### MP\_0478

#### UF4

##### Actividad 8

### Entornos de desarrollo

#### Control de versiones\_

[Tarea en equipo] Control de versiones con GIT y un IDE

Autores: Mikolaj Karol Bernecki,Margarita Manzano López,Francisco José Castuera Martín

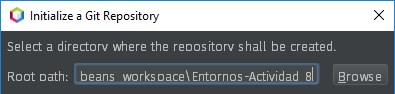
# Índice de contenidos

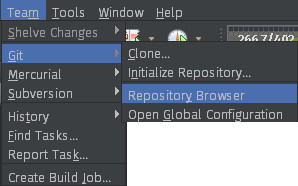
Índice de contenidos 2

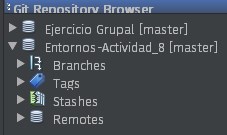
Conclusiones 4

Bibliografía 4

# Creación del repositorio local

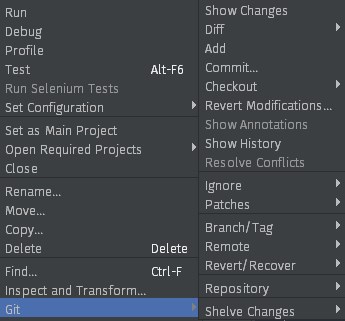
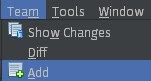
Creamos un proyecto en netbeans. Será una aplicación para crear una factura para un cliente. Después de rellenar ficheros con el borrador de nuestra aplicación, creamos un repositorio local con Egit dentro de netbeans, haciendo Click-dcha en el nombre de proyecto y elegimos la opción Versioning/Initialize Git repository. Nos sale una pantalla con dirección donde queremos guardar el repositorio. Si todo lo hemos hecho bien, no hace falta cambiar nada. Sólo hay que comprobar, que la dirección apunta a la carpeta donde guardamos nuestro proyecto.

Luego, comprobamos en Repository Browser, entrando en la opción Team/Repository Browser. Nos sale un navegador de repositorios, donde miramos si sale el nombre de nuestro proyecto.



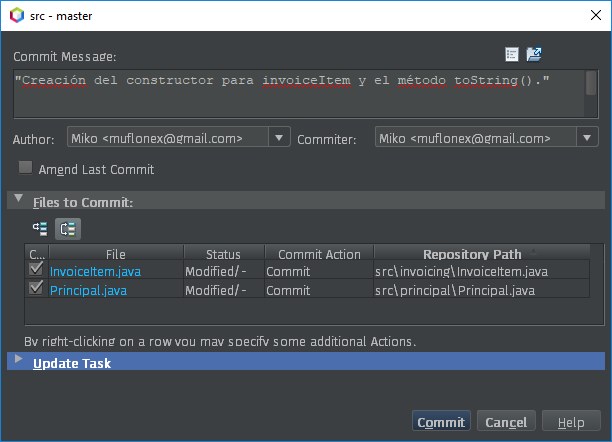
# Realizar algún cambio en el proyecto y añadirlo al repositorio local

Realizar un cambio es tan fácil como escribir algo en cualquier de los ficheros de nuestro proyecto dentro del IDE y guardarlo. Para añadir los cambios al repositorio existen tres maneras.



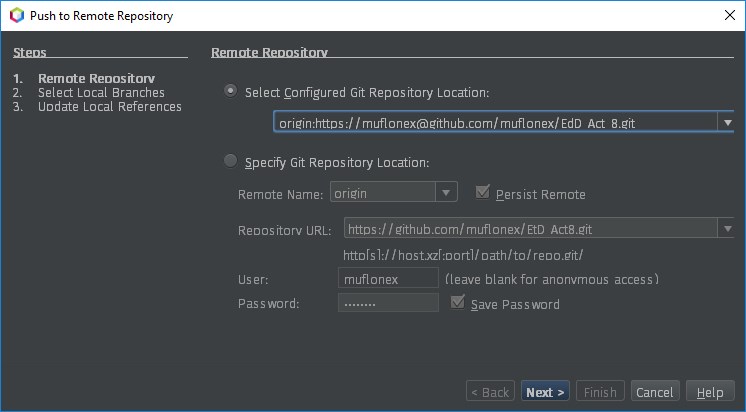
Primero, tenemos que añadir cambios al index con Add. Una manera de hacer esto es utilizar el menú de arriba y usar la opción Team/Add. También podemos hacer Click-dcha sobre el nombre del proyecto y elegir opción Git/Add. La opción para más atrevidos es crear un atajo en ajustes de netbeans en Tools/Options/Keymap.



Luego, podemos hacer el commit en mismos lugares (Menú de arriba —> Team o Proyectos —> Git), utilizando la opción Commit. Lo cual nos abre la ventana de insertar el comentario y el resumen de archivos cambiados.

# Subir los archivos al repositorio remoto

Para poder subir nuestros archivos a algún almacén compartido, necesitamos establecerlo. En nuestro caso será GitHub. Con una cuenta hecha, solo nos hace falta crear allí un repositorio, pulsando el icono + y New repository. Se abre nueva ventana donde le damos nombre al repositorio y pulsamos Create repository para crearlo.



