



Ingeniería en Software

Repositorios

Repaso

*¿Qué es Ingeniería de Software?

*¿Cuales son los 4 pasos del Proceso de desarrollo Software?

*¿Qué es SCRUM?¿Qué roles hay?¿Cuales son sus Artefactos? ¿Que reuniones tiene?

Objetivos de la clase

- Aprender sobre el control de versiones.
- Problemas del control de versiones
- GIT y Subversion.

Evolución del Software

El software está constantemente evolucionando durante el **Desarrollo**.

Durante el mantenimiento surgen cambios en el producto.

Muchas veces se necesita retornar a versiones anteriores, por inestabilidad o cambios bruscos.

Que hacíamos?



CControl de
versiones ultimo



Cntrl revision2



Cntrl revision3



Control bueno



Control bueno
corregido



Control de
versiones 1



Control de
versiones 1.2



Control de
versiones Final



Control de
versiones Final1



Control de
versiones Final2



Control de
versiones Revisado



CONTROOOOL



Versiones listo



Versiones listo (2)

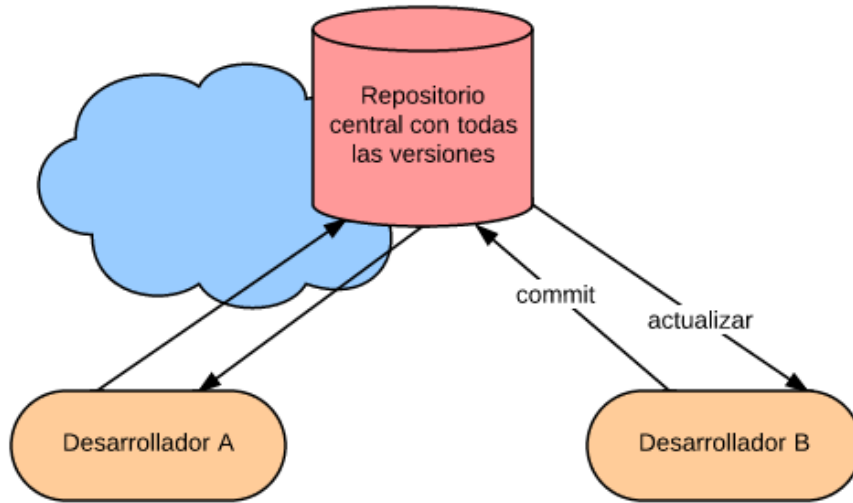
Control de Versiones

Un control de versiones es una herramienta para el desarrollo de software que facilita principalmente dos funciones:

1. Ayuda a que varios programadores pueden trabajar en paralelo.
2. Permite mantener varias versiones de un mismo programa a la vez. (De ahí “control de versiones”)

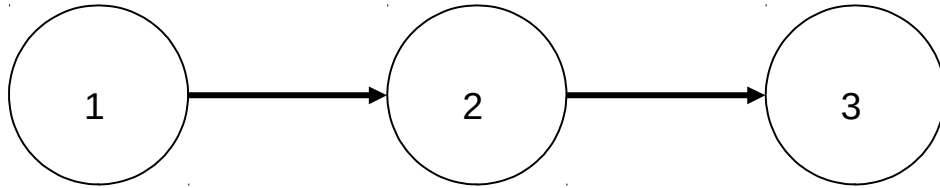
Repositorio

Almacena las Versiones, o sea mantiene la evolución del mismo.



