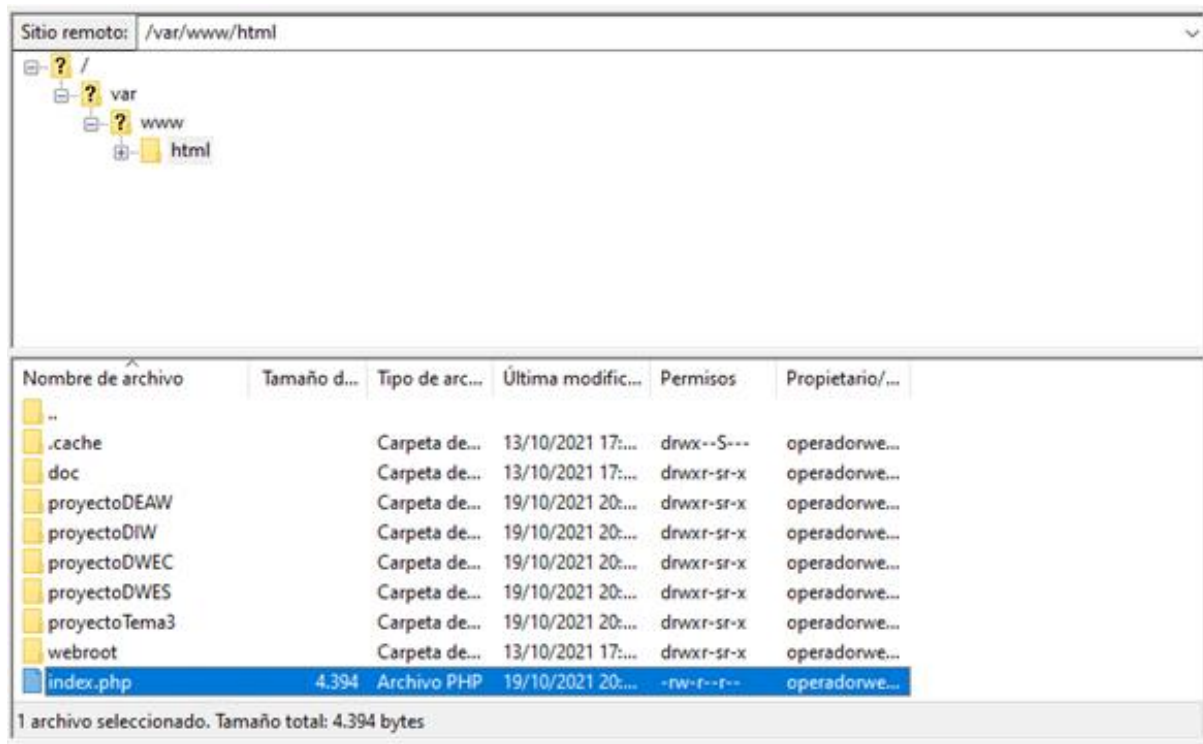
Para subir los archivos de nuestra web a el servidor, tendremos que irnos a el Sitio local: en la parte izquierda de la ventana y aquí seleccionar los archivos a subir, le damos clic derecho y a subir. **IMPORTANTE:** Estar ubicados en el directorio remoto donde queremos subir los archivos en el lado del servidor.



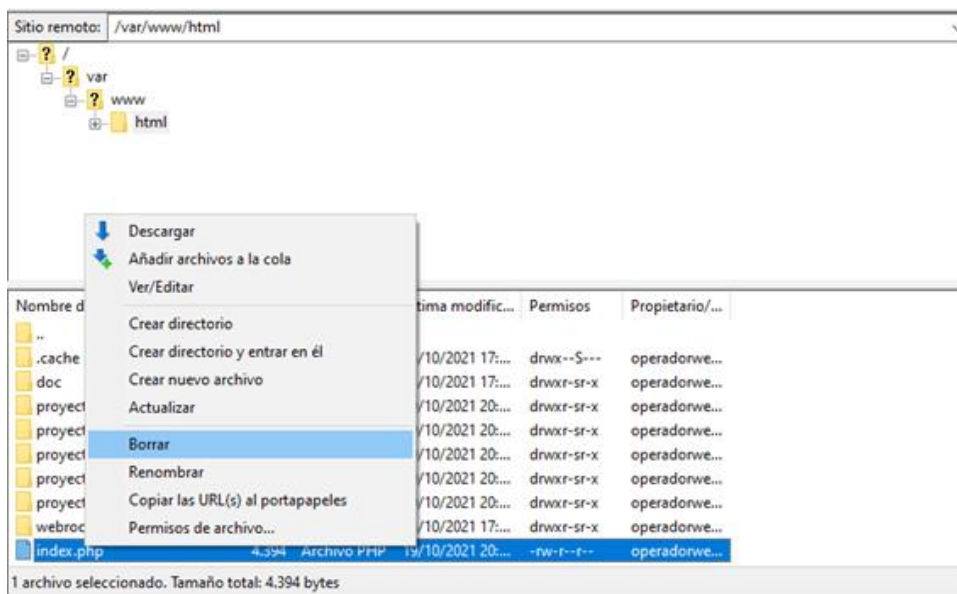
Si la transferencia se ha realizado con éxito, nos lo mostrara en la parte inferior de la ventana, aquí nos muestra donde se ha subido y en que directorio del sitio remoto.



Si la transferencia se ha realizado con éxito, solo tendremos que comprobar en la parte derecha de la ventana donde nos sale el sitio remoto si están el archivo o archivos que hemos subido.



También podemos borrar del Sitio Remoto archivos con clic derecho y borrar.



## 4. NetBeans: instalación y uso.

### 4.1. Instalación y configuración inicial de NetBeans.

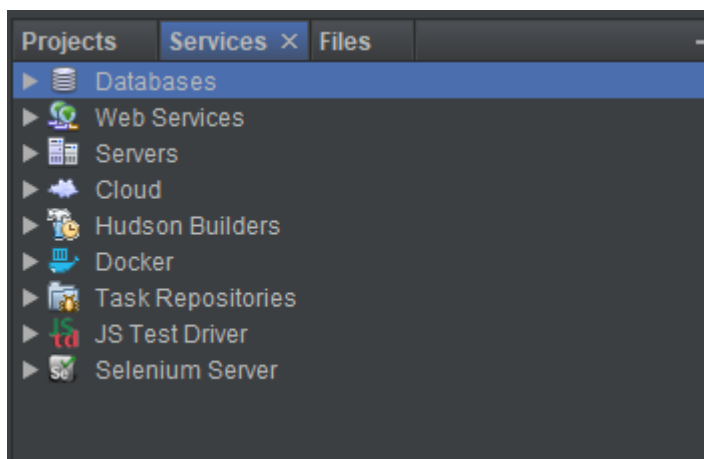
4.2. Creación, modificación, transporte y borrado de un proyecto.

4.3. Proyectos remotos.

4.4. Depuración de un programa.

## 1. MySQL: administración de las bases de datos.

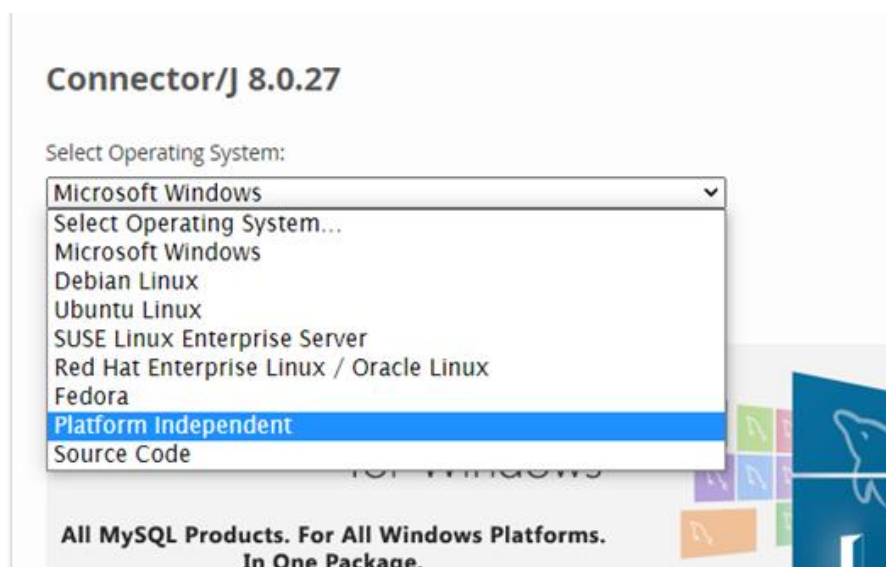
MySQL se utiliza en NetBeans desde la pestaña de Servicios y el apartado de “Databases”



Lo primero que hay que hacer es descargar el conector de MySQL para NetBeans. Se trata de un archivo .jar que se especificará al IDE. Lo recomendable es que los archivos de este tipo estén dentro de una carpeta llamada “software” para poder indicar su ruta sin pérdida.

El conector se puede descargar desde <https://dev.mysql.com/downloads/connector/j/>

Antes que nada, en el apartado del Sistema Operativo hay que elegir la opción de “Platform Independent” para poder descargar el .zip con el archivo .jar que necesitamos:



Seleccionamos la segunda opción, el fichero .zip:

Select Operating System:  
Platform Independent

Platform Independent (Architecture Independent), Compressed TAR Archive (mysql-connector-java-8.0.27.tar.gz)	8.0.27	4.0M	<a href="#">Download</a>
MD5: 9243dc26efa3909b13517e002a89d7f5   <a href="#">Signature</a>			
Platform Independent (Architecture Independent), ZIP Archive (mysql-connector-java-8.0.27.zip)	8.0.27	4.8M	<a href="#">Download</a>
MD5: 3d054e96f5b78537cf53c4dd31f11eca   <a href="#">Signature</a>			

Pulsamos abajo, en “No thanks, just start my download”:

## MySQL Community Downloads

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL Discussion Forums
- Report and track bugs in the MySQL bug system

[Login »](#)  
using my Oracle Web account

[Sign Up »](#)  
for an Oracle Web account

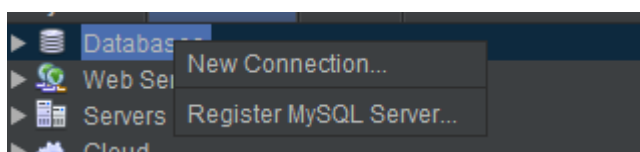
MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can signup for a free account by clicking the Sign Up link and following the instructions.

