Programación Web – 7 GIT

Por: Ing. Luis Daniel Lepe Rodríguez

Git es, con diferencia, el sistema de control de versiones moderno más utilizado del mundo.

Un asombroso número de proyectos de software dependen de Git para el control de versiones, incluidos proyectos comerciales y de código abierto.

Los desarrolladores que han trabajado con Git cuentan con una buena representación en la base de talentos disponibles para el desarrollo de software, y este sistema funciona a la perfección en una amplia variedad de sistemas operativos e IDE.

Sistemas de control de versiones

- Los **sistemas de control de versiones** (VCS) son programas cuyo objetivo es controlar los cambios en el desarrollo de cualquier tipo de software.

Características de Git

Es un sistema de control de versiones distribuido.

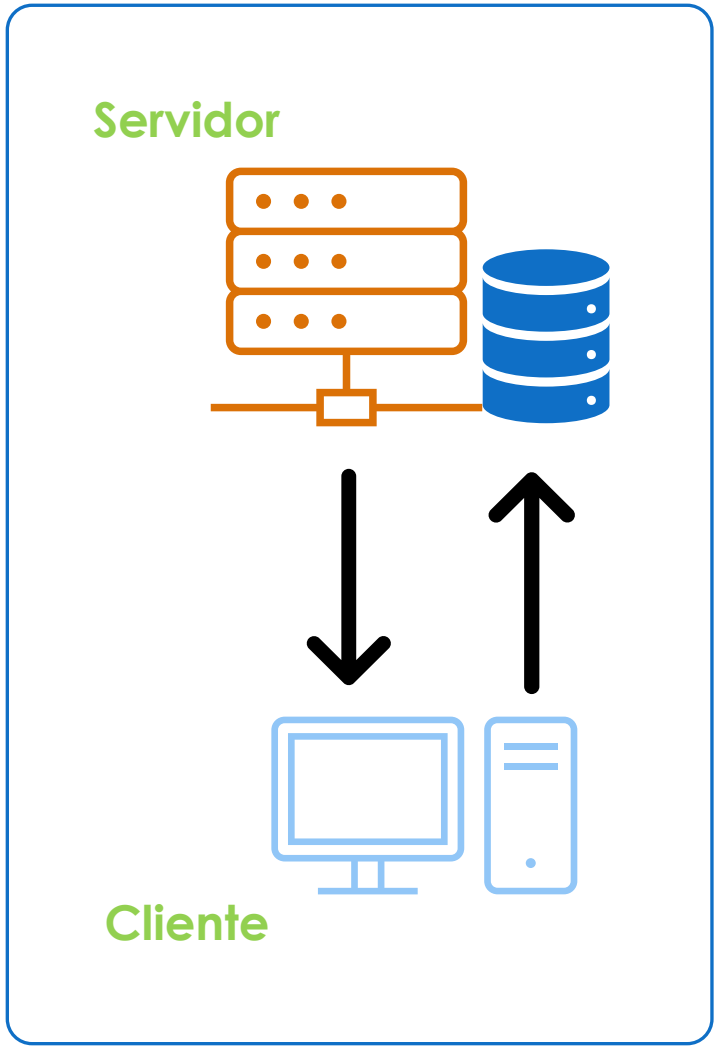
Es Open Source y multiplataforma.

Puede ser utilizado de manera offline.