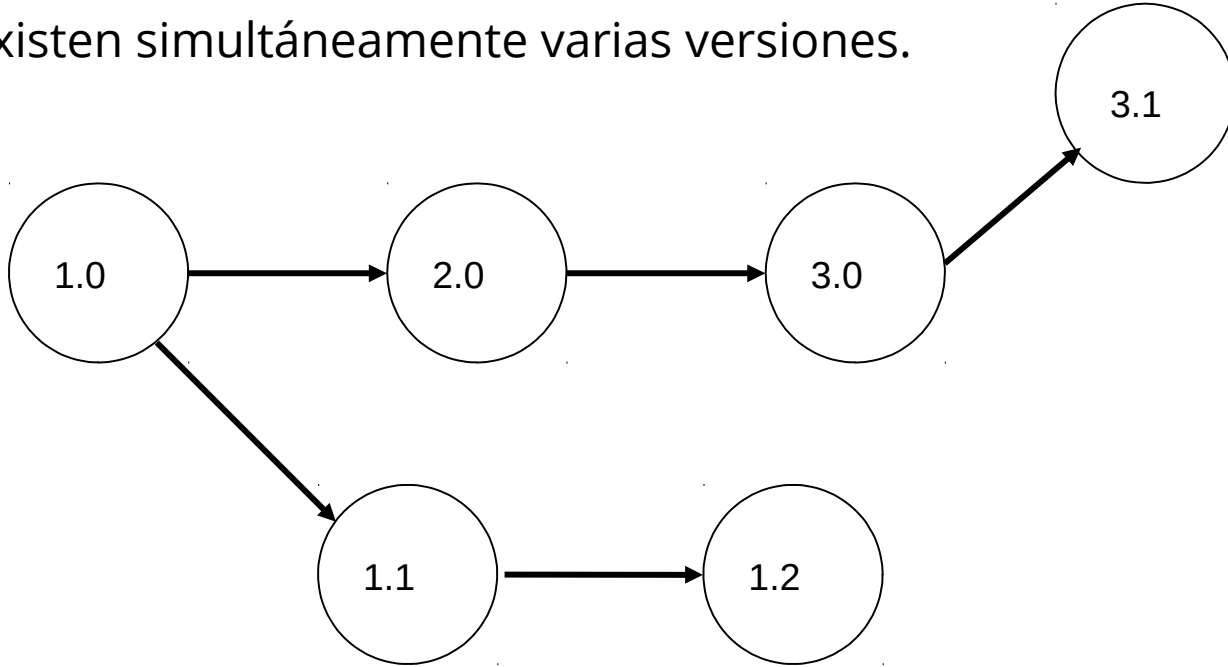
Control de versiones

Evolución Simple.



Control de versiones

Variante: Existen simultáneamente varias versiones.



Control de versiones

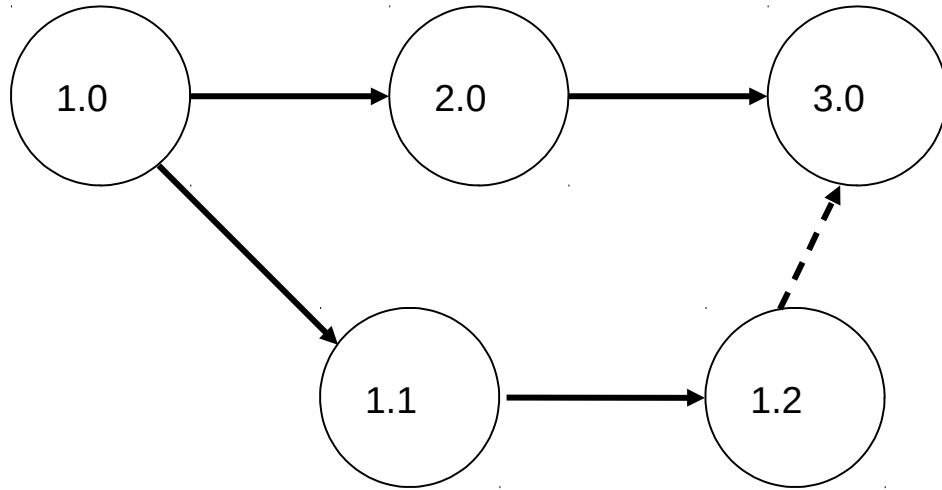
TRUNK: La variante principal

BRANCH: Las variantes Secundarias

DELTA: Diferencias de una versiona la otra

Propagación de cambios

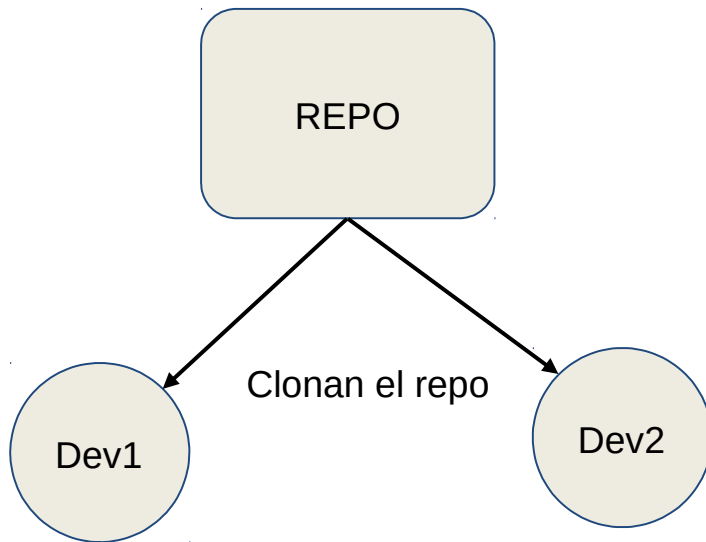
Podemos tener branches paralelos, solo para implementar funcionalidades nuevas. A la Unión se la llama Merge.