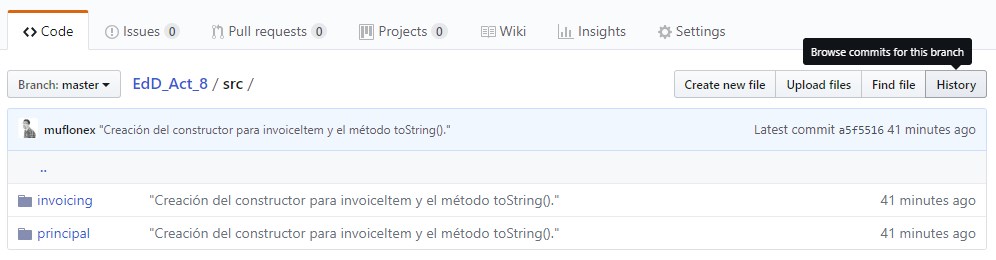
Al crearlo recibimos su enlace con pistas como añadir la referencia al repositorio remoto dentro de uno local. En nuestro caso lo vamos a hacer con el IDE, con Menú —> Team/Remote/Push o Click-dcha en Proyecto —> Git/Remote/Push.

Se abre la ventana donde especificamos el nombre de la referencia y su enlace. En siguientes pushes ya podemos utilizar el configurado. También se puede configurar bajo Team/Repository/Open Configuration.

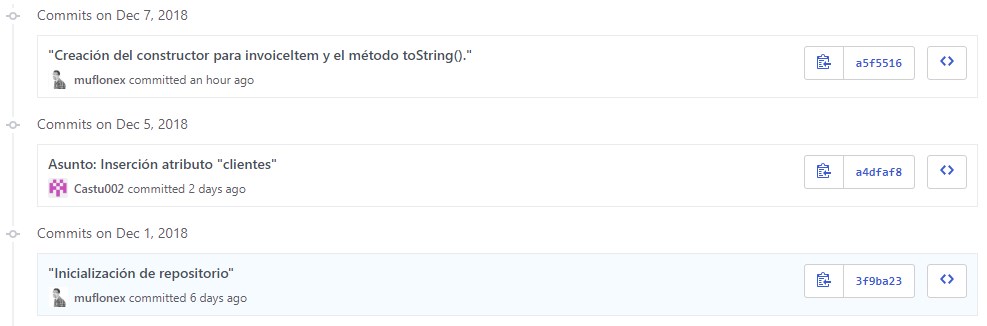
No es necesario llamarlo origin, pero es la convención que se suele utilizar.

# Comprobar el estado del repositorio remoto

Para ver los cambios dentro del repositorio remoto, necesitamos ir a la página del proyecto en GitHub, los comentarios de commits se heredan de versión local. Para navegar entre los archivos viene útil pulsar en la esquina superior izquierda que nos abre el menú lateral. Si queremos ver el historial de cambios, tenemos que entrar en la carpeta que nos interesa y pulsar el botón History.

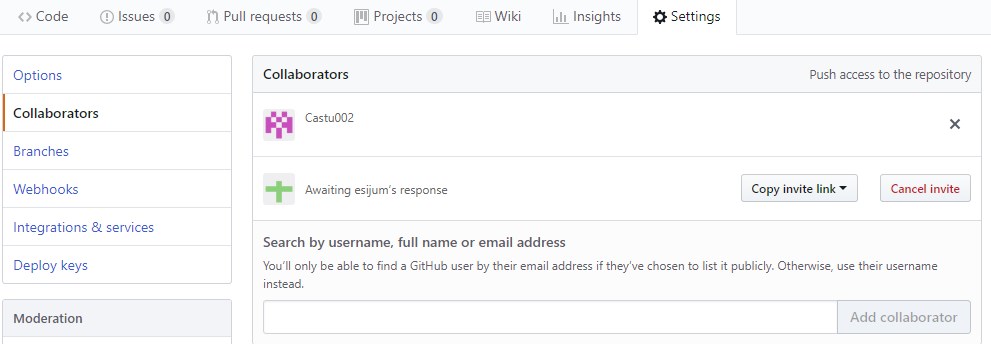


Nos sale el historial de commits hechos por todos. Podemos ver identificadores de cada uno, pulsando al <> podemos ver el estado de repositorio en aquel momento cuando se realizó el commit. Y pulsando al ícono de agenda podemos ver los cambios que se ha introducido archivo por archivo.



# Dar acceso a otros usuarios

Una cosa imporante: obviamente para poder cooperar en mismo repositorio con los demás, sin tener que manejar pull requests, tenemos una opción de añadir colaboradores. Tenemos que pulsar al Settings —> Collaborators para acceder a los ajustes del repositorio remoto.



Allí podemos buscar nombres de usuarios de GitHub para encontrar a nuestros compañeros.

# Conclusiones

# Enlaces

Repositorio remoto: <https://github.com/muflonex/EdD_Act_8>