INTRODUCCIÓN. –

HISTORIA. –

**Git** es un *software* de [**control de versiones**](https://es.wikipedia.org/wiki/Control_de_versiones) diseñado por [**Linus Torvalds**](https://es.wikipedia.org/wiki/Linus_Torvalds), pensando en la eficiencia, la confiabilidad y compatibilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de [**código fuente**](https://es.wikipedia.org/wiki/C%C3%B3digo_fuente). Su propósito es llevar registro de los cambios en archivos de computadora incluyendo coordinar el trabajo que varias personas realizan sobre archivos compartidos en un repositorio de código.

Al principio, Git se pensó como un motor de bajo nivel sobre el cual otros pudieran escribir la interfaz de usuario o [**front end**](https://es.wikipedia.org/wiki/Front-end_y_back-end) como [**Cogito**](https://es.wikipedia.org/wiki/Cogito_(software)) o [**StGIT**](https://es.wikipedia.org/w/index.php?title=StGIT&action=edit&redlink=1). ​Sin embargo, Git se ha convertido desde entonces en un sistema de control de versiones con funcionalidad plena. ​Hay algunos proyectos de mucha relevancia que ya usan Git, en particular, el grupo de [**programación**](https://es.wikipedia.org/wiki/Programaci%C3%B3n) del [**núcleo Linux**](https://es.wikipedia.org/wiki/N%C3%BAcleo_Linux).

El [**mantenimiento del software**](https://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. En cuanto a derechos de autor Git es un [**software libre**](https://es.wikipedia.org/wiki/Software_libre) distribuible bajo los términos de la versión 2 de la [**Licencia Pública General de GNU**](https://es.wikipedia.org/wiki/GNU_General_Public_License).

El diseño de Git se basó en [**BitKeeper**](https://es.wikipedia.org/wiki/BitKeeper) y en [**Monotone**](https://es.wikipedia.org/wiki/Monotone). Originalmente fue diseñado como un motor de sistema de control de versiones de bajo nivel sobre el cual otros podrían codificar interfaces frontales, tales como [**Cogito**](https://es.wikipedia.org/wiki/Cogito_(software)) o [**StGIT**](https://es.wikipedia.org/w/index.php?title=StGIT&action=edit&redlink=1).​ Desde ese entonces hasta ahora el núcleo del proyecto Git se ha vuelto un sistema de control de versiones completo, utilizable en forma directa.

[**Linus Torvalds**](https://es.wikipedia.org/wiki/Linus_Torvalds) buscaba un sistema distribuido que pudiera usar en forma semejante a BitKeeper, pero ninguno de los sistemas bajo software libre disponibles cumplía con sus requerimientos, especialmente en cuanto a desempeño. El diseño de Git mantiene una enorme cantidad de código distribuida y gestionada por mucha gente, que incide en numerosos detalles de rendimiento, y de la necesidad de rapidez en una primera implementación.