

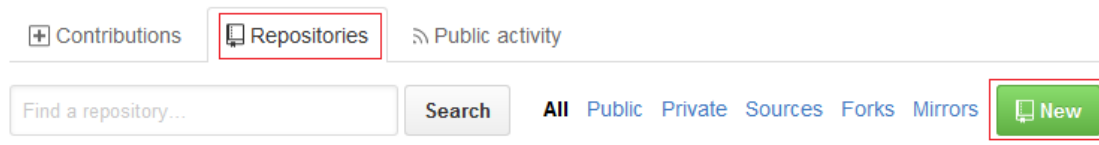
Instalación y configuración GitHub

1º) Para instalarlo, solo es necesario ir a la siguiente página, y elegir el archivo que más le convenga dependiendo de la distribución del sistema operativo que tengas.

<https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>

2º) Configuración

La configuración funciona mediante comandos. Lo que tenemos que hacer, antes que nada, es un repositorio en github para que sea accesible por la gente. Basta con ir a tu perfil dentro de github, pestaña de repositorios, y después a New:



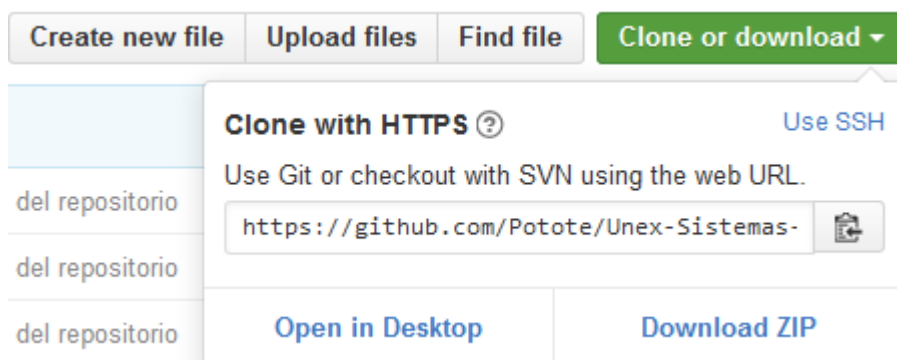
Unex-Sistemas-de-Informacion

Python ★ 0 0

Repositorio para la asignatura de Sistemas de Información de la universidad de Extremadura impartida durante el año académico 2015/2016

Updated 34 minutes ago

Cuando hayamos creado el repositorio, le damos sobre el mismo, y nos aparecerá una pestaña a la derecha que pone Clone or download, nosotros lo vamos a clonar, para ello copiamos la dirección URL que nos sale:



En una carpeta dentro de nuestra sistema, tenemos que poner **git clone "url"** con eso, ya nos esta clonando el repositorio en local.

3º) Comandos, estos son los comandos usados en la consola para poder manejar git, y las funciones que realizan:

- ➔ **git init:** Con esta función, lo que le decimos a git, es que a partir de ahora, esa carpeta donde se encuentran nuestros archivos va a ser un repositorio local, y se va a crear una carpeta oculta denominada .git, a partir de ahora, podemos trabajar como si fuese un repositorio normal y corriente y usando los comandos de git
- ➔ **git remote add origin “url del repositorio”:** Con esta función ya estamos indicando que vamos a subir nuestro repositorio local a dicho repositorio remoto. A partir de ahora, los pull se subirán a dicho repositorio.
- ➔ **git add “nombre_del_fichero”:** Esta función lo que hace es registrar los cambios que hayamos realizado en el archivo “Nombre_del_Fichero”, lo que decimos con esto es que, es que los cambios que vayamos almacenando en local antes de subirlo al repositorio, queremos que dicho archivo también se tenga en cuenta
- ➔ **Git add . :**Esta función lo que hace es añadir TODOS los ficheros dentro de la carpeta donde se encuentra nuestro proyecto, con esta función, nos ahorramos trabajo de ir metiendo uno por uno todos los ficheros. Vamos a ejecutar esta función para nuestro ejercicio, y así ahorrar tiempo.
- ➔ **Git status :** En cualquier momento podemos ver esta función para ver la diferencia que existe entre el repositorio local con respecto al repositorio remoto. Como nosotros hemos realizado cambios en nuestro repositorio local, pero no en el remoto, nos tiene que avisar que existen cambios que queramos modificar con respecto a lo que hay en remoto:

```
C:\Users\Potote\Desktop\Repositorio_prueba>git status
On branch master

Initial commit

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   Fichero_Cambio.txt
        new file:   Fichero_Permanente.txt
```