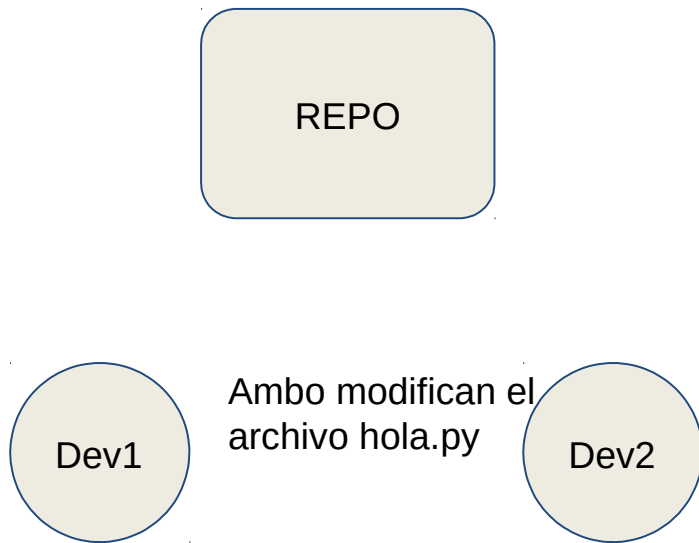
Complejidad del control de versiones

1. Se generan conflictos cuando dos (o más) personas trabajan en el mismo archivo, al terminar de modificarlo y querer impactar sus cambios en el repositorio, corren el riesgo de sobrescribir las modificaciones del otro.

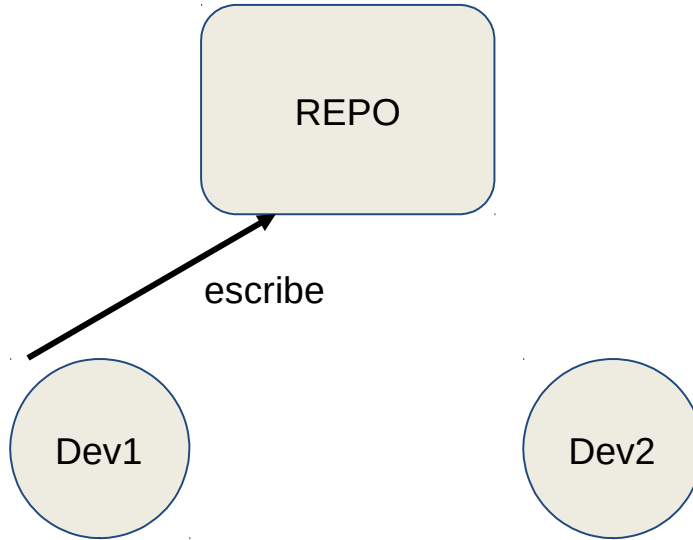
Conflicto de Sobrescritura



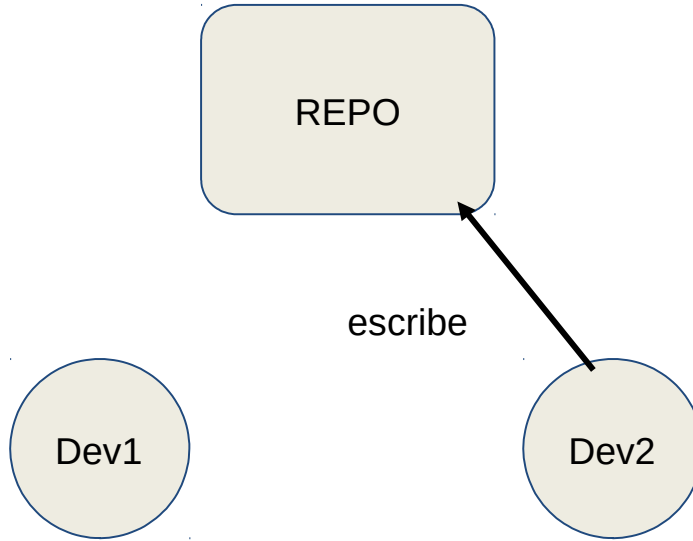
Conflicto de Sobrescritura



Conflicto de Sobrescritura



Conflicto de Sobrescritura



Soluciones

- Cada Dev debe actualizar y arreglar los cambios en su código antes de subir los datos
- Los Dev no pueden modificar código que otro Dev. esté trabajando, si encuentra una falla, comunica el bug
- Cuando se designan las tareas se lo hace modular
- Nunca se copian o pegan archivos todos los cambios se realizan a nivel código y con el versionador

