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

R/ Git fue inventado por Linus Torvalds el creador del kernel Linux, Ing. de Software Finlandés-Estadounidense, Git nace como una herramienta de control de versiones en el año 2005 como alternativa a BitKeeper (La cual dejo de ser gratuita), Él pensó en crear y utilizar esté software para tener un control total sobre los aportes que otros desarrolladores que apoyaban en su momento el proyecto, Kernel Linux.

Se puede decir que Linus Torvalds y la comunidad aportante creo dos tecnologías muy importantes para la industria hoy en día. Git como software de control de versiones y Linux como SO.

Breve historia de Git.

<https://git-scm.com/book/es/v2/Inicio---Sobre-el-Control-de-Versiones-Una-breve-historia-de-Git>

b. ¿A quién pertenece actualmente GitHub y por qué?

R/ GitHub actualmente es propiedad de Microsoft, y les costó 7.500 millones de dólares su adquisición, fue la estrategia perfecta comprar el repositorio de código más grande del mundo, GitHub aproximadamente tiene registrados 27 millones de usuarios (no todos desarrolladores) y 80 millones de proyectos (repositorios) de tipo Open Source, es claro que Microsoft cada vez quiere atraer más y más desarrolladores a su familia, y aunque puede ser bueno por la cantidad de herramientas y productos que liberan gradualmente bajo la licencia Open Source, puede ser contraproducente para las entidades o Startups que compiten directamente contra los productos de Microsoft, pensar en que Microsoft tiene acceso a los secretos más profundos de sus adversarios. Y Google al ver que perdió la posibilidad de adquirir GitHub, empezó a invertir en GitLab, competidor directo de GitHub.

c. ¿Hay otra forma que no sea la terminal para trabajar con GitHub?

R/ Actualmente GitHub tiene herramientas muy potentes en su GUI Web para realizar la gestión de proyectos (repositorios), estás mismas funcionalidades ahora están disponibles para distintas plataformas, como ejemplo tenemos GitHub desktop, software que permite gestionar repositorios locales y sincronización remota.

También está la posibilidad de utilizar otros softwares con GUI de control de versiones, los cuales se pueden conectar con GitHub, como GitCraken.