1. **¿Quién inventó el sistema de control de versión Git y por qué?**

Git es un sistema de control de versiones distribuido, fue creado por Linus Torvalds en 2005 para ser utilizado en el desarrollo del kernel de Linux.

Originalmente, la comunidad de desarrolladores de Linux utilizaba [BitKeeper](http://www.bitkeeper.com/" \t "_blank), un sistema de control de versiones distribuido y comercial, aunque gratuito para proyectos de código abierto. Como el propio Linus Torvalds explica, BitKeeper suponí­a un gran paso adelante comparado con otros SCMs como [CVS](http://www.nongnu.org/cvs/) y [Subversion](https://subversion.apache.org/" \t "_blank) y, sin duda, inspiró muchos de los conceptos en los que se basa git (especialmente el ser un sistema distribuido).

Sin embargo, el hecho de ser un software propietario y ciertas desavenencias con su creador, Larry McVoy, llevaron a Linus Torvalds a decantarse por buscar un nuevo sistema de control de versiones. Tras probar muchas de las soluciones existentes y descubrir que ninguna cumplí­a con sus requisitos, Torvalds decidió diseñar e implementar su propio sistema de control de versiones. Tan sólo un dí­a después ya habí­a desarrollado las bases de git para poder auto-gestionar el propio código de git en git y, una semana después, ya tení­a una primera versión. Resulta curioso que en la [primera mención pública de git](http://lkml.iu.edu/hypermail/linux/kernel/0504.0/2022.html), cuando llevaba un par de dí­as de desarrollo, Torvalds describe git no como un sistema de control de versiones sino simplemente como una herramienta de distribución y archivado de ficheros. El resto, como suele decirse, es historia.

Git fue diseñado por Torvalds con el proyecto de Linux y el workflow de sus desarrolladores en mente y, obviamente, estos fueron los primeros en adoptarlo como sistema de control de versiones. El modo de funcionamiento distribuido, su eficiencia a la hora de trabajar con múltiples ramas y otras operaciones comunes como diffs (que en git se ejecutan en un suspiro), dan testimonio de cómo git fue diseñado con la intención de gestionar proyectos grandes como el kernel sin problemas.

1. **¿A quién pertenece actualmente Github y por qué?**

Microsoft desde el 4 de junio de 2018anunció la compra de la compañía por un importe de 7500 millones de dólares. Cerrándose la adquisición el 26 de octubre de 2018.

En plena explosión de cloud services (servicios en la nube), con un [crecimiento previsto para este año del 21.4%](https://www.gartner.com/newsroom/id/3871416), Microsoft tiene que competir con el líder indiscutible a día de hoy, la todopoderosa [Amazon con AWS](https://aws.amazon.com/es/). Qué mejor forma de hacerlo que estar en contacto directo con los desarrolladores que trabajan en esos servicios, ver sus necesidades y ofrecerles herramientas que faciliten su integración con Azure. Es así pues, que esta adquisición es, sin duda, un acercamiento a la comunidad más vibrante en el desarrollo de software libre y abierto.

1. **¿Hay otra forma que no sea la terminal para trabajar con Github?**

Aunque la terminal es la forma más eficiente de trabajar con Git, también existen herramientas de interfaz gráfica que permiten visualizar de manera más amigable con el usuario las ramas y el flujo de trabajo del proyecto en Git. Entre las interfaces gráficas más populares se encuentran:

* GitKraken
* Sourcetree
* GitHub Desktop
* Fork