[No thanks, just start my download.](#)

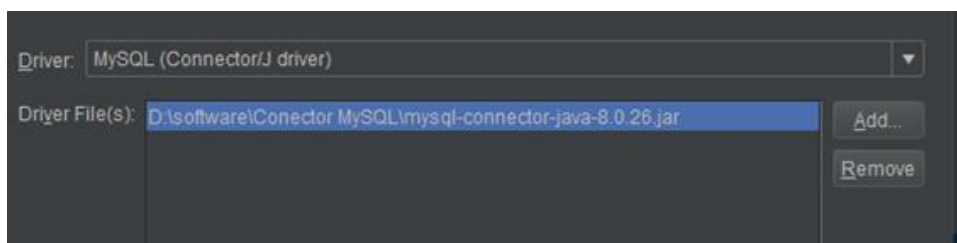
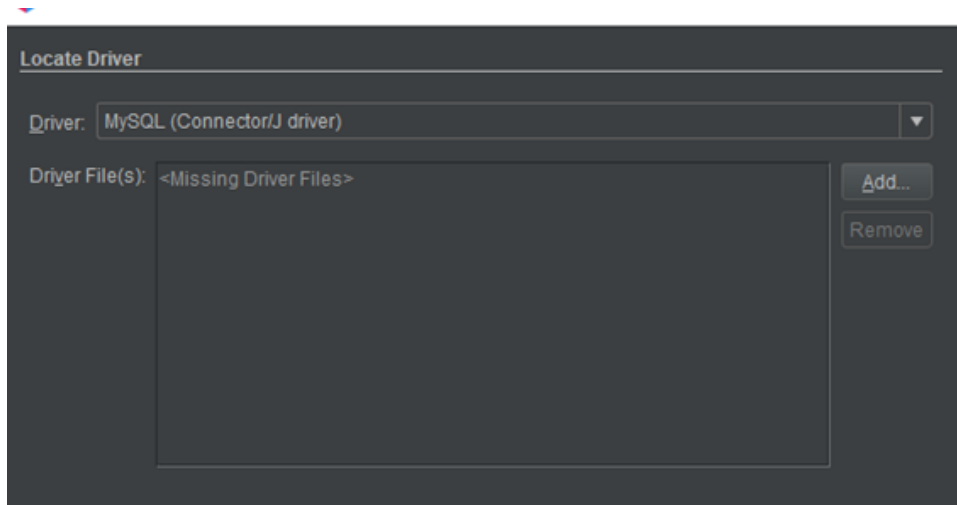
Dentro del .zip están estos archivos. El fichero llamado “mysql-connector-java-8.0.26” es el que necesitamos:

src	15/10/2021 12:30	Carpeta de archivos
build	15/10/2021 12:29	Documento XML
CHANGES	15/10/2021 12:29	Archivo
INFO_BIN	15/10/2021 12:29	Archivo
INFO_SRC	15/10/2021 12:29	Archivo
LICENSE	15/10/2021 12:29	Archivo
mysql-connector-java-8.0.26	15/10/2021 12:29	Executable Jar File
README	15/10/2021 12:29	Archivo

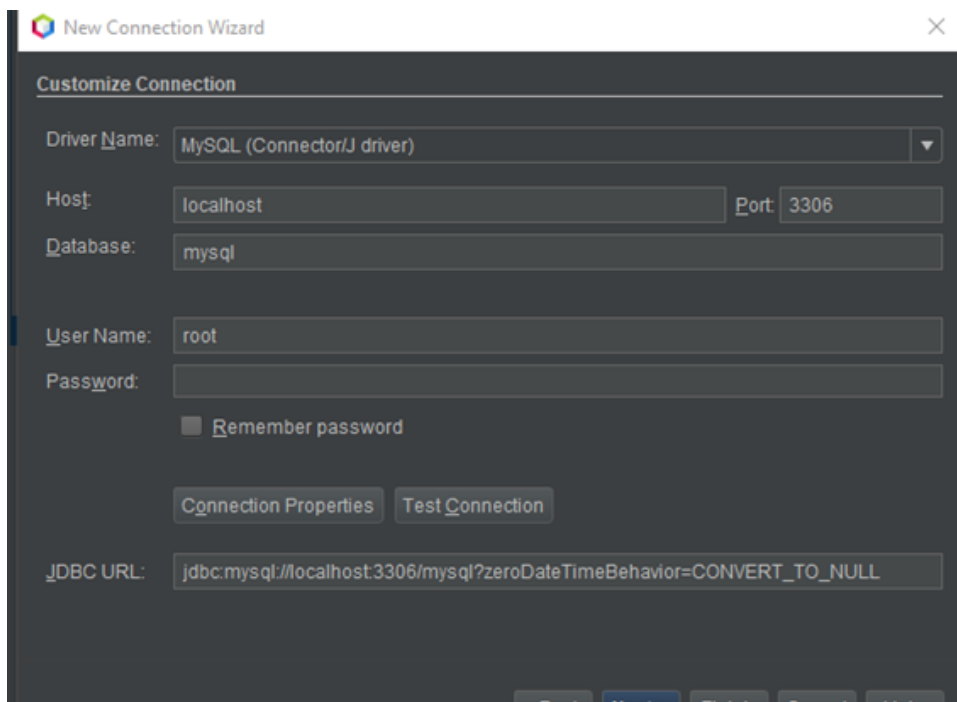
Ahora le tenemos que decir a NetBeans el sitio donde esta nuestro archivo.jar. Para ello, pulsamos sobre “Databases” con el botón secundario del ratón y le damos a “New Connection...”



Indicamos que el driver que se va a indicar es de MySQL y pulsamos el botón de “Add...” para indicar la ruta del archivo .jar.

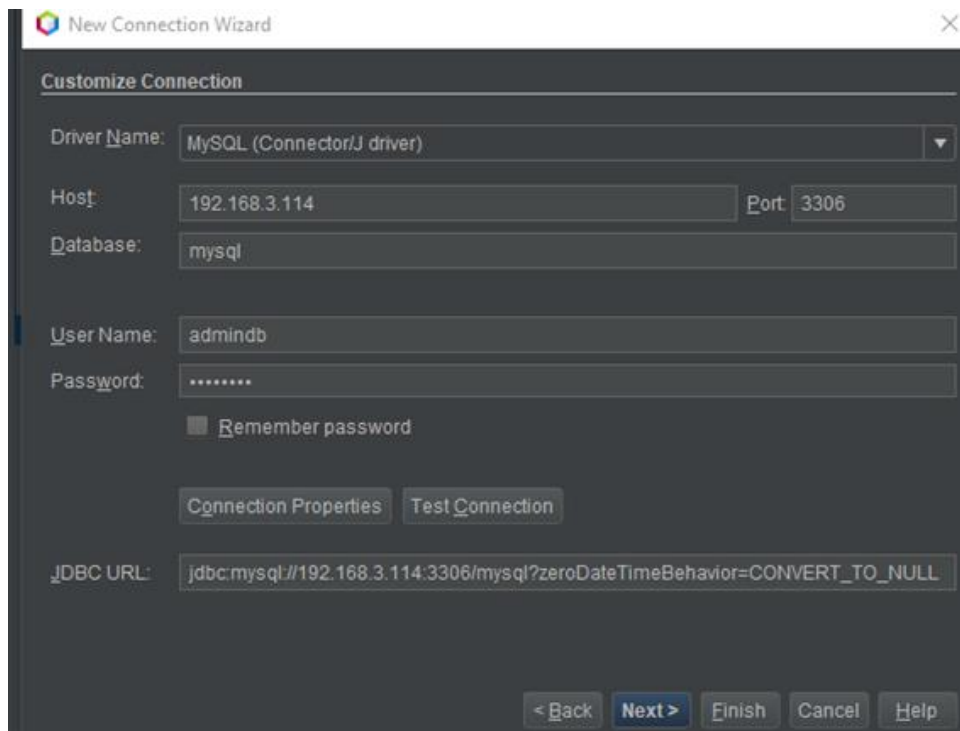


En la siguiente pantalla hay que indicar el servidor en el que se encuentra la base de datos con la que se quiere conectar y el nombre de la misma. También hay que loguearse como el usuario que va a administrarla. (NOTA: El usuario y la contraseña son de MySQL, no del Servidor Ubuntu). Lo ideal es que exista un usuario en MySQL específicamente para gestionar las bases de datos desde NetBeans:

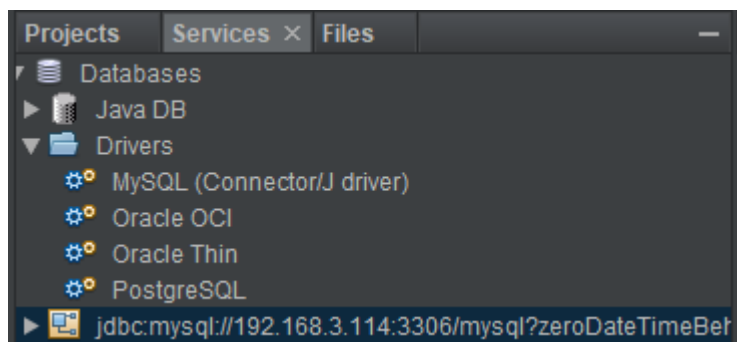


En este caso establecemos la conexión en la base de datos “mysql”, que aparece por defecto en el sistema y se utiliza para gestionar sus usuarios:





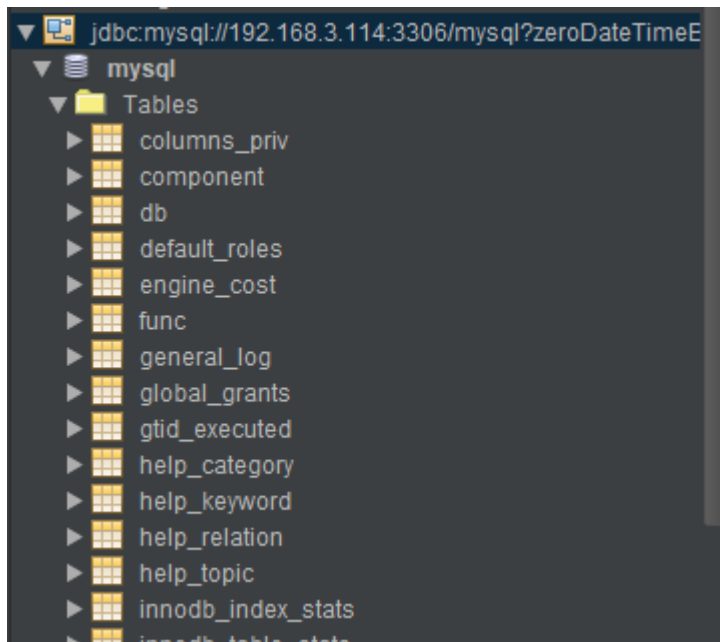
Ya le podemos dar a “Finish” y la conexión aparecerá en el apartado de “Databases”, dentro de Services



