

SOFTWARE DE CONTROL DE VERSIONES

SOFTWARE	Características	Ventajas	Desventajas	paga / no paga
GIT	<p>Es un sistema de control de versiones distribuido.</p> <p>Permite a cada usuario tener una copia completa del repositorio, lo que hace posible trabajar de manera autónoma.</p>	<p>Desempeño rápido: permite gestionar versiones de código de manera ágil sin necesidad de conexión a un servidor central, lo que optimiza el flujo de trabajo</p> <p>Soporte amplio: proporciona documentación, foros y herramientas adicionales para facilitar su uso</p> <p>Flexible: adaptar sus flujos de trabajo según sus necesidades, ya sea mediante ramas, bifurcaciones o integraciones en plataformas</p>	<p>Curva de aprendizaje: su uso puede ser complicado para principiantes debido a la variedad de comandos y opciones disponibles</p> <p>Complejidad innecesaria para proyectos pequeños: donde un control de versiones más sencillo podría ser suficiente. Esto puede generar una sobrecarga de procesos y herramientas que no siempre son requeridas en desarrollos de menor escala.</p>	Es gratuito
Subversion (svn)	<p>Es un sistema de control de versiones centralizado.</p> <p>Los usuarios deben conectarse al repositorio central para obtener los archivos y hacer cambios.</p>	<p>Fácil de aprender: su estructura de trabajo basada en un servidor central facilita la administración y el seguimiento de cambios, lo que resulta beneficioso para equipos que buscan una solución</p>	<p>No permite trabajar offline: depende de un servidor central para acceder a los archivos y realizar cambios. Esto puede ser un inconveniente en situaciones donde los desarrolladores</p>	Es gratuito

		<p>sencilla y estructurada</p> <p>Control centralizado: permite una gestión más ordenada y un mayor control sobre el acceso y las modificaciones realizadas por los usuarios</p>	<p>no tienen conexión a internet o cuando el servidor presenta fallos</p> <p>Escalabilidad limitada: puede volverse menos eficiente a medida que el proyecto crece y el número de archivos y usuarios aumenta</p>	
Mercurial	<p>Sistema de control de versiones, más sencillo en su diseño.</p> <p>Permite a cada usuario tener una copia completa del repositorio.</p> <p>Utiliza una interfaz de línea de comandos bastante amigable</p>	<p>Fácil de usar: su interfaz y comandos son más intuitivos, lo que facilita su adopción por parte de nuevos usuarios</p> <p>Buena para proyectos medianos: ofrece un rendimiento eficiente y estable sin la complejidad innecesaria de otros sistemas más avanzados. Su diseño busca minimizar errores y proporcionar un flujo de trabajo claro para los desarrolladores</p>	<p>Menos soporte y herramientas: puede dificultar su integración con ciertas plataformas y limitar las opciones para la gestión de proyectos</p> <p>No tan robusto: otros sistemas de control de versiones, especialmente en proyectos de gran escala donde la flexibilidad y la personalización son factores clave. Aunque es una alternativa confiable, en muchos casos los equipos prefieren herramientas con mayor soporte y comunidad activa</p>	Es gratuito

Bazaar (Bzr)	<p>Sistema de control de versiones distribuido</p> <p>Diseñado para ser fácil de usar y configurar.</p> <p>Tiene una integración estrecha con Ubuntu, aunque su uso ha disminuido con el tiempo.</p>	<p>Fácil de aprender: su diseño intuitivo y su sintaxis clara permiten que los usuarios lo adopten rápidamente sin necesidad de un aprendizaje complejo</p> <p>Buena para proyectos pequeños: ofrece una gestión eficiente de versiones sin la sobrecarga de funciones avanzadas que pueden ser innecesarias en desarrollos de menor escala</p>	<p>No recomendado para grandes proyectos: ya que su rendimiento puede verse afectado cuando se maneja una gran cantidad de archivos y cambios en el código</p>	Es gratuito
Perforce Helix Core	<p>Sistema de control de versiones centralizado, pero optimizado para rendimiento escalable.</p> <p>Maneja archivos binarios de manera eficiente, lo que lo hace adecuado para proyectos con grandes archivos.</p>	<p>Alto rendimiento: está optimizado para manejar grandes volúmenes de datos y cambios de manera rápida y eficiente</p> <p>Ideal para archivos binarios grandes: excelente opción para industrias como el desarrollo de videojuegos y la ingeniería de software, donde es común manejar archivos pesados que otros sistemas</p>	<p>Licencia de pago para equipos grandes: puede representar una barrera para startups o pequeños equipos de desarrollo que buscan soluciones gratuitas o de código abierto. Aunque ofrece versiones gratuitas para individuos o equipos pequeños, su implementación a gran escala implica costos que pueden ser elevados</p>	Tiene una versión gratuita para equipos pequeños (hasta 5 usuarios) y versiones de pago para equipos grandes

		de control de versiones pueden gestionar con menos eficacia.		
--	--	--	--	--

Gráfica de comparación entre los software de control de versiones en los últimos 7 días

