



Git: el sistema de control de versione por excelencia, y con github, la plataforma de código colaborativo

Registrar el histórico de trabajo de nuestro código,

Crear copias de seguridad

Trabajar en equipo de forma rápida y sin errores

Son las principales caracteristiacas de git y de github

GIT:

Control de versiones distribuido

Ls que tengo aki

Mkdir crear directorio

Pwd donde estoy

Code . abre visual en este path

Cd muéveme a

type nul >> "*file.txt" Crear fichero*

Global—todo el got de el equipo

Git config –global user.name (nombre)

Git config –global user.mail (email)

Git init -> iniciamos el git en ese directorio

La rama master es el “main” de tu git (vambiar nombre -> git **Branch** -m (nombre))

Git add -> agregamos el primer archivo para trabajar con el

Texto

Descripción generada automáticamente

Lo hemos preparado para hacer la fotografía.

Git commit

Texto

Descripción generada automáticamente

O git commit -m "Este es mi primer commit" (mejor)

Texto

Descripción generada automáticamente

No hay nada?:

Texto, Aplicación

Descripción generada automáticamente

Eso es que se ha realizado

Git log

Texto

Descripción generada automáticamente

Creeemos otro fichero

Texto

Descripción generada automáticamente

Los he editao que pasa?

Texto

Descripción generada automáticamente

Quiero volver a antes,

Como no hemos hecho un commit ( foto) con la verion de los archivos editada, podemos volver con un comando fácilmente a la verion anterior en caso que queramos con un simple:

Git checkout ( fichero )

Texto

Descripción generada automáticamente

LLEGAR A UN PUNTO

Imagen que contiene Diagrama

Descripción generada automáticamente

Actualizado:

Vemos un nuevo commit:

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Agregamos otra vez el ( los que hemos tocadao hora 6 minutos de la clase ) add hola para el nuevo futuro commit, y hacemos un commit, con la actualización

De estas manera podemos verlo por ejemplo:

Texto

Descripción generada automáticamente

GIT alias

Piens albert:

Texto

Descripción generada automáticamente

Gitignore

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

Los archivo que nos la pele que nos recuerde que hay que agregarlos al sitio para hacerles un commit por que en el status solo nos recuerda los que han sido modificados

Que curioso, pos lo añadimos:

Texto

Descripción generada automáticamente

mmm… ver que he cambiado respecto a la anterior fotografía.

Si haber hecho la fotografía, de momento hemos visto que cambios hemos hecho respecto a la fotografía anterior con git diff:

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Desplazamiento

Cuando hazes un desplazamiento lo que pasa, es que quermos cambiar de commit es decir de una foto, siempre es mejor tener un commit de el trabajo hecho desde la ultima foto para hacer esto, entonces al movernos, que es simplemente hacer un git checkout pero con el hash de la foto donde queremos ir, la liamos

Donde estamos: HEAD

El ultimo hecho master( ultimo)

Texto

Descripción generada automáticamente

Que es coña que nos volvemos al ultimo comit hecho el MASTER!:

Texto

Descripción generada automáticamente

Git reset –hard

Volvemos a otro pero durismo

Texto

Descripción generada automáticamente

Que noo que lo podemos ver

Git reflog todos los logs de comits hechos

Texto

Descripción generada automáticamente

Git tag agregar un tag a ese commit

Nos podemos mover con los tags también git checkout tag/(el tag de el commit)

Git add . (todos)

Hasta aqui tenemos el movimiento en la rama principal

Git login (nombre de la nueva rama)

Get siwtch (nombre a la rama que movermnos)

Git merge ( donde quieres): juntar dos comits de dos ramas distintas de donde estas a donde quieres

Git stash : me cambio de rama pero no he hecho un commit, guardame este reabajo sin terminar, y sin hacerle un comit, que luego vuelvo

Es un comit pero que no se genera

Me cambio de rama lalala

Vuelvo a la que estaba trabajando y no veo lo que estaba trabajando, pero con git stash list si que veo que estaba haciendo algo así que

Git stash pop y volvemos a donde lo dejamos.

Pero me la pela el stash -> git stash drop

GITHUB