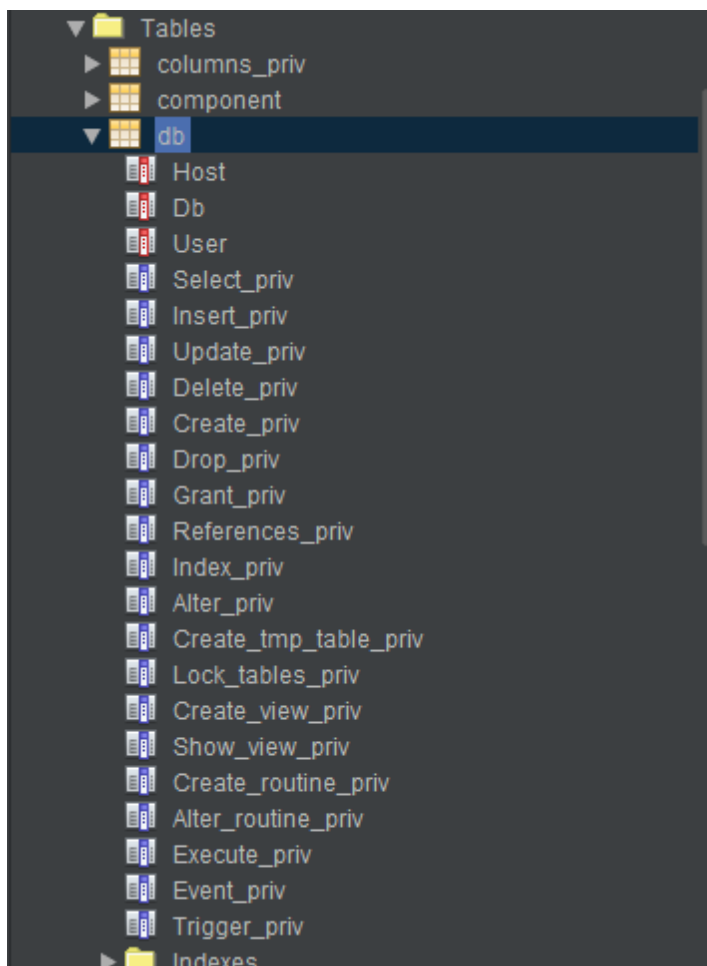
La conexión se puede deshabilitar y habilitar pulsando el botón secundario y seleccionando “Connect...” o “Disconnect”. Es importante asegurarse en cada sesión de que la conexión está habilitada:



Dentro de la conexión estarán la base de datos y todas sus tablas:

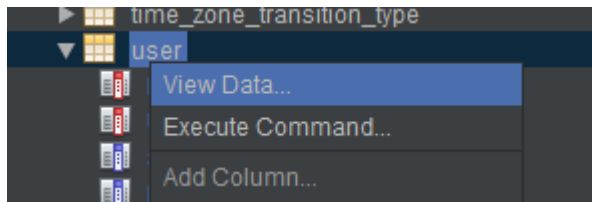


Dentro de las tablas también se pueden ver sus columnas:

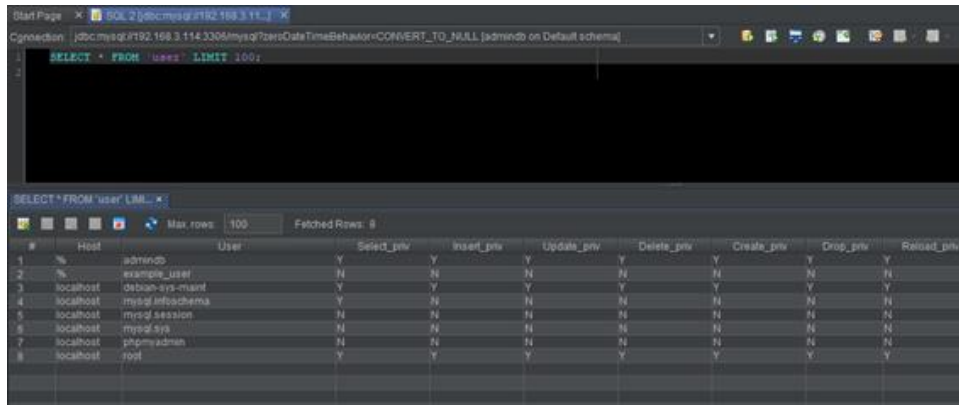


Para visualizar el contenido de una tabla se pulsa con el botón secundario en una tabla y en “View Data...”





Aparecerá el código que se utiliza para ver los datos y la tabla con toda la información debajo:



REVISIÓN A el 21/10/2021 por ÓSCAR LLAMAS PARRA

## 2. GitHub: administración de repositorios.

### 6.1. GitHub y los tokens de acceso.

Desde NetBeans, al tratar de sincronizar un repositorio, bien sea subiendo o bajando, por primera vez o después de unas veces, puede dar problemas la contraseña.

Bien dará un error en la parte inferior del diálogo si se especifica el repositorio:

Specify Git Repository Location:

Remote Name:  ☒ Persist Remote

Repository URL:

User:  (leave blank for anonymous access)

Password:  ☐ Save Password

! Incorrect credentials for repository at https://github.com/SashaMGuerra/proyectoC3PO.git

Bien continuará si ya tiene uno configurado, pero saldrá un diálogo para introducir la contraseña, que al cancelar da error:

Specify Git repository location

Repository URL:

User:  (leave blank for anonymous access)

Password:  ☐ Save Password

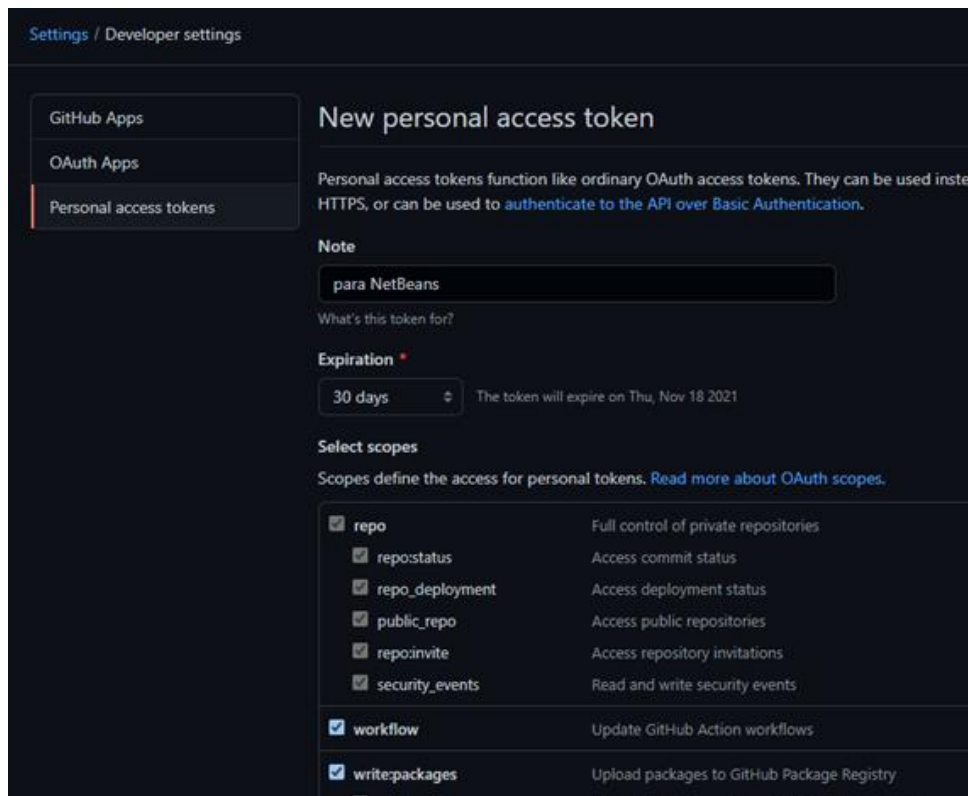
Git Command Failed

Git command returned with the following error:

```
https://SashaMGuerra@github.com/SashaMGuerra/proyectoC3PO.git: not authorized
```

No es suficiente con la contraseña de GitHub, se necesita generar un token de acceso personal.

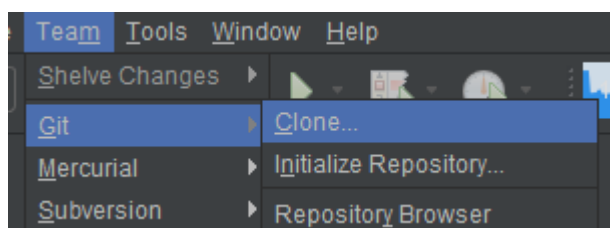
Iniciada sesión en github.com, se debe acceder a Settings > Developer Settings > Personal access tokens:



Dará una clave que solo mostrará esa vez, por lo que si se olvida, se deberá generar una nueva. Se utiliza como contraseña.