Repositorios

Los más usados:

- **VN, Subversion**
- **GIT**
- Mercurial, Hg
- Bazaar

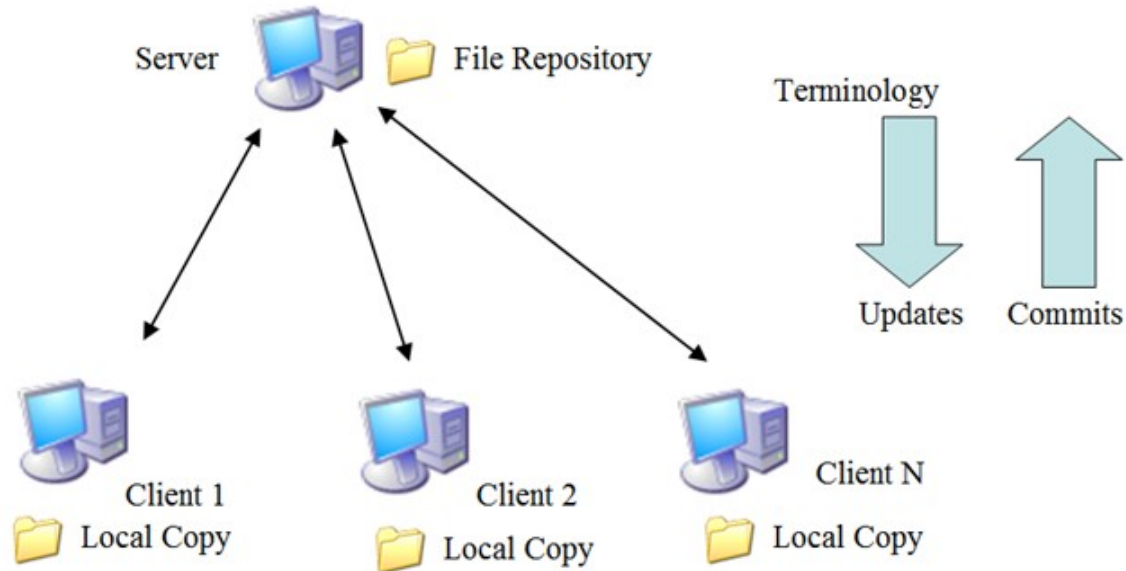
Subversion

Es uno de los primeros gestores de repositorios.

Orientado al Versionado de Proyectos.

Solo realiza una copia Local y cada cambio se debe enviar al server.

Subversion



Subversion

- Es simple de usar en principio
- A mayor cantidad de personas tiende a volverse muy complejo su uso
- Se debe aplicar gestión sobre el repositorio para que funcione en grandes proyectos

GIT

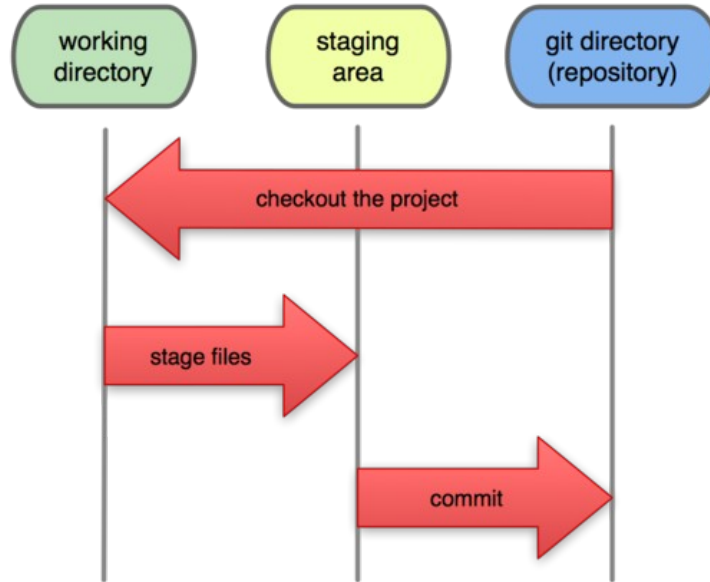
Nació por un conflicto entre Linus Torvald y Subversion para llevar el desarrollo del Kernel de Linux

