GIT y GITHUB

Instalamos GIT desde git-csm.com. Buscamos la última versión para el sistema operativo correspondiente.

Con el Git Bash. O sea usamos el GIT con línea de comandos.

Para poder instalar GITHUB tenemos que hacernos una clave SSH, que es el modo de establecer una conexión segura entre nuestro ordenador y el github en la nube.

Es un proceso sencillo

Comprobamos si hay alguna clave ya en nuestro ordenador

**cd ~/.ssh**

Si nos acepta significa que ya tenemos alguna otra clave.

Si no hubiera ninguna clave aparecería un mensaje que no se ha encontrado tal archivo o directorio.

Por lo tanto en cualquiera de los dos casos . Tenemos que hacer lo siguiente, que si ya existiera alguna clave va a hacer un backup de las claves existentes y eliminarlas para poder crear unas nuevas

Entonces

**ls** (nos va a mostrar los archivos que se encuentran pueden ser key\_backup y known\_hosts)

Vamos a crear un directorio

**mkdir key\_backup**

Si ya esta creado nos va a decir que esta creado.

Ahora vamos a copiar las claves al directorio key\_backup

**cp id\_rsa\* key\_backup**

Si no los encuentra nos avisara.

Ahora borramos las claves existentes una vez copiadas.

**rm id\_rsa\***

Si no los encuentra nos avisara.

AHORA CREAMOS NUESTRA NUEVA CLAVE SSH

**ssh-keygen –t rsa –C “**[**guille971172@gmail.com**](mailto:guille971172@gmail.com)**”**

Que es la dirección de email que vamos a utilizar tanto en GIT como en GITHUB.

Luego nos dice que se genera un par de llaves publicas y privadas y no indica donde queremos que se guarde.

Le damos enter.

Luego escribimos un password y lo repetimos, no te aparecen los caracteres. OJO.

Luego no indica que nuestra identificación ha sido guardada.

Y nos aparece la clave en un código.

Luego tenemos que abrir el archivo publico para pasarle los datos a GITHUB para poder enlazar GIT en nuestro ordenador con GITHUB en la nube.

Creamos una cuenta en GITHUB

Usuario : WilliVal

Email : [guille971172@gmail.com](mailto:guille971172@gmail.com)

Contraseña : Nautica1979

Dentro de GIT HUB en Setting / SSH Keys

Agregamos una clave en New SSH Key

Y el ámbito solo en repo.

Le colocamos el nombre de la clave TITULO ---Mi clave

En KEY copiamos el contenido de el archivo id\_rsa.pub .

Y generamos o añadimos la clave.

Nos aseguramos de copiar el nuevo token en un archivo txt , porque no lo vamos a ver mas.

Ahora tenemos que comprobar en nuestro GIT que lo hemos hecho todo bien y que la clave publica SSH que hemos introducido funciona.

Asi que en la consola GIT colocamos

ssh [git@github.com](mailto:git@github.com) (dirección de email , que es la dirección que nos pone en contacto con GITHUB)

Esperamos que se ponga en contacto, y nos pide la contraseña que hemos puesto en GIT.

Las contraseñas de GIT y GITHUB no tienen porque ser iguales.

Y observamos que nos que WilliVall se autentificado del modo correcto pero de momento no tenemos el Shell Access, esto es normal. ES CORRECTO.  
 Ya tenemos configurado GIT y en GITHUB tenemos nuestra clave SSH.

Ahora tenemos que configurar nuestra información personal en GIT.

Que es lo que vamos hacer ahora.

GIT sigue la pista de cada uno de los commit que se hagan por lo tanto, cada commit necesita un nombre de usuario y un email del responsable y además vamos a utilizar esa información para asociar nuestros commit con nuesta cuenta en GITHUB.

Para hacerlo vamos a tener que escribir el siguiente código .

El nombre que tenemos que escribir ahora debe ser nuestro nombre real , el que usemos en el grupo de desarrollo en el que estamos trabajando.

git config –global user.name “Vallejos Guillermo”

git config –global user.email [guille971172@gmail.com](mailto:guille971172@gmail.com)

Con esto tenemos configurado en GIT nombre de usuario y el email del usuario.

45:57 Curso de Git y Github. Jesus Conde

Tenemos entonces configurado lo fundamental de GIT y el enlace entre GIT y GITHUB pero hay herramientas en GIT que conectan a GITHUB sin usar SSH y para que esto nos funcione del modo apropiado vamos a tener que configurar el TOKEN de la API de GITHUB

Buscamos el token copiado anteriormente que es un ASH y tenemos que volve a la consola e indicarlo.

git config –global github.user WilliVal

git config –global github.token  a6246d9682cd05fceacd5fcefb06c4f599fbe789

Este cambia si cambiamos la contraseña. Asi que de nuevo es una encriptación , un ahs que utiliza GIT.