## 6.2. Clonar un repositorio.

Cuando se desea bajar un repositorio de GitHub desde NetBeans, se selecciona Team > Git > Clone:



En el diálogo, se le indica la dirección del repositorio junto a usuario y contraseña correspondiente, y la carpeta en que debe descargarse:

**Remote Repository**

Specify Git Repository Location:

Repository URL:  ▼  
http[s]://host.xz[:port]/path/to/repo.git/

User:  (leave blank for anonymous access)

Password:  ☒ Save Password

[Proxy Configuration...](#)

Specify Destination Folder:

Clone into:   [Browse...](#)  
(Leave empty to specify the destination later)

Pedirá seleccionar la rama a clonar, y a continuación, el directorio padre y el nombre que tendrá este clon:

Steps	Remote Branches
1. Remote Repository	
2. Remote Branches	Select Remote Branches
3. Destination Directory	<input checked="" type="checkbox"/> master*

**Destination Directory**

Specify the Parent Directory and Name for this Clone

Parent Directory:  [Browse...](#)

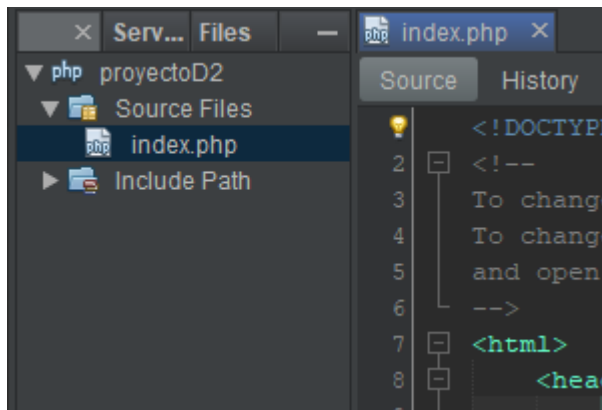
Clone Name:

Checkout Branch:  ▼

Remote Name:

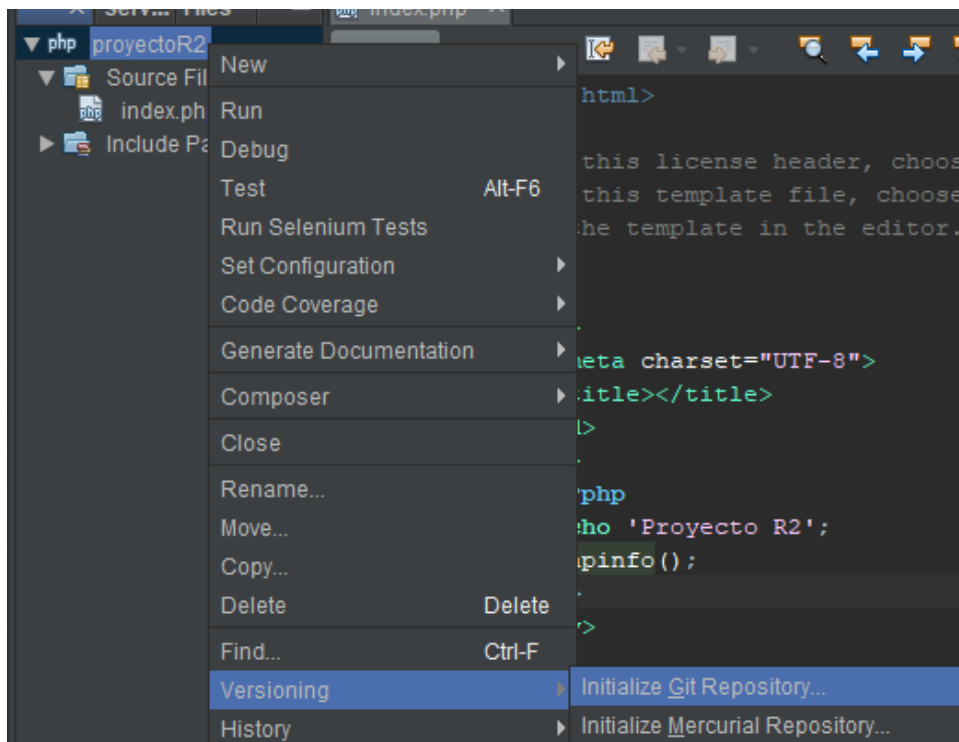
☒ Scan for NetBeans Projects after Clone

El proyecto puede abrirse en el mismo momento si se desea:

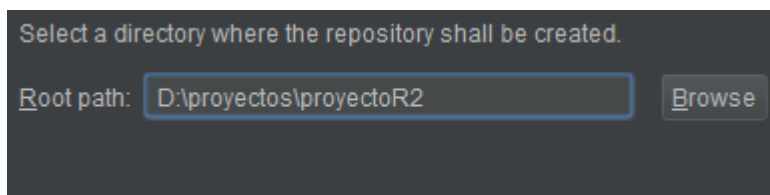


### 6.3. Inicializar un repositorio.

Para inicializar un repositorio desde GitHub, se hace clic, bien en Team > Intialize Repository o con clic inverso sobre el proyecto > Versioning > Initialize Git Repository:



Pedirá inmediatamente la carpeta en que se inicializará, que suele ser la misma que la del proyecto:

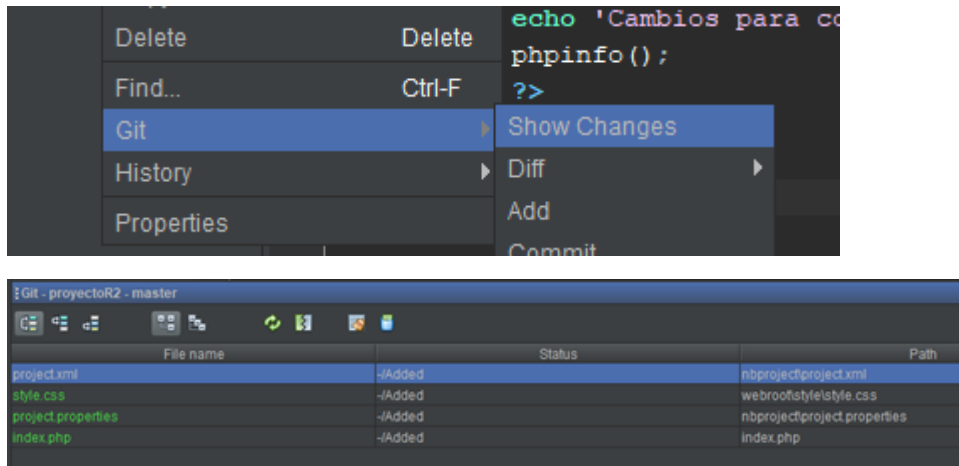


El repositorio ya está inicializado, pero aún no está subido a la nube.

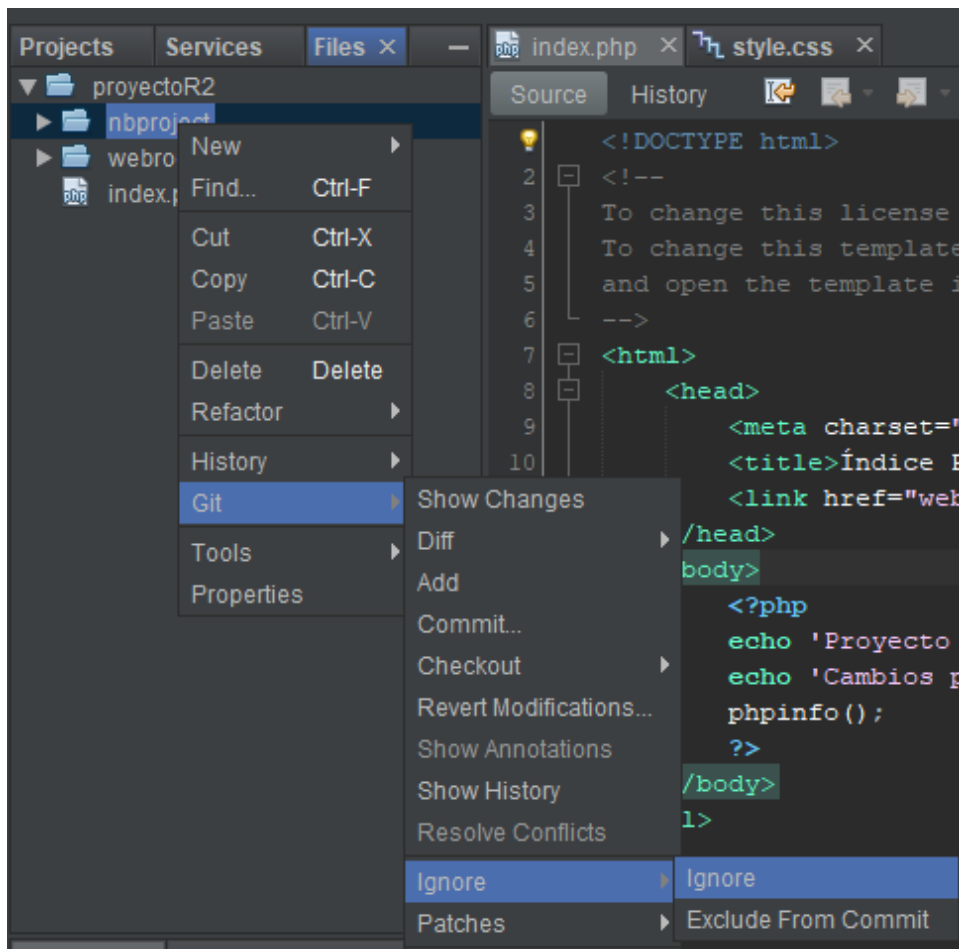
## 6.4. Guardar cambios y sincronizar el repositorio (*commit*, *push* y *pull*).

### 6.4.1. Hacer *commit*.

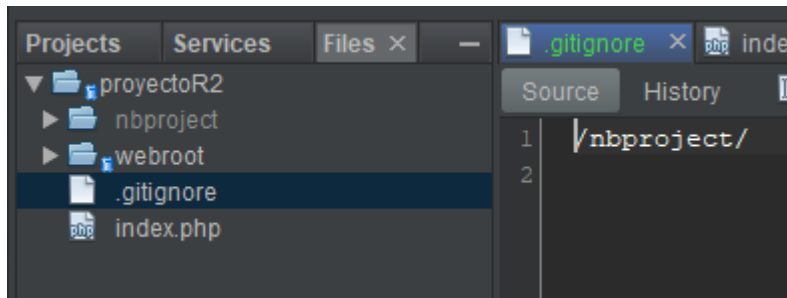
Los cambios que se han realizado hasta el momento se pueden comprobar mediante Git > Show Changes:



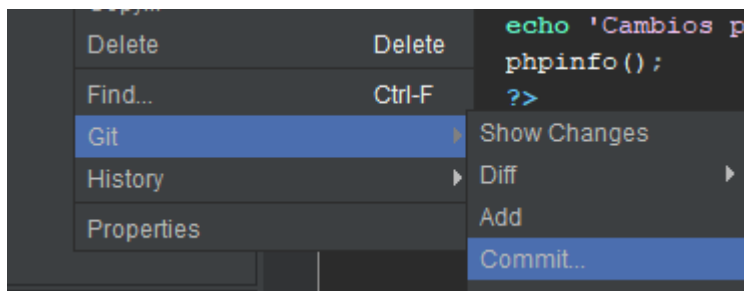
Si hay algún archivo o carpeta no quiere ser tomada en cuenta por el control de cambios, debe añadirse al archivo .gitignore. Aunque el archivo no se crea por defecto al inicializar el repositorio, sí se crea automáticamente si se le indica que un archivo debe ignorarse:



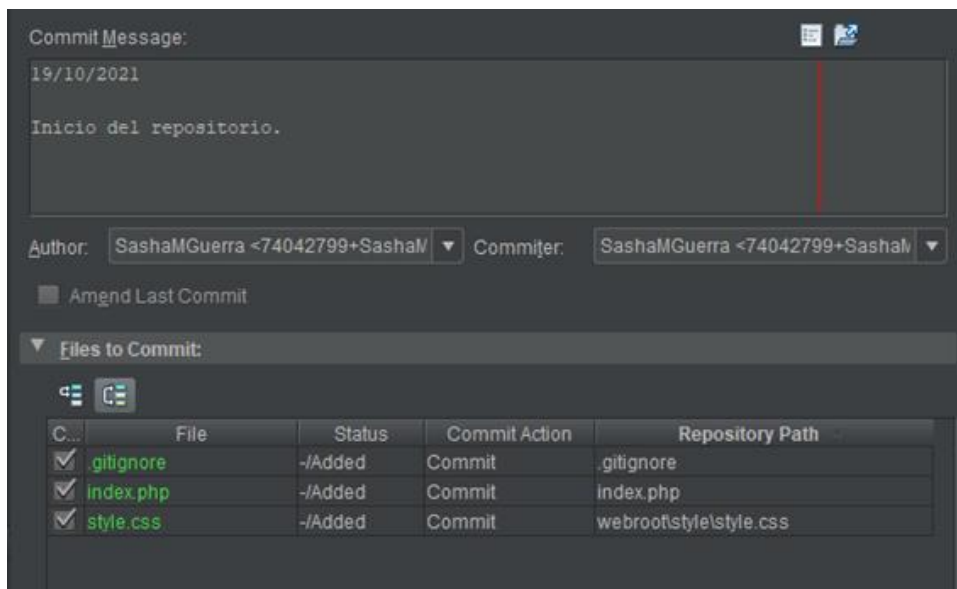
En este caso se ignorará la carpeta que hace de este directorio un proyecto NetBeans:



Los cambios se guardan constantemente mediante commits. Para hacerlo, se hace clic inverso en el proyecto y se selecciona Git > Commit:

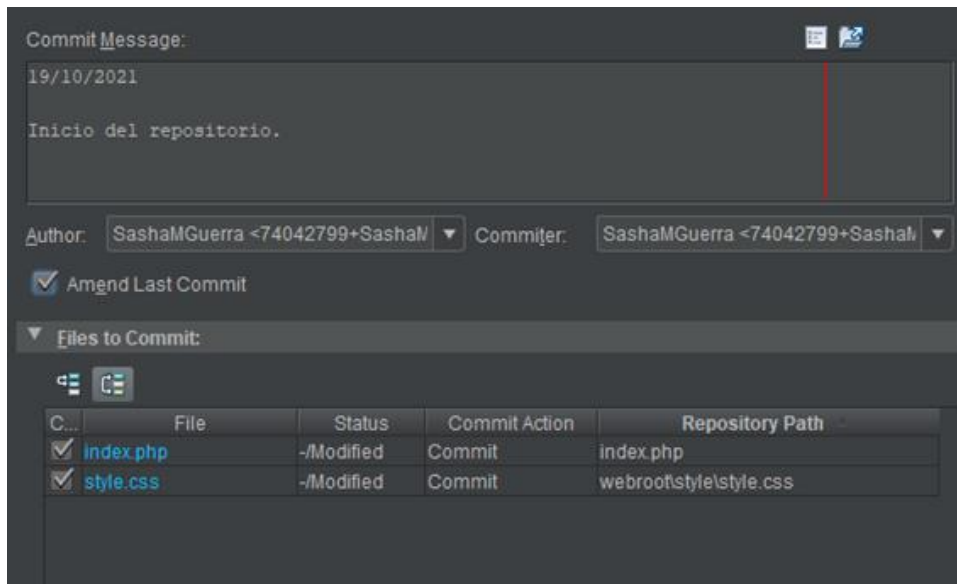


En el diálogo de commit permite escribir un mensaje. En éste, la primera línea será el título del commit, y lo demás su descripción. Permite elegir los modificados archivos a incluir:



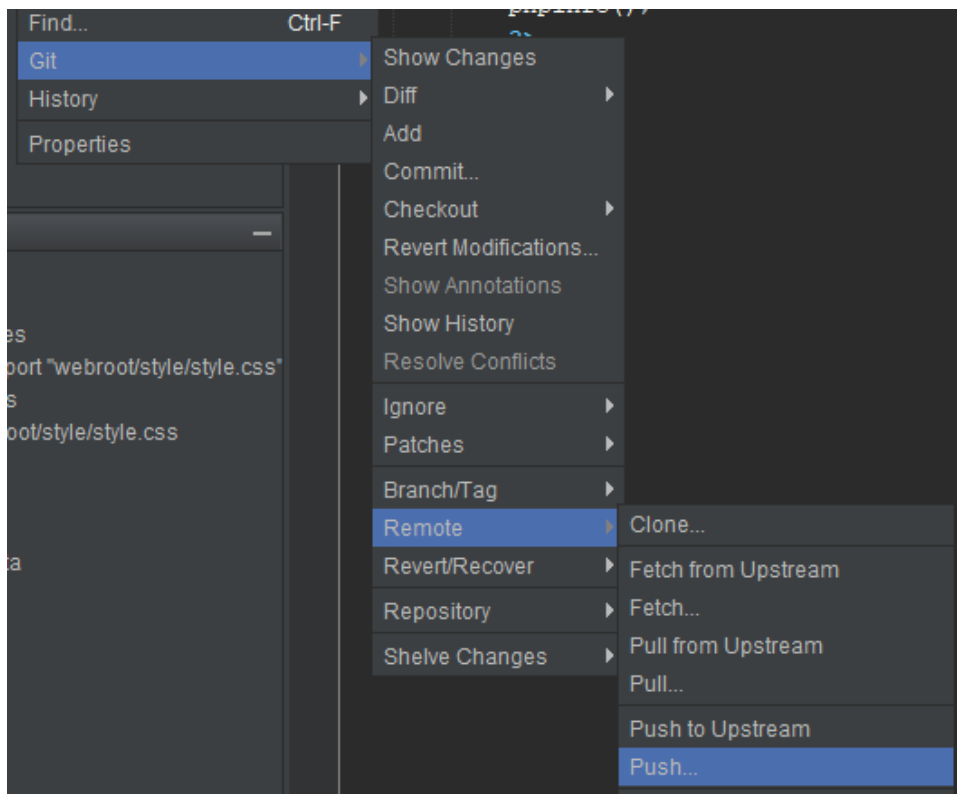
#### *Amend last commit.*

Esta opción permite modificar el commit anterior, de modo que los nuevos cambios desde el último commit no se guarden como un nuevo commit, sino en el mismo anterior:



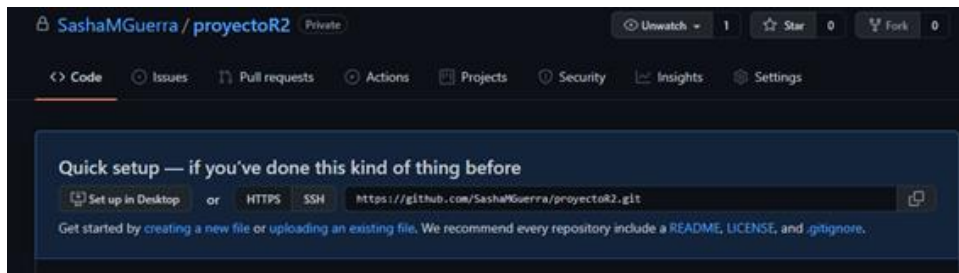
#### 6.4.2. Subir commits: *push*.

Ya con commits que subir, se sube mediante push:



Para poder hacerlo, el repositorio debe existir ya en GitHub, aunque vacío:

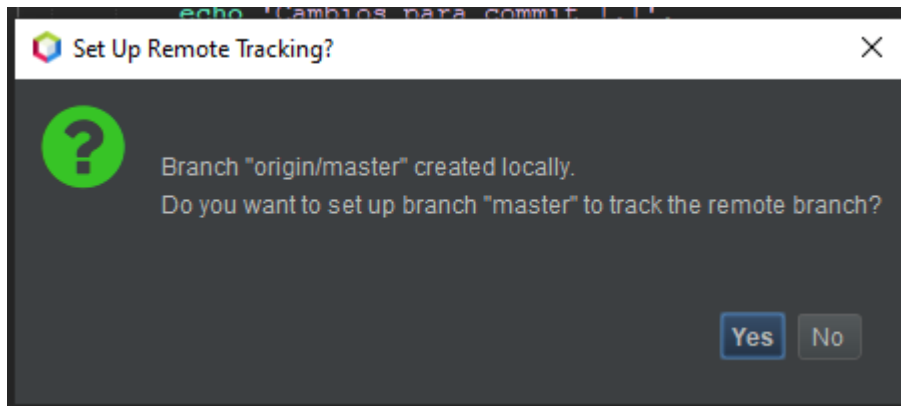




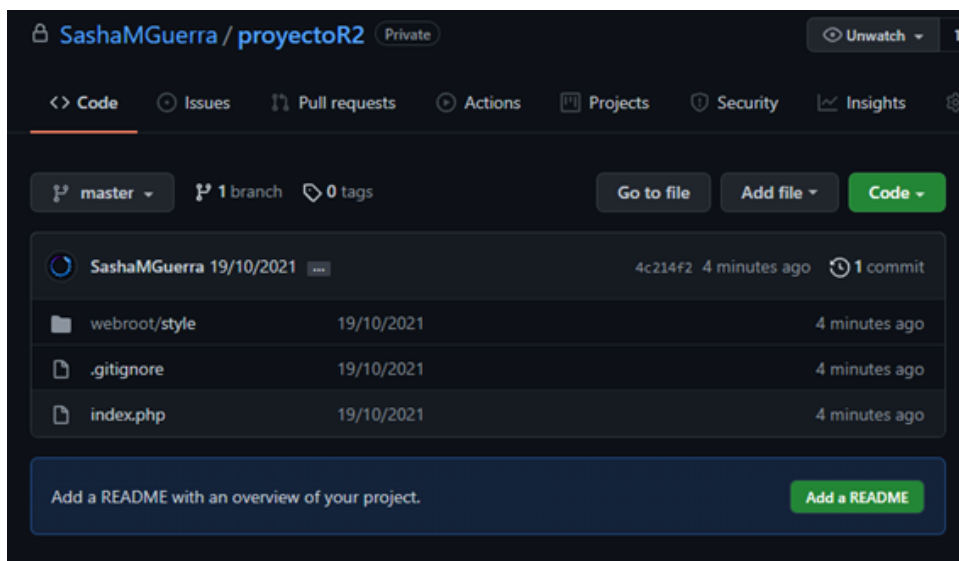
Esa dirección es la que debe indicarse en el URL del repositorio:

