

Git

Giordano AF

Git instala un bash predefinido para usar los comandos de git.
Estados dentro de git:

1. Working Directory → los archivos que actualmente trabajo. Para pasar al siguiente estado se ingresa el comando git add ...
2. Staging Area → los archivos que serviran para una primera version. Para pasar al siguiente estado de ingresa el comando git commit ...
3. Repository → los archivos que ya estan finalizados para la version final

Comandos más usados

1. git init
Es una manera de decir que se va a iniciar un proyecto nuevo en git o vas a empezar a trabajar un proyecto con git.
2. git add <file>
Para pasar un file de working directory a staging area
3. git status
Para visualizar los estados de los archivos
4. git commit <file> git commit -m "message" <file>
Para pasar a un files a repository para un primer snapshot
5. git push
Subir los files a servidor para que los demas desarrolladores con permiso puedan verlos.
6. git pull
Traer los cambios de los otros desarrolladores
7. git clone
Clonar o copiar el proyecto desde el servidor hasta tu equipo
8. git log
Primer ver los cambios historicos que se dieron al proyecto, se puede ver los snapshots.
9. git checkout - <file>
revierte los cambios en el <file>
10. git diff <file>
Permite ver las diferencias en <file>
11. git branch "branchname"
Permite crear una rama
12. git branch
Permite visualizar las ramas existentes.

13. `git checkout "branchname"`login
Permite cambiar de rama

Observaciones::

Antes de hacer un commit es necesario configurar los registros de autenticacion para ello seguimos los siguientes comandos:

- `git config --global user.email ...`
- `git config --global user.name ...`

Si hay algun problema al tipear mas los datos, simplemente vuelve a ejecutar el comando con la configuracion correcta.

Cuando queremos que ciertas carpetas y/o archivos sea ignorados por git, entonces creamos la carpeta `.gitignore` y colocamos todo lo que queremos que sea ignorado por git dentro de esta carpeta.

No olvidar agregar(`git add`) y `commit` `.gitignore`, ya que esta carpeta si debe ser agregado y comitiado a git.

Una vez tengamos nuestro proyecto avanzado podemos subirlo a un repo de github. Seguimos los siguientes pasos:

- `git init` en la carpeta que esta mos trabajando
- `git add README.md`
- `git commit -m "message"`
- `git remote add origin <el enlace de nuestro repo en github>`
- `git push -u origin master`, aqui agregamos lo que tenemos en la rama master de github