**UNIVERSIDADTECNICA NACIONAL (UTN)**

**TEMA**

**TAREA #1**

**PROFESOR:**

**ESTUDIANTE:**

**DIEGO GARCIA FONSECA**

**SEDE SAN CARLOS**

**9 DE MAYO DEL 2014**

Contenido

[**ADMINISTRADOR DE CONTROL DE VERSIONES:** 3](#_Toc387342186)

[**CARACTERISTICAS DEL CONTROL DE VERSIONES:** 3](#_Toc387342187)

[**Git** 3](#_Toc387342188)

[**Mercurial** 4](#_Toc387342189)

[**Subversion (SVN)** 5](#_Toc387342190)

[**BITBUCKET** 5](#_Toc387342191)

[**GITHUB** 6](#_Toc387342192)

# **ADMINISTRADOR DE CONTROL DE VERSIONES:**

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 el mismo en un momento dado de su desarrollo o modificación. Aunque un sistema de control de versiones puede realizarse de forma manual, es muy aconsejable disponer de herramientas que faciliten esta gestión dando lugar a los llamados **sistemas de control de versiones** o SVC (del inglés *System Version Control*). Estos sistemas facilitan la administración de las distintas versiones de cada producto desarrollado, así como las posibles especializaciones realizadas (por ejemplo, para algún cliente específico)

## **CARACTERISTICAS DEL CONTROL DE VERSIONES:**

Un sistema de control de versiones debe proporcionar:

* Mecanismo de almacenamiento de los elementos que deba gestionar (ej. archivos de texto, imágenes, documentación...).
* Posibilidad de realizar cambios sobre los elementos almacenados (ej. modificaciones parciales, añadir, borrar, renombrar o mover elementos).
* Registro histórico de las acciones realizadas con cada elemento o conjunto de elementos (normalmente pudiendo volver o extraer un estado anterior del producto).

Aunque no es estrictamente necesario, suele ser muy útil la generación de informes con los cambios introducidos entre dos versiones, informes de estado, marcado con nombre identificativo de la versión de un conjunto de ficheros, etc.

# **Git**

 es un software de [control de versiones](http://es.wikipedia.org/wiki/Control_de_versiones) diseñado por [Linus Torvalds](http://es.wikipedia.org/wiki/Linus_Torvalds" \o "Linus Torvalds), pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de [código fuente](http://es.wikipedia.org/wiki/C%C3%B3digo_fuente). Al principio, Git se pensó como un motor de bajo nivel sobre el cual otros pudieran escribir la interfaz de usuario o [front end](http://es.wikipedia.org/wiki/Front-end_y_back-end" \o "Front-end y back-end) como [Cogito](http://es.wikipedia.org/wiki/Cogito_(software)) o [StGIT](http://es.wikipedia.org/w/index.php?title=StGIT&action=edit&redlink=1).[2](http://es.wikipedia.org/wiki/Git#cite_note-2) Sin embargo, Git se ha convertido desde entonces en un sistema de control de versiones con funcionalidad plena. [3](http://es.wikipedia.org/wiki/Git#cite_note-3) Hay algunos proyectos de mucha relevancia que ya usan Git, en particular, el grupo de [programación](http://es.wikipedia.org/wiki/Programaci%C3%B3n) del [núcleo Linux](http://es.wikipedia.org/wiki/N%C3%BAcleo_Linux).

El [mantenimiento del software](http://es.wikipedia.org/wiki/Mantenimiento_de_Software) Git está actualmente (2009) supervisado por Junio Hamano, quien recibe contribuciones al código de alrededor de 280 programadores.

Entre las características más relevantes se encuentran:

* Fuerte apoyo al desarrollo no lineal, por ende rapidez en la gestión de ramas y mezclado de diferentes versiones. Git incluye herramientas específicas para navegar y visualizar un historial de desarrollo no lineal. Una presunción fundamental en Git es que un cambio será fusionado mucho más frecuentemente de lo que se escribe originalmente, conforme se pasa entre varios programadores que lo revisan.
* Gestión distribuida. Al igual que [Darcs](http://es.wikipedia.org/wiki/Darcs" \o "Darcs), [BitKeeper](http://es.wikipedia.org/wiki/BitKeeper" \o "BitKeeper), [Mercurial](http://es.wikipedia.org/wiki/Mercurial), [SVK](http://es.wikipedia.org/w/index.php?title=SVK&action=edit&redlink=1), [Bazaar](http://es.wikipedia.org/wiki/Bazaar_(software)" \o "Bazaar (software)) y [Monotone](http://es.wikipedia.org/wiki/Monotone" \o "Monotone), Git le da a cada programador una copia local del historial del desarrollo entero, y los cambios se propagan entre los repositorios locales. Los cambios se importan como ramas adicionales y pueden ser fusionados en la misma manera que se hace con la rama local.
* Los almacenes de información pueden publicarse por [HTTP](http://es.wikipedia.org/wiki/HTTP), [FTP](http://es.wikipedia.org/wiki/File_Transfer_Protocol), [rsync](http://es.wikipedia.org/wiki/Rsync" \o "Rsync) o mediante un protocolo nativo, ya sea a través de una conexión TCP/IP simple o a través de cifrado[SSH](http://es.wikipedia.org/wiki/SSH). Git también puede emular servidores [CVS](http://es.wikipedia.org/wiki/CVS), lo que habilita el uso de clientes CVS pre-existentes y módulos IDE para CVS pre-existentes en el acceso de repositorios Git.

