**¿QUE ES GIT?**

**Git es un sistema de control de versiones de software que permite llevar un registro de los cambios en archivos de un proyecto de software y coordinar el trabajo de múltiples desarrolladores en un proyecto. Con Git, los desarrolladores pueden trabajar en paralelo, realizar versiones de sus cambios y resolver conflictos de forma eficiente. Es ampliamente utilizado por desarrolladores de todo el mundo y es uno de los sistemas de control de versiones más populares en la industria. GitHub utiliza Git en la parte técnica de su plataforma y les permite a los usuarios alojar y compartir sus proyectos de software con otros.**

**Kinsta. (s.f.). Qué es GitHub. Recuperado el 7 de febrero de 2023, de** [**https://kinsta.com/es/base-de-conocimiento/que-es-github/**](https://kinsta.com/es/base-de-conocimiento/que-es-github/)

**Microsoft Learn. (s.f.). ¿Qué es Git? Recuperado el 7 de febrero de 2023, de** [**https://learn.microsoft.com/es-es/devops/develop/git/what-is-git**](https://learn.microsoft.com/es-es/devops/develop/git/what-is-git)

**Objetos Git.**

**En Git, los objetos son los elementos básicos que componen un repositorio de Git. Hay tres tipos principales de objetos en Git:**

**Blobs: Representan los archivos y su contenido en un repositorio.**

Árboles: Representan la estructura de archivos y directorios en un repositorio.

Commits: Representan un punto en la historia del repositorio y contienen información sobre quién hizo los cambios, cuándo se hicieron y un puntero al árbol de archivos correspondiente.

Estos objetos son identificados por un identificador único de hash SHA-1 y se almacenan en el sistema de archivos de Git en forma comprimida. Git utiliza estos objetos para llevar un seguimiento de los cambios en el repositorio y permitir la colaboración y la gestión de versiones.

Git Community Book. (s.f.). Los entresijos internos de Git: Los objetos Git. Recuperado el 7 de febrero de 2023, de <https://git-scm.com/book/es/v2/Los-entresijos-internos-de-Git-Los-objetos-Git>

**¿Qué es GITHUB?**

es una plataforma web que ofrece servicios de control de versiones y alojamiento para proyectos de software. Permite a los desarrolladores colaborar, compartir y gestionar su código fácilmente. Además, ofrece una amplia variedad de herramientas y funciones para el seguimiento de problemas, la revisión de código y la gestión de proyectos. GitHub es ampliamente utilizado por desarrolladores de todo el mundo y es una de las plataformas de control de versiones más populares en la industria.

WebEmpresa. (s.f.). ¿Qué es GitHub? Recuperado el 7 de febrero de 2023, de <https://www.webempresa.com/hosting/que-es-github.html>

Xataka. (s.f.). ¿Qué es GitHub y qué les ofrece a los desarrolladores? Recuperado el 7 de febrero de 2023, de <https://www.xataka.com/basics/que-github-que-que-le-ofrece-a-desarrolladores>

**Git&GitHub, Setup?**

PASO 1: Instalar Git está disponible para Windows, MacOS y Linux, puedes descargar la versión adecuada para tu sistema operativo desde la página web oficial de Git.

PASO 2: Configuración de su nombre de usuario y correo electrónico en Git. Una vez que haya instalado git en su máquina, ahora debe informar a git quién es usted.

Debe darle a git su nombre y correo electrónico.

Abra su línea de comandos, ejecute esto con su nombre, para configurar su nombre

git config --global user.name "Your Name".

Paso 3: Y ejecute esto con su correo electrónico, para configurar su correo electrónico

git config --global user.email "yourname@example.com"

Cambie el maestro predeterminado a main:

En 2020, GitHub cambió el nombre maestro de rama predeterminado a principal.

Cualquier repositorio nuevo que cree usará main como rama predeterminada, en lugar de máster.

Pero si crea un repositorio con **git init**, entonces hará un maestro de rama.

Para solucionarlo podemos configurar **git** para crear un repositorio con un nombre principal con este comando

**git config --global init.defaultBranch main**

Nuestra configuración local ha finalizado. Puede ejecutar estos comandos para ver que las cosas funcionan correctamente.

Debería poder ver su correo electrónico y nombre de usuario.

**git config --get user.name**

**git config --get user.email**

**Paso #3: Generar clave SSH**

Configuramos el git, ahora necesitamos generar una clave SSH y agregarla a nuestra cuenta de Github.

**Primero, necesitamos ver si tenemos una clave SSH con este comando:**

**ls ~/.ssh/id\_rsa.pub**

Podemos generar una clave con este comando:

**ssh-keygen -C yourEmailHere**

Generamos la clave, ahora necesitamos imprimir en la pantalla y copiarla para que podamos agregarla a Github:

Para imprimir necesitamos ejecutar este comando:

**cat ~/.ssh/id\_rsa.pub**

**Paso #4: Agregar clave SSH a GitHub:**

Ahora necesitamos agregar la clave a GitHub.

Vaya a Github, haga clic en su foto de perfil y luego vaya a configuración.

***Todos los comandos:***

***git config --global user.name "Your Name"***

***git config --global user.email "yourname@example.com"***

***git config --global init.defaultBranch main***

***git config --get user.name***

***git config --get user.email***

***ls ~/.ssh/id\_rsa.pub***

***ssh-keygen -C <youremail>***

***cat ~/.ssh/id\_rsa.pub***

***ssh -T*** [***git@github.com***](mailto:git@github.com)

**Set uP configuration.**

**Instalar Git:** Descarga e instala la última versión de Git en tu computadora desde la página oficial de Git.

**Configurar tu nombre de usuario:** Ejecuta el siguiente comando en la línea de comandos para configurar tu nombre de usuario:

**$ git config --global user.name "Tu nombre"**

**Configurar tu dirección de correo electrónico:** Ejecuta el siguiente comando en la línea de comandos para configurar tu dirección de correo electrónico:

**$ git config --global user.email "tu@correo.com"**

**Configurar el editor de texto predeterminado:** Ejecuta el siguiente comando en la línea de comandos para configurar el editor de texto predeterminado:

**$ git config --global core.editor nano**

Puedes reemplazar "nano" con el editor de texto que prefieras usar.

Estos son los pasos básicos para configurar Git en tu computadora. También puedes configurar otras opciones de Git, como el color de la salida o la configuración de merge.

Conclusiones.

En conclusión:

* Git es un sistema de control de versiones distribuido que permite a los desarrolladores llevar un registro de cambios en el código de un proyecto y colaborar de manera eficiente.
* GitHub es una plataforma web que brinda servicios de alojamiento y gestión de repositorios de Git, permitiendo a los desarrolladores colaborar y trabajar juntos en proyectos.
* La configuración de Git y GitHub incluye la instalación de Git en la computadora, la configuración de tu nombre de usuario y dirección de correo electrónico, y la configuración de la clave SSH para conectarte de manera segura a GitHub.
* Los objetos de Git son elementos básicos que forman la estructura de un repositorio de Git, y se utilizan para almacenar y gestionar los cambios en el código.

Es importante tener en cuenta que Git y GitHub son herramientas avanzadas y requieren una cierta cantidad de conocimiento técnico para usarlas de manera efectiva. Sin embargo, una vez que se comprende su funcionamiento, pueden ser muy útiles para los desarrolladores y facilitar el proceso de colaboración y desarrollo de software.