

19 de mayo de 2021

Resumen Curso de Git

Sebastián Arrazate Martínez

Hoy en día, Git es el sistema de control de versiones moderno más utilizado en el mundo, es un proyecto de código abierto muy probado y con mantenimiento activo. Fue desarrollado originalmente por Linus Torvalds, el famoso creador del kernel del sistema operativo Linux, en 2005.

En la actualidad Git se utiliza en su mayoría para diseñar cualquier sitio web, software o código de forma rápida e inteligente controlando el trabajo en equipo de manera que disminuye el riesgo de perder informaciones, sea por error humano o simplemente sobrescribiendo un código. Es aquí en donde entra mi primer comentario sobre lo aprendido en el curso y es que yo desconocía el hecho de que Git no es utilizado únicamente para el control de versiones de código (Sitios web, Software, Scripts, etc.), sino que es capaz de trabajar con archivos de texto, ediciones de diferentes tipos, etc.

Por otro lado, existen compañías que ofrecen servicios de hosting (almacenamiento) de repositorios para almacenarlos en la nube, lo cual facilita a los desarrolladores el acceso a estos y también facilita el uso de Git para el control de versiones y colaboración. La empresa mas conocida y con ventajas por ser de código abierto es GitHub, s bastante fácil de usar para el desarrollador novato que quiera aprovechar las ventajas del Git. Sin GitHub, usar un Git generalmente requiere de un poco más de conocimientos de tecnología y uso de una línea de comando. Existen otras plataformas para el controlado de versiones que trabajan con Git como:

- GitLab
- SourceForge
- Cloud Source Repositories
- GitKraken

En Git hay una comunidad muy grande de desarrolladores que suelen aportar conocimientos en distintos proyectos ayudando al desarrollo de estos. Para poder hacer esto, Git nos da la posibilidad de crear copias de los repositorios (Carpeta del proyecto) almacenados en un servidor, esto se logra haciendo uso de comandos que proporciona

Git (git clone URL) para el manejo de los repositorios, desde la descarga del repositorio hasta las solicitudes de cambios, agregar cambios, generar ramas de trabajo alternas, mezclar ramas, etc. Como administrador de un repositorio existe la posibilidad de otorgar permisos y privilegios sobre estos dependiendo de tus necesidades y conveniencia.

Algunos de los comandos básicos de Git son:

1. git branch <nombre-rama>: crea una nueva rama en repositorio local
2. git push <nombre-remoto> <nombre-rama>: envía la nueva rama a repositorio remoto
3. git checkout <nombre-de-la-rama>: cambiar de una rama a otra
4. git status: nos da información necesaria sobre la rama actual
5. git add <archivo>: incluir cambios del archivo al próximo commit
6. git commit -m "mensaje de confirmación": establecer un punto de control en el proceso de desarrollo al cual puedes volver más tarde si es necesario y agregamos mensaje que explica lo que se ha desarrollado o modificado.
7. git push <nombre-remoto> <nombre-de-tu-rama>: envía tus commits al repositorio remoto.
8. git pull <nombre-remoto>: recibir actualizaciones del repositorio remoto.

Me gustaría agregar como ultimo comentario que me di cuenta de que el uso de Git es mucho mas sencillo desde sistema operativo Linux debido a el sencillo uso de sus editores de texto como nano, a su sencillo manejo de sistemas de archivos, las descargas de librerías y muchas otras cosas más que facilitan el trabajo en estos sistemas operativos. En Windows es más complicado desde el momento de acceder a la ruta del repositorio ya es más trabajo y por momentos puede llegar a ser más confuso, además, cabe mencionar que la mayoría de la documentación de Git es para sistemas Linux.