Se seleccionan las ramas local a subir y remota a actualizar:

Dado que la rama remota no estaba configurada, propondrá hacerlo. Se acepta:

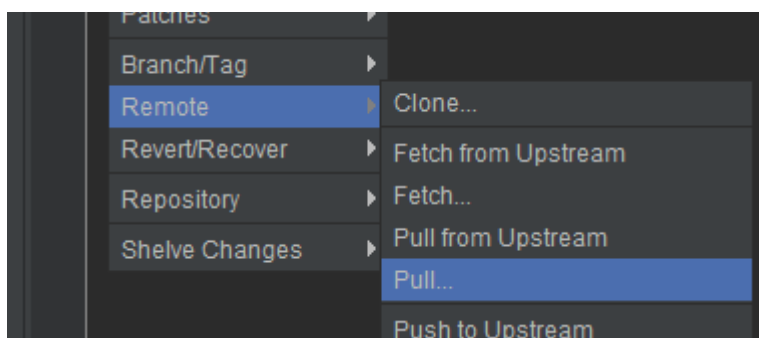


Al recargar la página del repositorio en GitHub ya debería tener subido el commit:

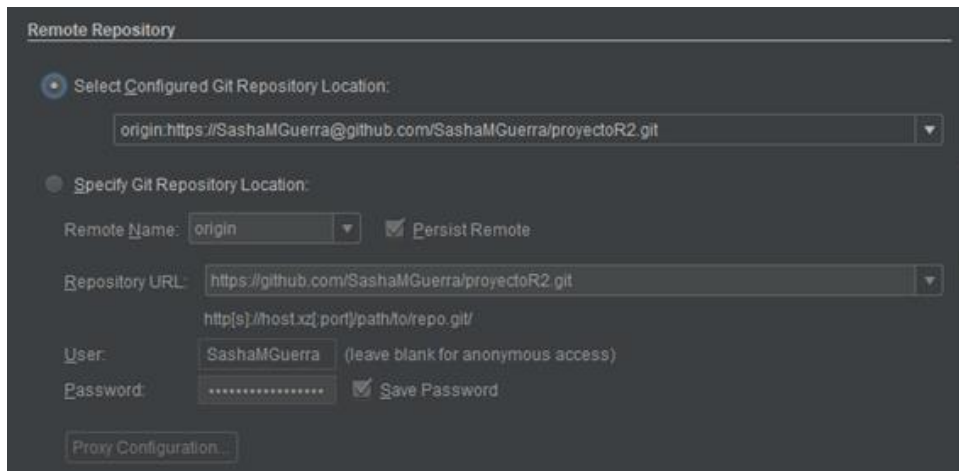


#### 6.4.3. Bajar commits: *pull*.

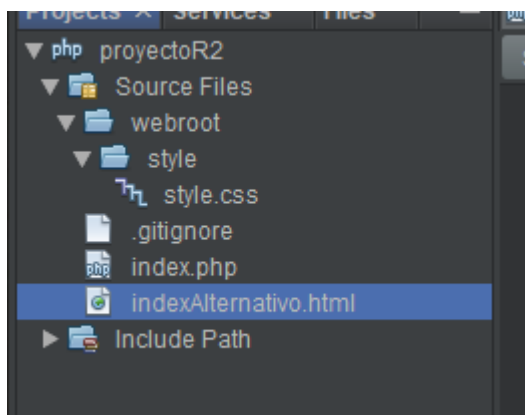
Si se ha realizado algún cambio desde otro lugar, se puede bajar mediante pull:



Del mismo modo que con el push, se indica el repositorio si no está todavía y las ramas a actualizar:



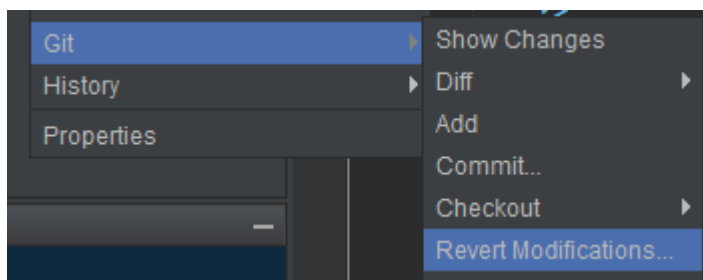
Se comprueba que haya funcionado:



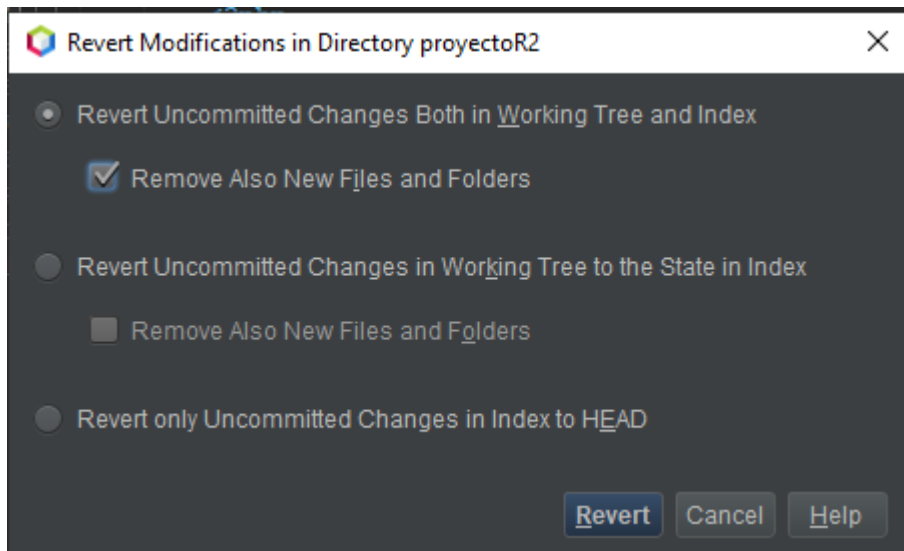
#### 6.4.4. Deshacer cambios y commits.

##### *Deshacer cambios.*

Cuando se han realizado cambios pero no se ha hecho commit de ellos, se pueden revertir al origen del último commit mediante Git > Revert modifications:

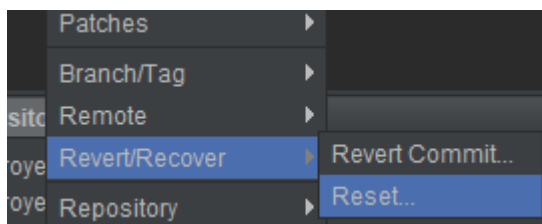


Ofrece tres posibilidades, según lo que se quiera revertir:

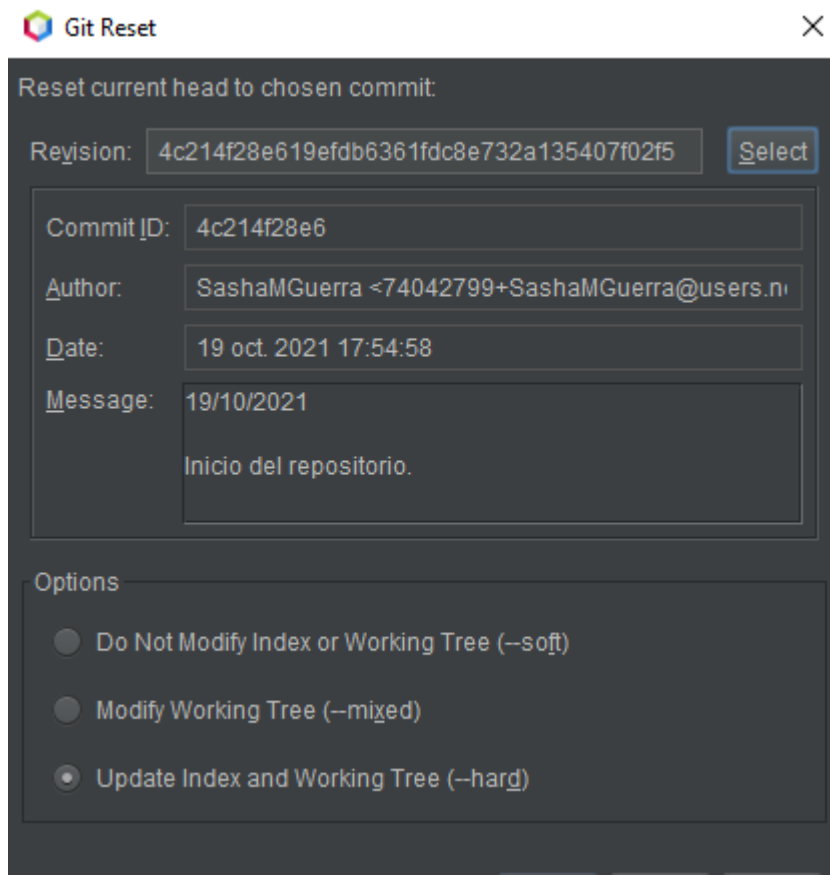


*Deshacer commits.*

Se elige la opción Revert/Recover > Reset:

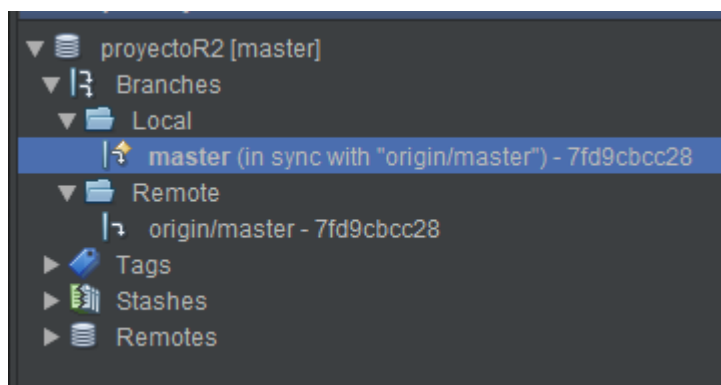
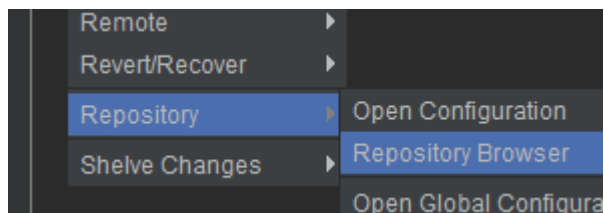


Se elige el commit al que quiere regresarse y una de las opciones respecto a cambios en el index y el árbol de trabajo:



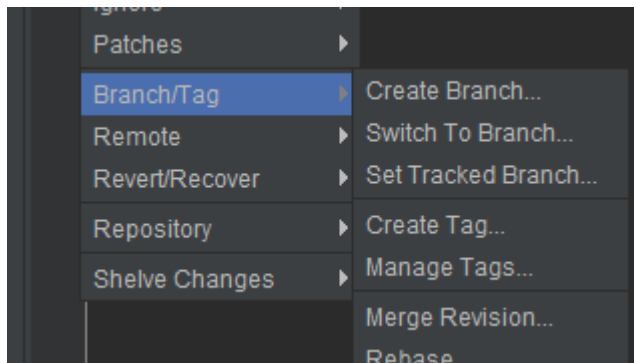
## 6.5. Creación y uso de ramas de trabajo (*branches*).

