

Git y GitHub

Sofía Mansilla

Practica Individual Obligatoria

1. ¿Quién inventó el sistema de control de versión Git y por qué?
 - Linus Torvalds, el creador de Linux, fue quien inventó el control de versión Git, esto debido a que antiguamente los cambios de software se realizaban a través de parches y archivos y posteriormente se creó BitKeeper, un VCS que pasó de ser gratuito a pago. Esto impulso a la comunidad de desarrollo de Linux a desarrollar su propia herramienta basada en algunas lecciones que aprendieron mientras usaban Bitkeeper. Algunos de los objetivos del nuevo sistema fueron los siguientes:
 - i. Velocidad
 - ii. Diseño sencillo
 - iii. Gran soporte para desarrollo no lineal (miles de ramas paralelas)
 - iv. Completamente distribuido
 - v. Capaz de manejar grandes proyectos eficientemente
2. ¿A quién pertenece actualmente GitHub y por qué?
 - Actualmente GitHub pertenece a Microsoft, lo que ha causado grandes polémicas. GitHub tiene un valor estratégico para la compañía pues esta siempre se ha enfocado en desarrolladores. Microsoft adoptó una estrategia nueva, mucho más abierta al mundo del open source y a la integración con otras plataformas y servicios. La compra de GitHub buscaba convencer a los desarrolladores de estas soluciones que trabajaban en entornos mixtos que podían basar su trabajo en Windows sin perder eficiencia. Probablemente otra de las motivaciones de Microsoft de adquirir GitHub fue querer mantener relevancia en el desarrollo de todo tipo de plataformas, incluidas por supuesto las orientadas a dispositivos móviles.
3. ¿Hay otra forma que no sea la terminal para trabajar con GitHub?
 - La terminal es el único lugar en donde puedes ejecutar todos los comandos de Git, pero hay una gran variedad de interfaces de usuario con distintas capacidades para usarlo (la mayoría de las interfaces graficas de usuario implementan una parte de las características de Git por motivos de seguridad).
 - i. Interfaces gráficas
 - ii. Git en Visual Studio
 - iii. Git en Eclipse
 - iv. Git con Bash
 - v. Git en Zsh
 - vi. Git en Powershell
 - Si uno quisiera integrar Git con una aplicación para desarrolladores, tendría tres opciones de uso:
 - i. Generar un Shell con CLI
 - ii. Libgit2
 - iii. JGit