

Control de versiones e Introducción a GIT

Definición

Se llama **control de versiones** a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración del mismo. Una versión, revisión o edición de un producto, es el estado en el que se encuentra dicho producto en un momento dado de su desarrollo o modificación

Características deseables

- Mecanismo de almacenamiento de los elementos que deba gestionar
- Posibilidad de modificar los elementos gestionados
- Registro histórico de las acciones realizadas con cada elemento o conjunto de elementos

Terminología - Repositorio

- El repositorio es el lugar en el que se almacenan los datos actualizados e históricos de cambios.

Terminología - Módulo

- Conjunto de directorios y/o archivos dentro del repositorio que pertenecen a un proyecto común

Terminología – Revisión

- Una **revisión** es una versión determinada de la información que se gestiona.
- HEAD: Es el puntero donde uno se encuentra trabajando
- Es una imagen del estado del proyecto al momento en que se realizó el commit

Terminología – Tag (rotular)

- Dar nombre a una versión para destacar e identificar fácilmente del resto.

Terminología – Baseline (línea base)

- Es un Tag que fue aprobado bajo revisión, por lo que sobre él pueden realizarse los cambios subsiguientes.

Terminología – Branch (rama)

- Es una bifurcación en el proyecto
- Copia del proyecto que continua evolucionando en forma independiente
- Puede utilizarse para ramas de prueba
- Permite unirse en un futuro con otras ramas
- Caso de uso: Nueva funcionalidad

Terminología – Checkout

- Se traduce como desplegar, permite crear una copia local del repositorio
- Puede especificarse una versión, por defecto es la última

Terminología – Commit

- Es un registro de cambio
- Es la acción que hace copia e integración de los cambios sobre un repositorio (que puede ser el local)

Terminología – Conflicto

- Es el evento que ocurre cuando el sistema no puede resolver una integración de cambios realizados por distintos usuarios, ya que sus códigos se superponen.
- Requiere asistencia humana para resolverse

Terminología – Resolver

- El acto de la intervención del usuario para atender un conflicto entre diferentes cambios al mismo documento.

Terminología – Diff/cambio/delta

- Es la porción de documento que varía entre una versión y otra

Terminología – Merge

- Es la integración o fusión de dos ramas (branch) del proyecto
- Pueden ser ramas dentro del proyecto o simplemente varias modificaciones de usuarios sobre un mismo archivo

Terminología – Sync/Update

- Una **actualización** integra los cambios que han sido hechos en el repositorio en la copia de trabajo local

Terminología – Workspace

- Es la copia de trabajo local de los ficheros de un repositorio, en un momento del tiempo o revisión específicos
- Conceptualmente, es un **cajón de arena** o **sandbox**.

Terminología – Congelar

- Es cambiar los permisos de una rama o versión del proyecto en solo lectura
- Su fin es obtener una versión consistente

Formas de colaborar

- Exclusiva
- Colaborativa

Arquitecturas de almacenamiento

- **Centralizados:** existe un repositorio centralizado de todo el código. Los cambios se realizan sobre este repositorio
- **Distribuidos:** Cada usuario tiene su propio repositorio. Los distintos repositorios pueden intercambiar y mezclar revisiones entre ellos. Es frecuente el uso de un repositorio común para unificar versiones

GIT

- Git es un Sistema de control de versiones distribuido
- GIT software libre y gratuito.
- Git es multiplataforma

GIT - Donde alojar nuestro repositorio

- **GitHub** es una plataforma que permite alojar código público o privado y brinda un conjunto de herramientas para facilitar distintos tipos de tareas sobre repositorios GIT.
- **GitLab** es otra alternativa para alojar nuestro código y permite que el mismo sea privado
- **Gestor de Repositorios Propio**

GIT - Pasos a seguir: Sin proyecto existente

- Crear un repositorio local en tu pc
- Realizar un commit sobre este repositorio.
- Crear un repositorio remoto
- Realizar un push desde el repositorio local al repositorio remoto

GIT - Pasos a seguir: Con proyecto existente

- **Clonar el repositorio remoto al local**

GIT - Pasos a seguir

- Realizar un nuevo branch para agregar funcionalidades
- Realizar commit periodicamente
- Una vez finalizada la funcionalidad se descargar la última version de la rama principal del proyecto
- Realizar un merge y commit
- Realizar un push sobre la rama a unificar

GIT - Interfaces de uso

- Consola de sistema operativo
- Plugin eclipse
- TortoiseGit
- GitHub Desktop client

Referencias y links recomendados

- http://es.wikipedia.org/wiki/Control_de_versiones
- <http://git-scm.com/>
- <https://try.github.io/levels/1/challenges/1>