Se pueden ver las branches del repositorio en Git > Repository > Repository Browser:

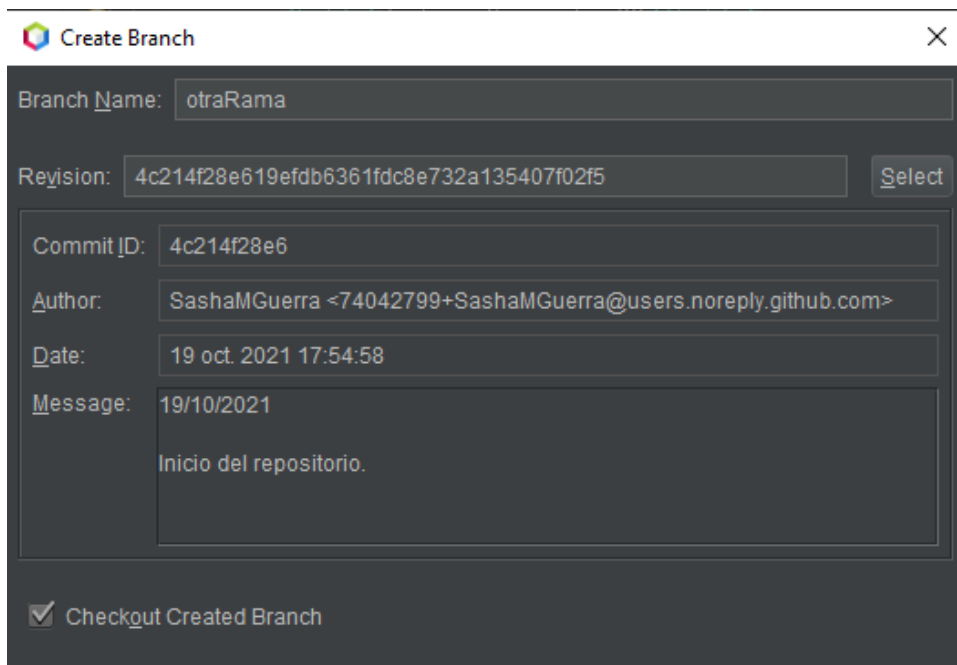
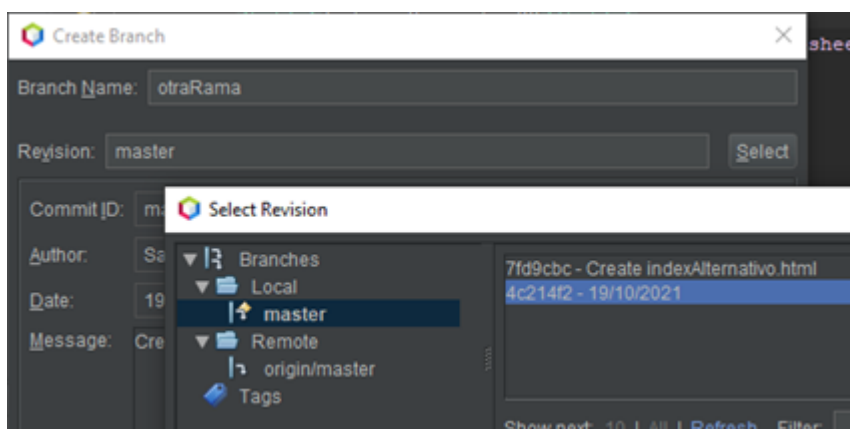


### 6.5.1. Crear ramas.

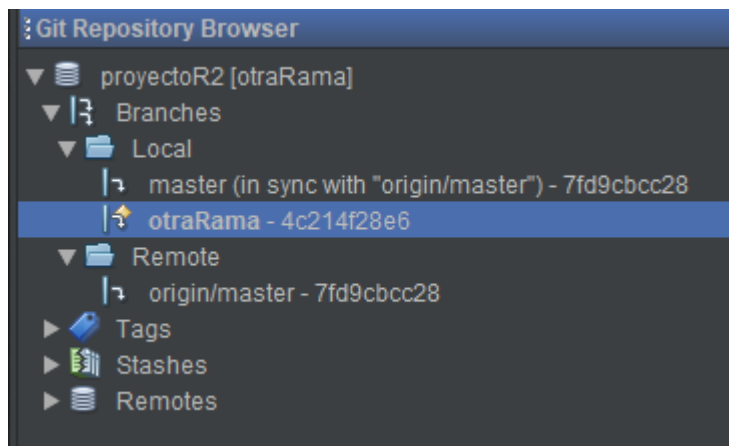
Se crean mediante Git > Branch/Tag > Create Branch:



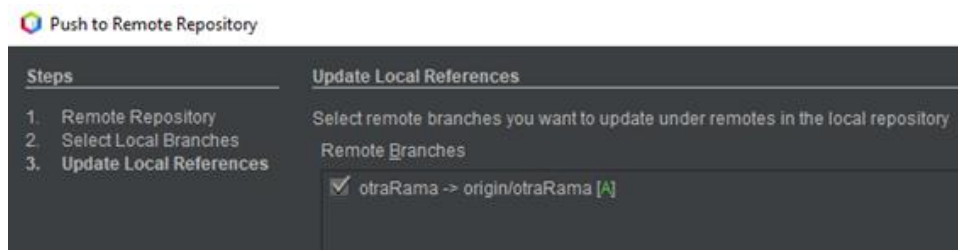
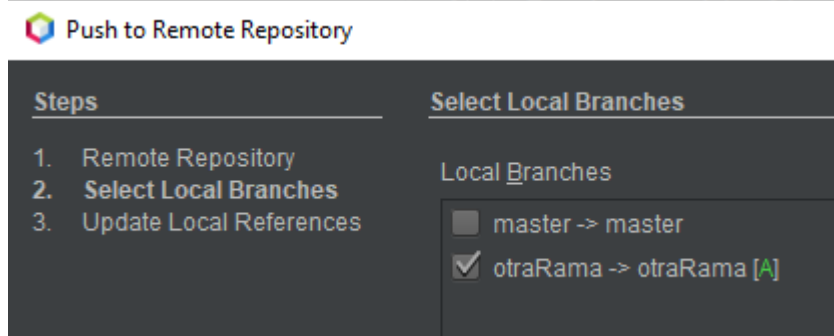
Se le deberá dar un nombre y el commit desde el que se crea. En este caso se hará, no desde este segundo commit que se ha bajado sino en el primero; para eso, se selecciona *Select* en la revisión, y se elige:



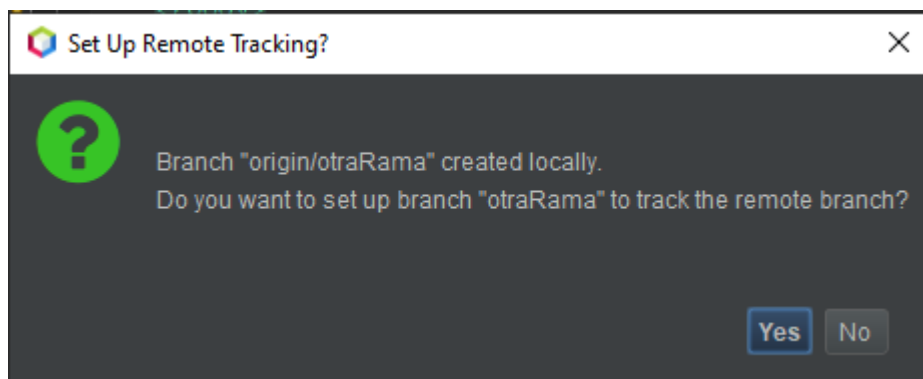
Se ha creado la rama, pero solo está en local:

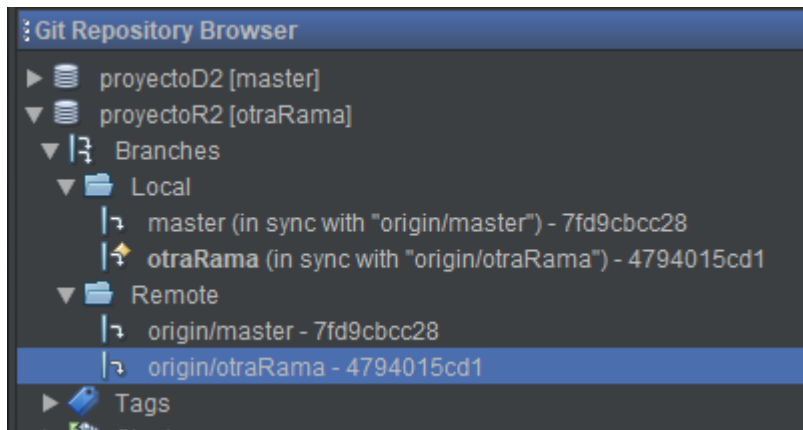


Para subirla a remoto, se necesitará hacer un commit en la nueva rama. Al hacer push se podrá elegir la nueva rama:



