**GIT**

**GIT** es uno **sistema de control de versiones distribuido**, creado por **Linus Torvalds** (el mismo creador de Linux) en 2005. Permite a los desarrolladores rastrear los cambios en el código fuente a lo largo del tiempo, colaborar de manera eficiente y revertir los cambios si es necesario.

**Objetivos clave de GIT**

* Rastree las modificaciones a los archivos a lo largo del tiempo.
* Trabajar en equipo sin sobrescribir el trabajo de los demás.
* Habilite el versionado de proyectos con seguridad e integridad.

**¿Cómo Funciona?**

GIT opera de alguna manera **distribuido**res decir, cada desarrollador tiene una copia completa del repositorio localmente, lo que permite:

* Trabajando fuera de línea.
* Realizar commits locales.
* Sincronizar cambios con un repositorio remoto (por ejemplo, GitHub, GitLab, Bitbucket).

**Estructura Básica**

* **Repositorio (Repositorio):** Donde se enmarca el código y su sonido histórico.
* **Compromiso:** Registro de un cambio en el proyecto.
* **Sucursal (sucursal):** Línea paralela de desarrollo.
* **Combinar:** Combinacion de sucursales.
* **Clon:** Copie desde un repositorio remoto a la computadora local.
* **Solicitud de Tiro/Tiro:** Traiga cambios desde el control remoto o solicite la inclusión de nuevos cambios.

**GIT Comandos principales**

git init # Crear un nuevo repositorio

git clone URL # Clonar un repositorio existente

git add . # Agregar archivos al área de preparación

git commit -m "mensaje" # Guardar cambios con un mensaje

git status # Ver estado de los archivos

git log # Ver historial de commits

git branch # Ver ramas existentes

git checkout nombre\_rama # Cambiar de rama

git merge nombre\_rama # Unir otra rama con la actual

git pull # Traer cambios del remoto

git push # Enviar cambios al remoto

**Ventajas de GIT**

* Velocidad y eficiencia.
* Seguridad de integridad de datos.
* Admite múltiples flujos de trabajo de desarrollo.
* Historial completo de cambios.
* Gran comunidad e integrar como diversas herramientas.

**Plataformas Populares que usan GIT**

* **GitHub**
* **GitLab**
* **Bitbucket**
* **DevOps Azure**

### ****Comandos básicos de Linux en la terminal****

| **Comando** | **Descripción** |
| --- | --- |
| ls | Lista los archivos y carpetas del directorio actual. |
| cd | Cambia de directorio (por ejemplo: cd Documentos). |
| pwd | Muestra la ruta del directorio actual. |
| mkdir | Crea un nuevo directorio (carpeta). |
| rm | Elimina archivos o carpetas. Usa con cuidado. Ejemplo: rm archivo.txt o rm -r carpeta/ para carpetas. |
| cp | Copia archivos o carpetas. Ejemplo: cp archivo.txt copia.txt |
| mv | Mueve o renombra archivos. Ejemplo: mv viejo.txt nuevo.txt |
| touch | Crea un archivo vacío. Ejemplo: touch nuevo.txt |
| cat | Muestra el contenido de un archivo. |
| nano | Abre un editor de texto en la terminal. Ejemplo: nano archivo.txt |
| clear | Limpia la pantalla de la terminal. |
| exit | Cierra la terminal o sesión actual. |

### ****Comandos de búsqueda****

| **Comando** | **Descripción** |
| --- | --- |
| find | Busca archivos. Ejemplo: find / -name archivo.txt |
| grep | Busca texto dentro de archivos. Ejemplo: grep "palabra" archivo.txt |

### ****Permisos y usuarios****

| **Comando** | **Descripción** |
| --- | --- |
| chmod | Cambia los permisos de un archivo. Ejemplo: chmod 755 script.sh |
| chown | Cambia el propietario de un archivo. |
| sudo | Ejecuta un comando con permisos de administrador. Ejemplo: sudo apt update |

### ****Sistema y archivos****

| **Comando** | **Descripción** |
| --- | --- |
| df -h | Muestra el uso del disco duro. |
| du -h | Muestra el tamaño de archivos o carpetas. |
| top | Muestra procesos activos y consumo de recursos. |
| ps aux | Muestra todos los procesos en ejecución. |
| kill | Mata un proceso. Ejemplo: kill 1234 (donde 1234 es el ID del proceso). |

### ****Red****

| **Comando** | **Descripción** |
| --- | --- |
| ping | Prueba la conexión con una dirección. Ejemplo: ping google.com |
| ifconfig o ip a | Muestra la configuración de red. |
| wget | Descarga archivos desde la web. Ejemplo: wget https://ejemplo.com/archivo.zip |
| curl | Transferencia de datos desde o hacia un servidor. Muy útil para APIs. |