Vemos que nos está diciendo que en la rama master, existen dos ficheros que necesitan ser pusheados al repositorio remoto. Pero antes de subir al repositorio remoto, es necesario que lo comiteemos en local:

- ➔ **git commit:** Con git commit, lo que hacemos es, una vez añadidos los archivos que queremos que subamos, guardas los cambios en LOCAL para luego subirlo a remoto. Existen varias opciones a la hora de hacer commit:
- ➔ **git commit -m "comentario":** Con esta opción lo que hacemos es añadir un comentario a la hora de ver el cambio en el repositorio remoto. Cuando hagamos el cambio y queramos ver ese cambio, aparecerá esta descripción. Vamos a poner lo siguiente para subir

```
C:\Users\Potote\Desktop\Repositorio_prueba>git commit -m "Subida del
 fichero cambio y del fichero permanente"

*** Please tell me who you are.

Run

  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: empty ident name (for <Potote@Potote-PC.(none)>) not allowed
```

Como podemos ver, sale un error, git no sabe quién somos, para ello simplemente corremos las sentencias que nos pone:

```
C:\Users\Potote\Desktop\Repositorio_prueba>git config --global user.email "Potote_18@hotmail.c
om"

C:\Users\Potote\Desktop\Repositorio_prueba>git config --global user.name "Potote"

C:\Users\Potote\Desktop\Repositorio_prueba>git commit -m "Subida del fichero cambio y del fich
ero permanente"
[master (root-commit) eb75ab5] Subida del fichero cambio y del fichero permanente
 2 files changed, 11 insertions(+)
 create mode 100644 Fichero_Cambio.txt
 create mode 100644 Fichero_Permanente.txt
```

Vemos que se han actualizado 2 ficheros, y que hay un total de 11 líneas de cambio. Si se nos ha olvidado cambiar algo, por ejemplo, añadir una línea al fichero permanente, podemos cambiarlo, y volver a añadir dicho fichero al commit, solo que ahora solo va a cambiar 1 fichero y va a haber solo una única inserción.

```
C:\Users\Potote\Desktop\Repositorio_prueba>git config --global user.email "Potote_18@hotmail.c
om"

C:\Users\Potote\Desktop\Repositorio_prueba>git config --global user.name "Potote"

C:\Users\Potote\Desktop\Repositorio_prueba>git commit -m "Subida del fichero cambio y del fich
ero permanente"
[master (root-commit) eb75ab5] Subida del fichero cambio y del fichero permanente
 2 files changed, 11 insertions(+)
 create mode 100644 Fichero_Cambio.txt
 create mode 100644 Fichero_Permanente.txt
```

Tenemos que subir dichos cambios al repositorio remoto:

➔ **git push -u origin master:** Con esta función, vamos a subir los cambios del local al remoto, a la rama master, cuando hagamos esto, nos va a pedir la contraseña de nuestra cuenta de bitbucket, para ello debemos haber metido de manera correcta quienes somos a git:

```
C:\Users\Potote\Desktop\Repositorio_prueba>git push -u origin master
Password for 'https://Potote@bitbucket.org':
Counting objects: 7, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 719 bytes | 0 bytes/s, done.
Total 7 (delta 1), reused 0 (delta 0)
To https://Potote@bitbucket.org/Potote/repositorio_text.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin.
```

Con esto ya tendremos los cambios realizados en nuestro repositorio remoto, ahora podemos ver como nuestro repositorio ha sido modificado añadiendo ese commit que habíamos hecho, y con ese mismo comentario:

Autor	Commit	Mensaje	Fecha
 Alvaro Nevado	291726e	Subida del fichero cambio y del fichero permanente	12 minutes ago
 Alvaro Nevado	eb75ab5	Subida del fichero cambio y del fichero permanente	13 minutes ago