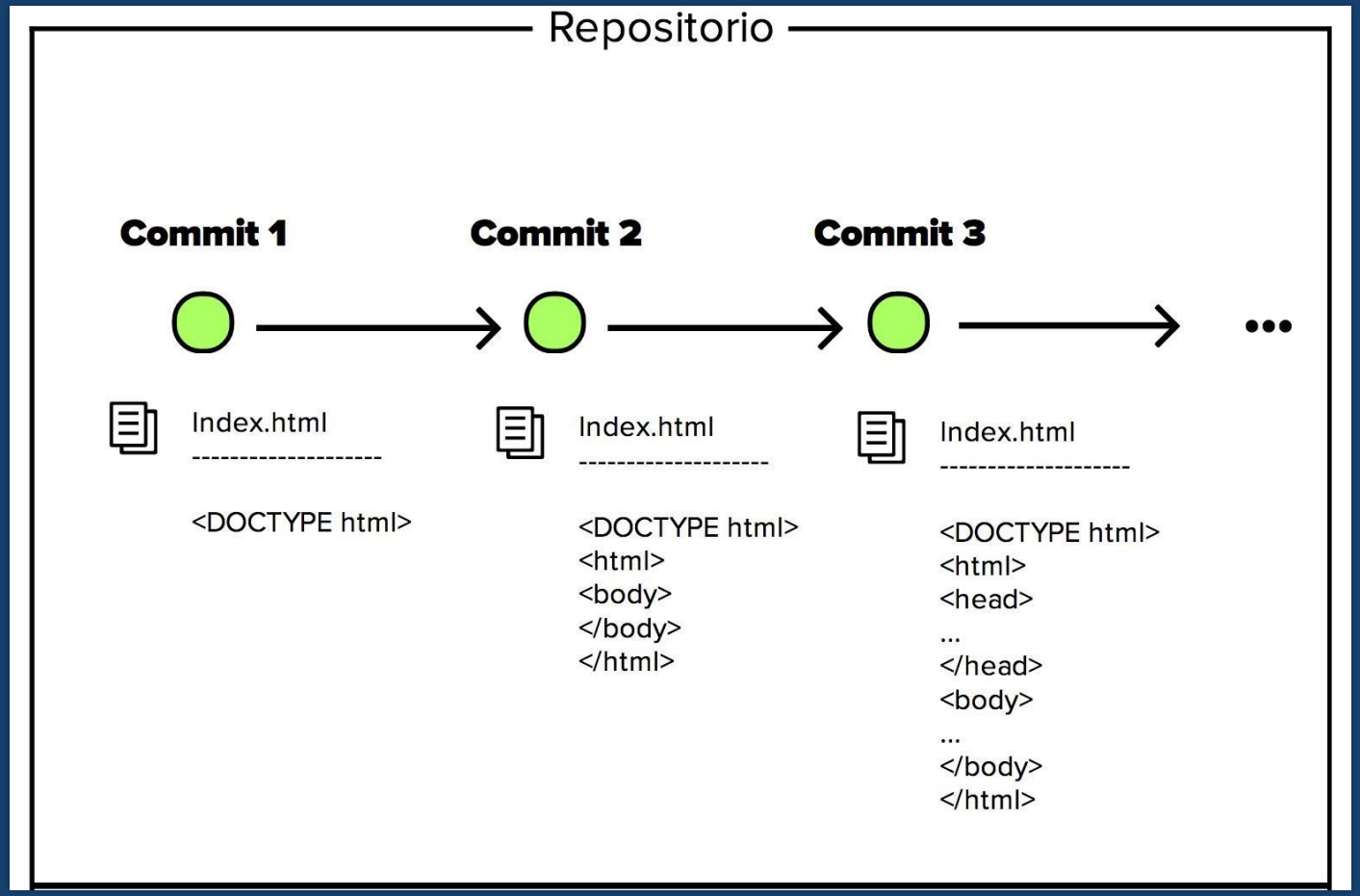
¿Cómo funciona Git?

Existen dos entidades, el **cliente** y el **servidor**:

- ***Servidor***: gestiona los distintos proyectos.
- ***Cliente***: es el software que instalamos para trabajar con el código, hacer peticiones al servidor, etc.