Es un sistema de Ficheros orientado a objeto versionable

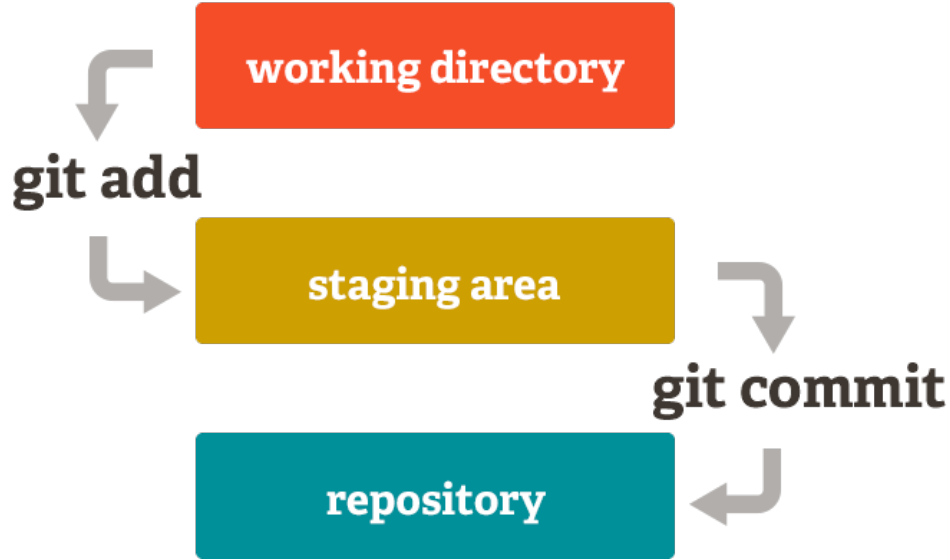
GIT usa Fotos del REPO

GIT

Local Operations



Flujo



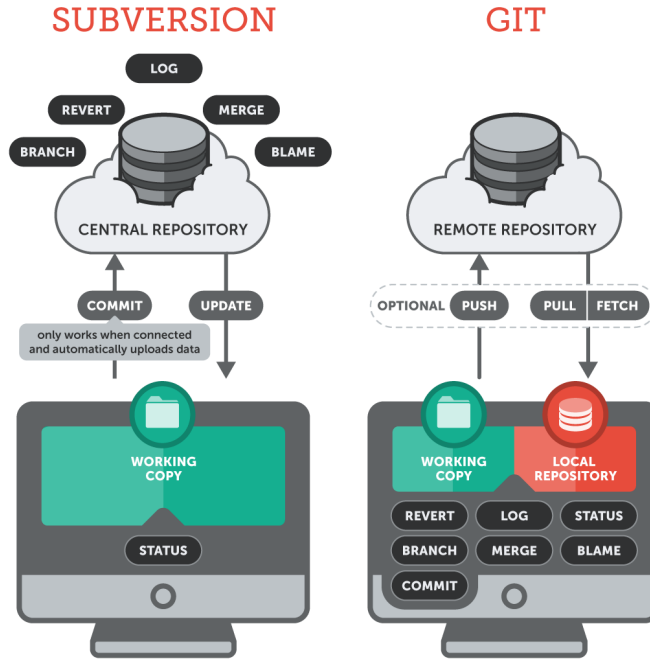
Clonamos y lo tenemos en Working directory --> **git clone**

Editamos, los añadimos a git y los tenemos en el index --> **git add**

Decidimos confirmar y los tenemos en el HEAD --> **git commit**

Y para subirlo lo enviamos al branch remoto --> **git push**

GIT vs Subversion



Diferencias

- **Subversion:** sólo el repositorio central tiene el historial completo. Esto significa que los usuarios deben comunicarse a través de la red con el repositorio central para obtener el historial acerca de un archivo.
- **Git:** la copia local en sí es un repositorio y se puede usar con todas la características y beneficios del control de código fuente. Al recuperar la conectividad con el repositorio principal si estuviese indisponible podemos subir los cambios al servidor

Como podemos tener un repositorio?

Podemos instalar nuestro propio server.



