Control de versiones

Es una forma por la cual mantengo los cambios, en distintas carpetas para que si me equivoco pueda volver hacia atrás.

**Es una herramienta de software que monitoriza y gestiona cambios en un sistema de archivos**.

Es la práctica de rastrear y gestionar los cambios en el código de software.

No es un sistema de copias de seguridad.

Existen tres tipos de control de versiones:

1 locales

Los cambios en el código fuente se guardan en carpeta nombrándolas con fechas, este ya no se debe utilizar ya que no se puede compartir con facilidad a distintos programadores, hay dificultad para controlar los cambios en los distintos cambios de los miembros del equipo.

El primer sistema de control de versiones fue llamado rcs y es únicamente local, este no soluciona como trabajar en equipo y como compartir con los compañeros. Este sistema genera un archivo con terminación v el cual indica que este archivo esta en el control de versiones. Para registrar los cambios en el archivo se utilizaba el comando ci –u nombre de archivo, este sistema lo único que soluciona es no tener muchas carpetas.

2 centrales

Este sistema se sigue utilizando y el primero que existió de este tipo es el CVS el cual evoluciono del rcs, este sistema ya utiliza un servidor para guardar u obtener los cambios en el código fuente. Otro sistema de este tipo es el SUBVERSION o svn. La desventaja es que no se pueden enviar los cambios en el código cuando el servidor esta caído. Ambos utilizan una terminal de comandos para utilizar estos sistemas.

3 distribuidos

En este tipo se encuentra git el cual es el sistema de control de versiones mas utilizado actualmente , pero existen otros como mercurial y bazaar.

Las ventajas de estos sistemas de control de versiones son :

* Trabajo en local y con servidor.
* Pueden enviar cambios a un repositorio dentro de un ordenador local es decir tienen una copia completa del código y se puede compartir con los compañeros a través de una red http y ssh y publicar por internet.
* Se tiene una copia completa del código fuente tanto en el servidor así como en el ordenador ya que git trabaja con referencias o punteros, en git se utiliza un historial de confirmaciones que se puede ver con el comando git log, en cual se pueden ver los cambios que se han tenido y se puede cambiar con facilidad de una versión a otra. Se puede trabajar aunque no haya conexión a internet; es decir se pueden modificar los ficheros, enviarlos al repositorio local y una vez que se tenga conexión a una red o a internet enviarlos a un repositorio remoto.
* Específicamente en los objects de la carpeta oculta .git se guarda el código fuente del proyecto de una forma comprimida. Y en el fichero índex se encuentra la configuración principal

/\*\* me quede en el minuto 35.52 del video introducción al control de versiones faltando 34:56 minutos para terminarlo \*\*/