Este es un ejemplo de Github para la comunidad de #mejorandola

Una vez instalas GIT, debes configurarlo:

git config - user.name "cristiano" configuración global git - user.email global " [xxxxxx@xxxx.com](mailto:xxxxxx@xxxx.com) "

Generando tu Public Key:

ssh-keygen

Leyendo tu llave para copiarla a Github:

cat ~ /. ssh / id\_rsa.pub

Arrancando tu proyecto:

git init

README Tocar

git add README

git commit-m "tu primer cometa"

origen master git push

/ \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* GIT de 0 - Por @ klinsmannf \*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* /

**configuración**

Descarga git para OSX

Descarga git para Windows

Descarga git para Linux

**crea un repositorio nuevo**

crea un directorio nuevo, ábrelo y ejecuta git init para crear un repositorio de git nuevo.

**HACER checkout de un repositorio de la ONU**

crea una copia local del repositorio ejecutando git clone /path/to/repository al utilizar un servidor remoto, el comando será git clone username@host:/path/to/repository

**flujo de trabajo**

tu repositorio local esta compuesto por tres "árboles" administrados por git. el primero es tu Directorio de trabajo el cual contiene los archivos. el segundo es el Index el cual actua como un área intermedia y finalmente el HEAD el cual apunta a el último commit realizado.

**añadir y comprometerse**

Puedes proponer cambios (agregarlos a el Index) usando git add git add \* Este es el primer paso en el flujo de trabajo básico. Para en realidad hacer commit a estos cambios usa git commit -m "Commit message" Ahora el archivo esta incluído en el HEAD, pero aún no en tu repositorio remoto.

**envío de cambios**

Tus cambios están ahora en el HEAD de tu copia local. Para enviar esos cambios a tu repositorio remoto ejecuta git push origin master Reemplaza master por la rama a la cual desees enviar tus cambios.

Si no has clonado un repositorio existente y quieres conectar tu repositorio a un repositorio remoto, necesitas agregarlo con git remote add origin Ahora puede subir tus cambios al repositorio remoto selecionado ramas Las ramas son utilizadas para desarrollar funcionalidades aisladas unas de otras. La rama master es la rama por "defecto" cuando creas un repositorio. Usa otras ramas para el desarrollo y fusionalas de regreso a la rama principal al terminar.

**crea una nueva rama llamada "feature\_x" y cambiate a ella usando:**

git checkout-b feature\_x Regresa un maestro la rama principal de git checkout y borra la rama git branch-d feature\_x Una rama no disponible no no está subas Para Los Demas ONU Menos de Que (Push) la rama de la ONU repositorio remoto de origen de git push mamá

**actualiza & fusiona**

para actualizar tu repositorio local al commit más nuevo, ejecuta git pull en tu directorio de trabajo fetch y merge cambios remotos. para fusionar otra rama a tu rama activa (por ejemplo master), utiliza git merge en ambos casos git intenta auto fusionar cambios. Desafortunadamente, esto no es posible todo el tiempo y resulta en conflictos. Tú eres responsable de fusionar esos conflictos manualmente al editar los archivos mostrados por git. Luego de modificarlos, necesitas marcarlos como fusionados con git add antes de fusionar cambios, también puedes dar un vistazo previo usando git diff

**etiquetas**

es recomendado crear etiquetas para cada versión publicada de un software. esto es un concepto conocido, el cual tambien existe en SVN. Puedes crear una nueva etiqueta llamada 1.0.0 ejecutando git tag 1.0.0 1b2e1d63ff el 1b2e1d63ff se refiere a los 10 caracteres de el commit id al cual quieres referirte con tu etiqueta. Puedes obtener el commit id con git log también puedes usar menos caracteres que el commit id, pero debe ser un valor único.

**reemplaza cambios locales**

En caso de que hagas algo malo (que de seguro nunca pasa ;) puedes reemplazar cambios locales usando el comando git checkout -- esto reemplaza los cambios en tu árbol de trabajo con el último contenido de HEAD. Los cambios que ya han sido agregados al índice, así como también nuevos archivos, se mantendrán sin cambio.

Si por otro lado quieres deshacerte de todos los cambios locales y commits, puedes traer la última versión del servidor y apuntar a tu copia local principal de esta forma git fetch origin git reset --hard origin/master

**datos útiles**

Interfaz gráfica por defecto gitk colores especiales para la consola git config color.ui true mostrar sólo una línea por cada commit en la traza git config format.pretty oneline agregar archivos de forma interactiva git add -i

**enlaces & recursos**

Clientes Gráficos GitX (L) (OSX, fuente abierta) Torre (OSX) árbol de código fuente (OSX, gratis) GitHub para Mac (OSX, gratis) Guias Git Community Book Pro Git Piense como un git GitHub Ayuda Guía Visual Git