# **Mercurial**

es un [sistema de control de versiones](http://es.wikipedia.org/wiki/Sistema_de_control_de_versiones) [multiplataforma](http://es.wikipedia.org/wiki/Multiplataforma), para [desarrolladores de software](http://es.wikipedia.org/wiki/Desarrollador_de_software). Está implementado principalmente haciendo uso del [lenguaje de programación](http://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n) [Python](http://es.wikipedia.org/wiki/Python" \o "Python), pero incluye una implementación binaria de [diff](http://es.wikipedia.org/wiki/Diff" \o "Diff) escrita en [C](http://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n_C). Mercurial fue escrito originalmente para funcionar sobre Linux. Ha sido adaptado para [Windows](http://es.wikipedia.org/wiki/Microsoft_Windows), [Mac OS X](http://es.wikipedia.org/wiki/Mac_OS_X) y la mayoría de otros sistemas [tipo Unix](http://es.wikipedia.org/wiki/Unix). Mercurial es, sobre todo, un programa para la [línea de comandos](http://es.wikipedia.org/wiki/L%C3%ADnea_de_comandos). Todas las operaciones de Mercurial se invocan como opciones dadas a su programa motor. El creador y desarrollador principal de Mercurial es [Matt Mackall](http://es.wikipedia.org/w/index.php?title=Matt_Mackall&action=edit&redlink=1" \o "Matt Mackall (aún no redactado)). El [código fuente](http://es.wikipedia.org/wiki/C%C3%B3digo_fuente) se encuentra disponible bajo los términos de la licencia [GNU GPL](http://es.wikipedia.org/wiki/GNU_General_Public_License) versión 2, lo que clasifica a Mercurial como [software libre](http://es.wikipedia.org/wiki/Software_libre).

SOFTWARE RELACIONADO CON MERCURIAL.

* El [IDE Netbeans](http://es.wikipedia.org/wiki/NetBeans) admite trabajar con Mercurial desde su versión 6.
* [Tortoise Hg](http://tortoisehg.sourceforge.net/) proporciona una interfaz amigable para Windows, con menús contextuales (de botón derecho).
* El [IDE Eclipse](http://es.wikipedia.org/wiki/Eclipse_(software)) soporta Mercurial a través de la extensión [MercurialEclipse](http://www.javaforge.com/project/HGE).
* [VisualHG](http://visualhg.codeplex.com/) es un proveedor de control de código fuente para MS Visual Studio 2005, 2008 y 2010 que habilita el uso de Mercurial.

# **Subversion (SVN)**

es una herramienta de control de versiones open source basada en un repositorio cuyo funcionamiento se asemeja enormemente al de un sistema de ficheros. Es software libre bajo una licencia de tipo Apache/BSD.

Utiliza el concepto de revisión para guardar los cambios producidos en el repositorio. Entre dos revisiones sólo guarda el conjunto de modificaciones (delta), optimizando así al máximo el uso de espacio en disco. SVN permite al usuario crear, copiar y borrar carpetas con la misma flexibilidad con la que lo haría si estuviese en su disco duro local. Dada su flexibilidad, es necesaria la aplicación de buenas prácticas para llevar a cabo una correcta gestión de las versiones del software generado.

VENTAJAS

Se sigue la historia de los archivos y directorios a través de copias y renombrados..

Las modificaciones (incluyendo cambios a varios archivos) son [atómicas](http://es.wikipedia.org/wiki/Atomicidad).

La creación de ramas y etiquetas es una operación más eficiente. Tiene coste de complejidad constante (O(1)) y no lineal (O(n)) como en CVS.

Se envían sólo las diferencias en ambas direcciones (en CVS siempre se envían al servidor archivos completos).

Puede ser servido mediante [Apache](http://es.wikipedia.org/wiki/Servidor_HTTP_Apache), sobre [WebDAV](http://es.wikipedia.org/wiki/WebDAV" \o "WebDAV)/[DeltaV](http://es.wikipedia.org/w/index.php?title=DeltaV&action=edit&redlink=1" \o "DeltaV (aún no redactado)). Esto permite que clientes WebDAV utilicen Subversion de forma transparente.

Maneja eficientemente archivos binarios (a diferencia de CVS que los trata internamente como si fueran de texto).

# **BITBUCKET**

Bitbucket es un servicio de alojamiento basado en web, para los proyectos que utilizan el sistema de control de revisiones MERCURIAL y GIT. Bitbucket ofrece planes comerciales y gratuitos. Se ofrece cuentas gratuitas con un número ilimitado de repositorios privados (que puede tener hasta cinco usuarios en el caso de cuentas gratuitas) desde septiembre de 2010, los repositorios privados no se muestran en las páginas de perfil - si un usuario sólo tiene depósitos privados, el sitio web dará el mensaje "Este usuario no tiene repositorios". El servicio está escrito en [Python](http://es.wikipedia.org/wiki/Python" \o "Python).

# **GITHUB**

GitHub es una [forja](http://es.wikipedia.org/wiki/Forja_(software)) para alojar proyectos utilizando el sistema de [control de versiones](http://es.wikipedia.org/wiki/Control_de_versiones) [Git](http://es.wikipedia.org/wiki/Git" \o "Git). Utiliza el [framework](http://es.wikipedia.org/wiki/Framework" \o "Framework) [Ruby on Rails](http://es.wikipedia.org/wiki/Ruby_on_Rails) porGitHub, Inc. (anteriormente conocida como Logical Awesome).

Desde enero de 2010, GitHub opera bajo el nombre de GitHub, Inc.

El código se almacena de forma [pública](http://es.wikipedia.org/wiki/C%C3%B3digo_abierto), aunque también se puede hacer de forma privada, creando una cuenta de pago.