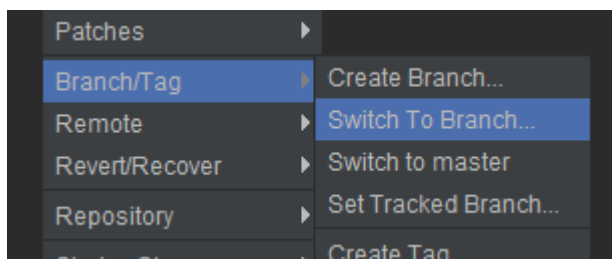
Dado que no existe en remoto, se podrá elegir crearse:





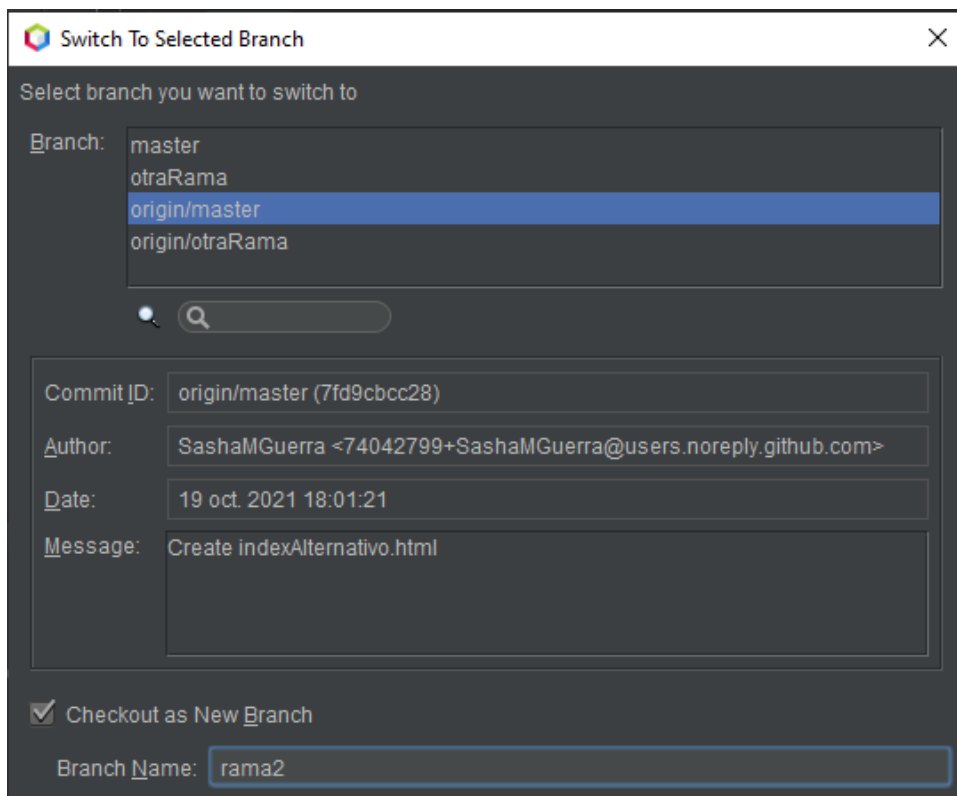
### 6.5.2. Cambiar entre ramas.

Para cambiar a una rama directamente, se selecciona Git > Switch to Branch



Da como opciones todas las ramas locales y remotas. Además, al final del diálogo ofrece la posibilidad de crear una nueva a partir de la elegida.

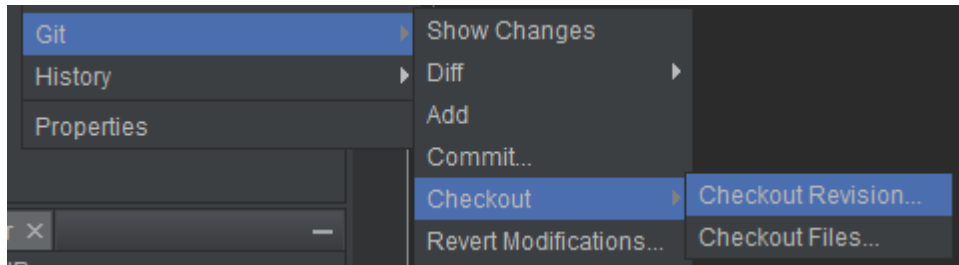
En este caso se creará una tercera rama a partir de la rama master en remoto:



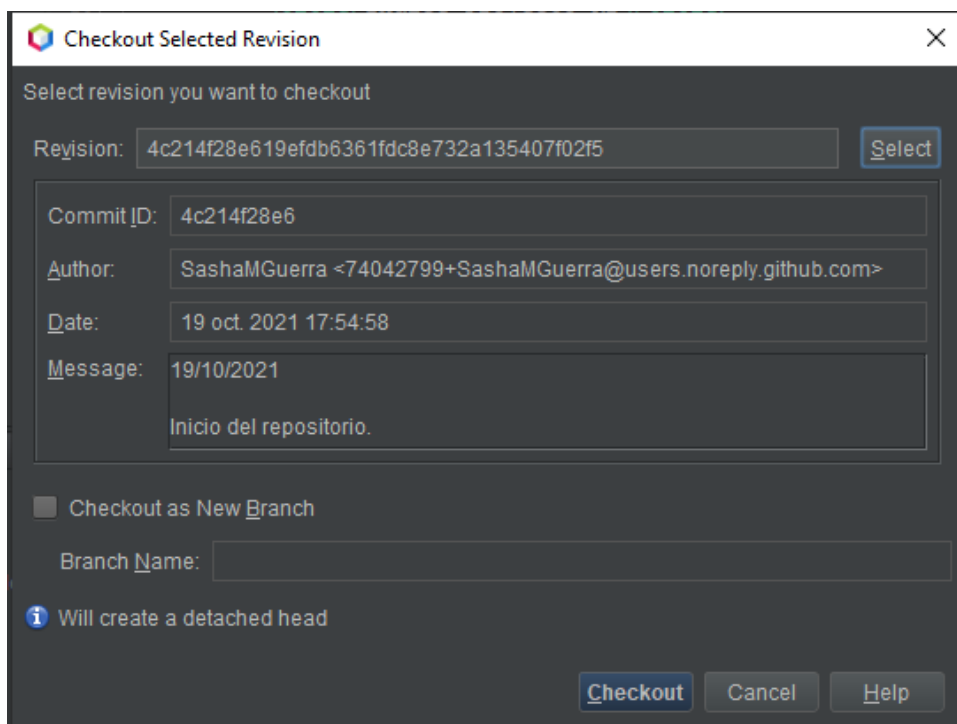


*Pasar a una revisión concreta de una rama.*

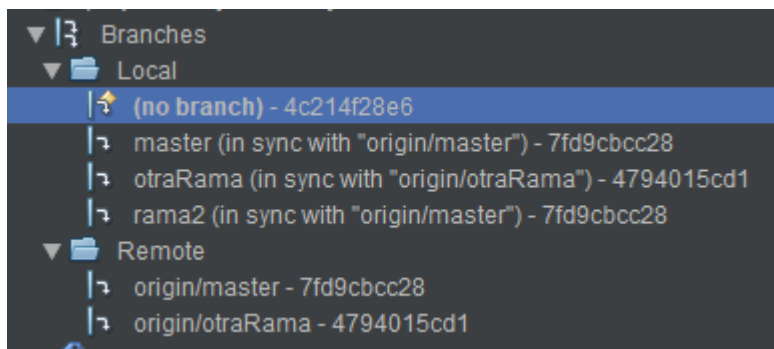
Sea de la misma rama o de otra, se selecciona Checkout revision en  
Git > Checkout > Checkout Revision



Como ejemplo, se va a volver al inicio del repositorio, al primer commit de la rama master:

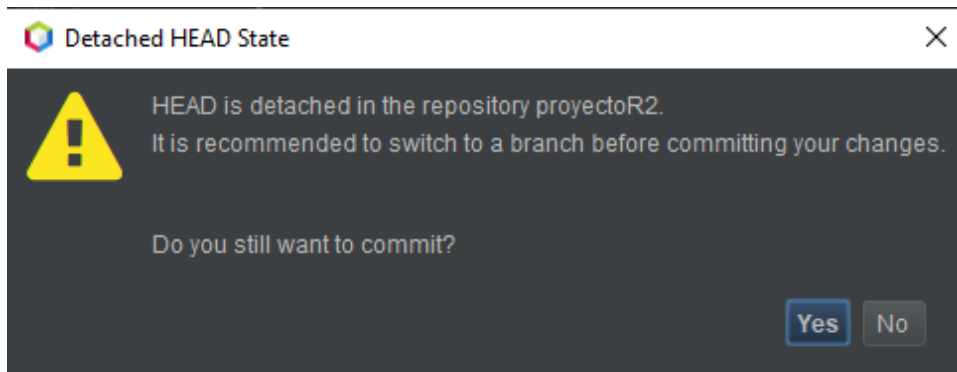


Al no crearse una nueva rama a partir de este commit, ha generado una “cabeza” (head) sin rama:

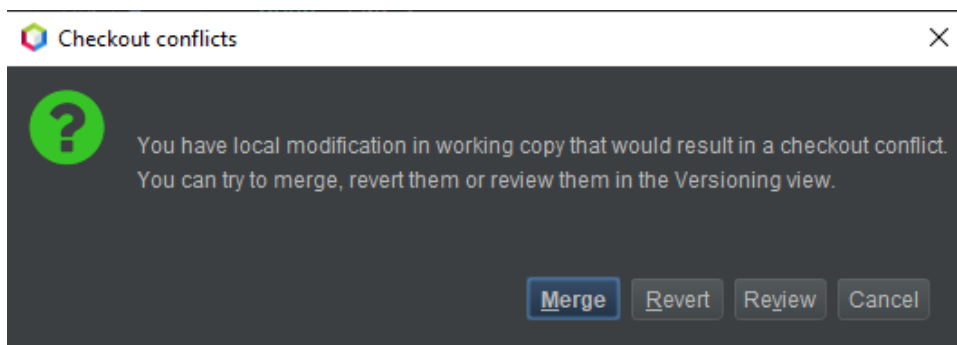


### Detached heads.

Aunque desde estas no-ramas se pueden hacer commits, salta una advertencia, y al hacer push requiere una rama verdadera.



Además, cuando se quiere cambiar de rama, si esta no se ha convertido en una y no se han guardado los cambios, deberá indicarse qué hacer con ellos, pues la no-rama se borra:



### 6.5.3. Eliminar ramas.

Simplemente, se elige la rama y la opción para eliminarla. Si la opción no se puede seleccionar se debe a que hay cambios sin guardar:

