1. Git es un software de versiones diseñado por linus torvalds, pensando en la eficiencia, la confiabilidad y compatibilidad del mantenimiento de aplicaciones cuando estas tienen un gran número de archivos de código 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.

2. ¿A quién pertenece actualmente GitHub? El 4 de junio de 2018, [Microsoft](https://es.wikipedia.org/wiki/Microsoft) compró GitHub por la cantidad de 7500 millones de dólares. Al inicio, el cambio de propietario generó preocupaciones y la salida de algunos proyectos de este sitio;[3](https://es.wikipedia.org/wiki/GitHub#cite_note-3)​ sin embargo, no fueron representativos. GitHub continúa siendo la plataforma más importante de colaboración para proyectos de [código abierto](https://es.wikipedia.org/wiki/C%C3%B3digo_abierto).

¿Por qué Microsoft? Desde el nombramiento de Satya Nadella como CEO de Microsoft (sustituyendo al Steve Ballmer), la compañía ha adoptado una estrategia nueva, mucho más abierta al mundo del software libre y a la integración con otras plataformas y servicios. Un ejemplo es la adquisición en 2016 de [Xamarin](https://www.xamarin.com/), una plataforma que permite crear aplicaciones móviles que pueden funcionar en distintos sistemas, como iOS, Android y (por supuesto) Windows. Y es, precisamente, el que fuera CEO de Xamarin, Nat Friedman, quien será nuevo CEO de GitHub. Así pues, esta adquisición es, sin duda, un acercamiento a la comunidad más vibrante en el desarrollo de software libre y abierto.

Además, hay otro motivo de peso: [Microsoft Azure](https://azure.microsoft.com/). 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.

3. Si, hay otras formas de trabajar con Git que no son a través de una terminal. Entre ellas, según leí en internet, destacan Tower, Fork y GitHub Desktop.