Ahora ya tenemos en GIT la información de nuestro nombre de usuario en GITHUB y nuestro token que es la contraseña encriptada para poder acceder.

Ya tenemos tanto GIT Y GITHUB configurados para que trabajen en equipo.

Ahora si no tenemos nuestro perfil publico , lo generamos en GITHUB y luego creamos un repositorio.

Como hacemos que GIT y GITHUB trabajen en equipo, porque recuerda que GIT esta en nuestro ordenador y GITHUB en la nube.

Vamos a crear un archivo README en el repositorio, ya que es necesario para pasar información y documentación para los usuarios.

Vamos a crearlo en la consola de GIT.

Algunos comandos ya lo hicimos pero podemos repetirlo sin ningún problema.

Creamos un repositorio en GITHUB llamado Jobs-Java

git config –global user.name “Vallejos Guillermo”

git config –global user.email [guille971172@gmail.com](mailto:guille971172@gmail.com)

mkdir Jobs-Java (vamos a crear el directorio con el nombre del repositorio)

cd Jobs-Java (para ir al directorio)

git init (nos indica si se inicializa y donde)

vamos a agregarle el README

touch README

git add README

Colocamos ese archivo README, en este punto lo tenemos en STAGED

Ahora

git commit –m ‘primer commit’

Nos aparece la información que se ha llevado a cabo .

Ahora para que aparez en remoto, usamos el código

git remote add origin [git@github.com:WilliVal/Jobs-Java.git](mailto:git@github.com:WilliVal/Jobs-Java.git)

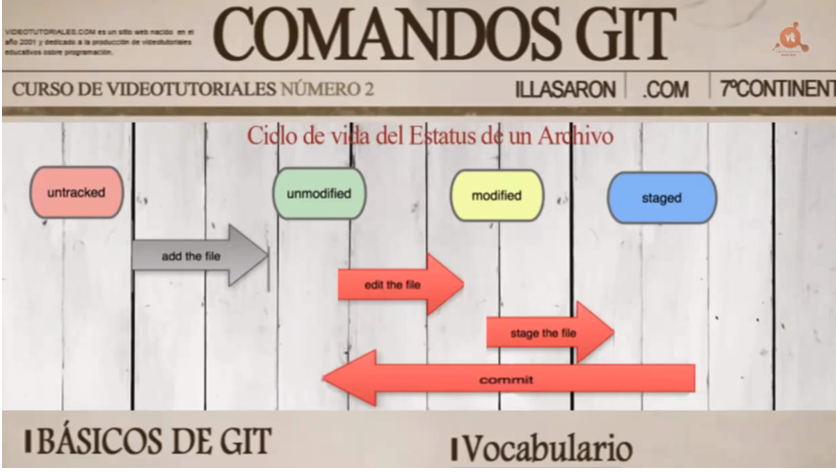
git push –u origin master (como esta colocando información en el GITHUB nos va a pedir la contraseña)

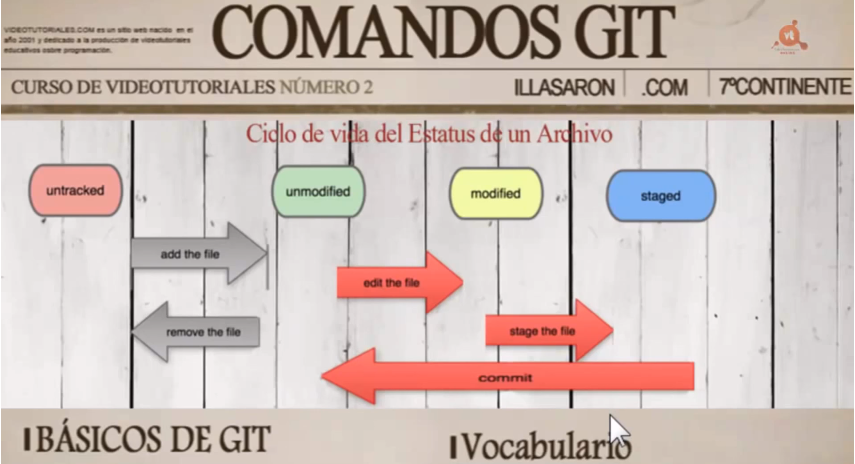
Ahora indica que se ha llevado a cabo el commit , la colocación de la información a GITHUB.

Si accedemos ahora GITHUB, veremos que aparecerá nuestro commit.

.

CICLO DE VIDA DE UN ARCHIVO 21:23





.