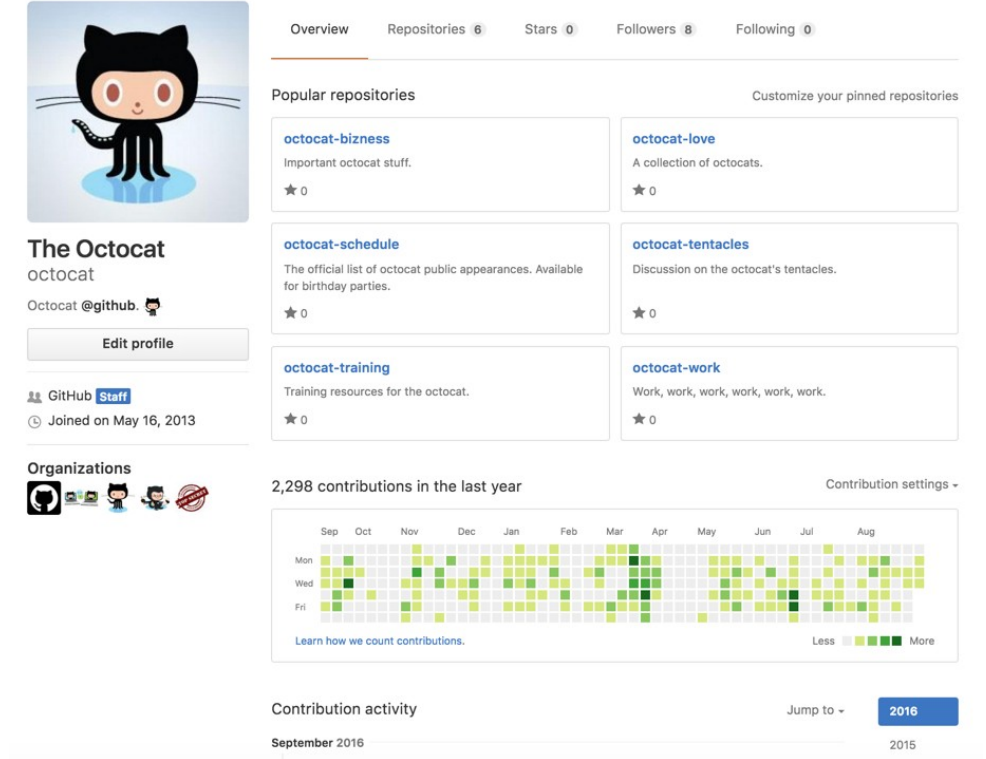
Utilizar los que se encuentran en la nube.



Vamos a usar github

Es el mayor contenedor de repositorios de software libre.

Sirve como curriculum. Muy importante



The screenshot shows the GitHub profile of 'The Octocat' (octocat). The profile includes a bio, a GitHub Staff badge, and a join date of May 16, 2013. The 'Popular repositories' section lists six repositories: octocat-bizness, octocat-love, octocat-schedule, octocat-tentacles, octocat-training, and octocat-work. The 'Contributions' section shows a heatmap of contributions over the last year, with a total of 2,298 contributions. The 'Contribution activity' section shows a calendar view for September 2016, with a 'Jump to' button for 2016.

Overview Repositories 6 Stars 0 Followers 8 Following 0

Popular repositories Customize your pinned repositories

- octocat-bizness**
Important octocat stuff.
★ 0
- octocat-love**
A collection of octocats.
★ 0
- octocat-schedule**
The official list of octocat public appearances. Available for birthday parties.
★ 0
- octocat-tentacles**
Discussion on the octocat's tentacles.
★ 0
- octocat-training**
Training resources for the octocat.
★ 0
- octocat-work**
Work, work, work, work, work, work.
★ 0

2,298 contributions in the last year Contribution settings -

Learn how we count contributions. Less More

Contribution activity Jump to 2016

September 2016 2015

Práctico

- En caso de no tener instalado Git, instalarlo utilizando **apt-get install git**
- Vamos a resolver la siguiente guía de ejercicios
<https://gitexercises.fracz.com/>
- Seguir las instrucciones descritas en la página anterior
- Por ahora, hacer hasta el ejercicio **ignore-them**

Nota:

Les será necesario utilizar los recursos de la siguiente filmína

Importante

Guia

<http://rogerdudler.github.io/git-guide/index.es.html>

Libro

<https://git-scm.com/book/es/v2>

Próximamente vamos a usar GIT FLOW

<http://aprendegit.com/que-es-git-flow/>