Cómo gestiona Git los cambios realizados



Ventajas de usar Git

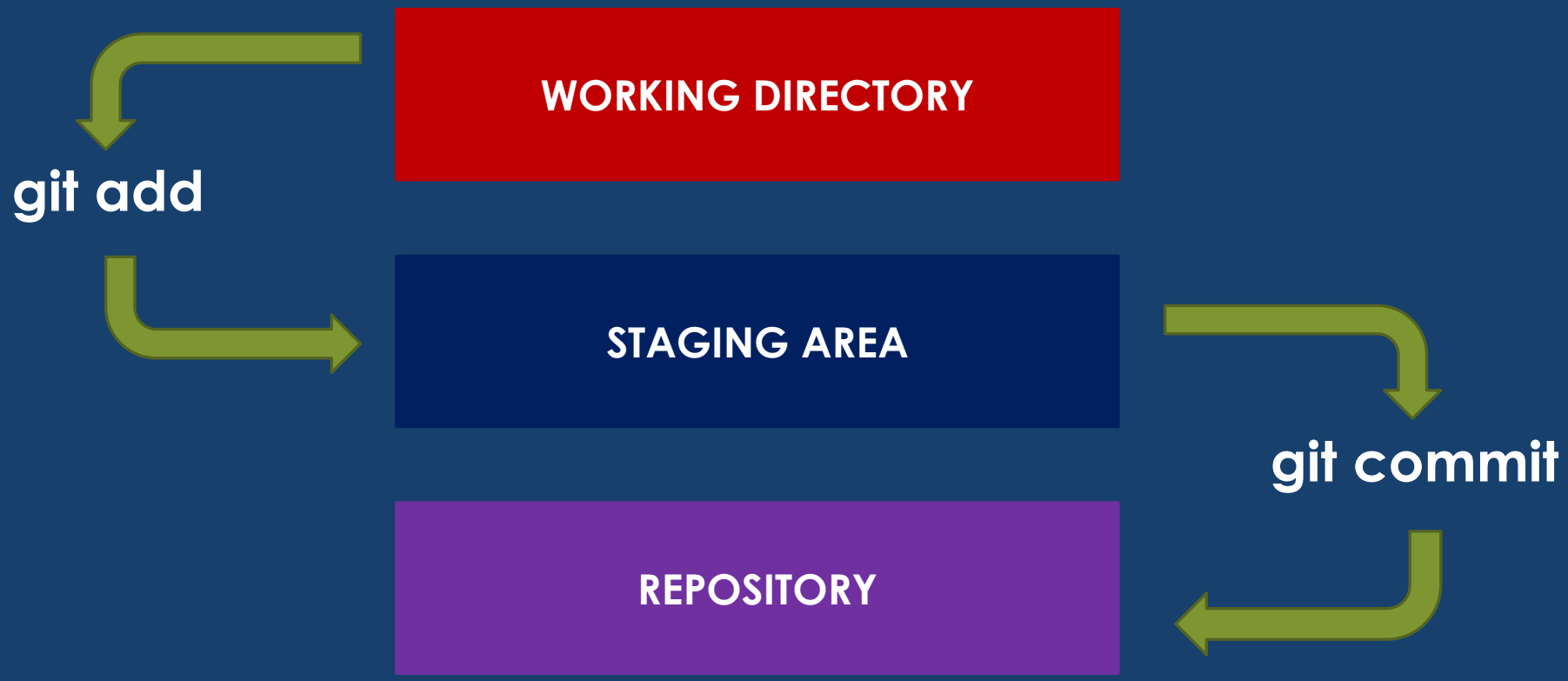
FACILITA EL TRABAJO COLABORATIVO.

CREA MÚLTIPLES VERSIONES Y SE PUEDE VOLVER A UNA VERSIÓN ANTERIOR.

REDUCE LOS TIEMPOS DE DESPLIEGUE DE UN PROYECTO.

PERMITE GENERAR FLUJOS DE TRABAJO EFICIENTE.

Áreas de trabajo en Git



GitHub

- Es una plataforma que sirve para alojar proyectos **utilizando** Git como sistema de control de versiones.

Diferencias entre Git y GitHub

- Git y GitHub se complementan, pero son personajes **independientes**

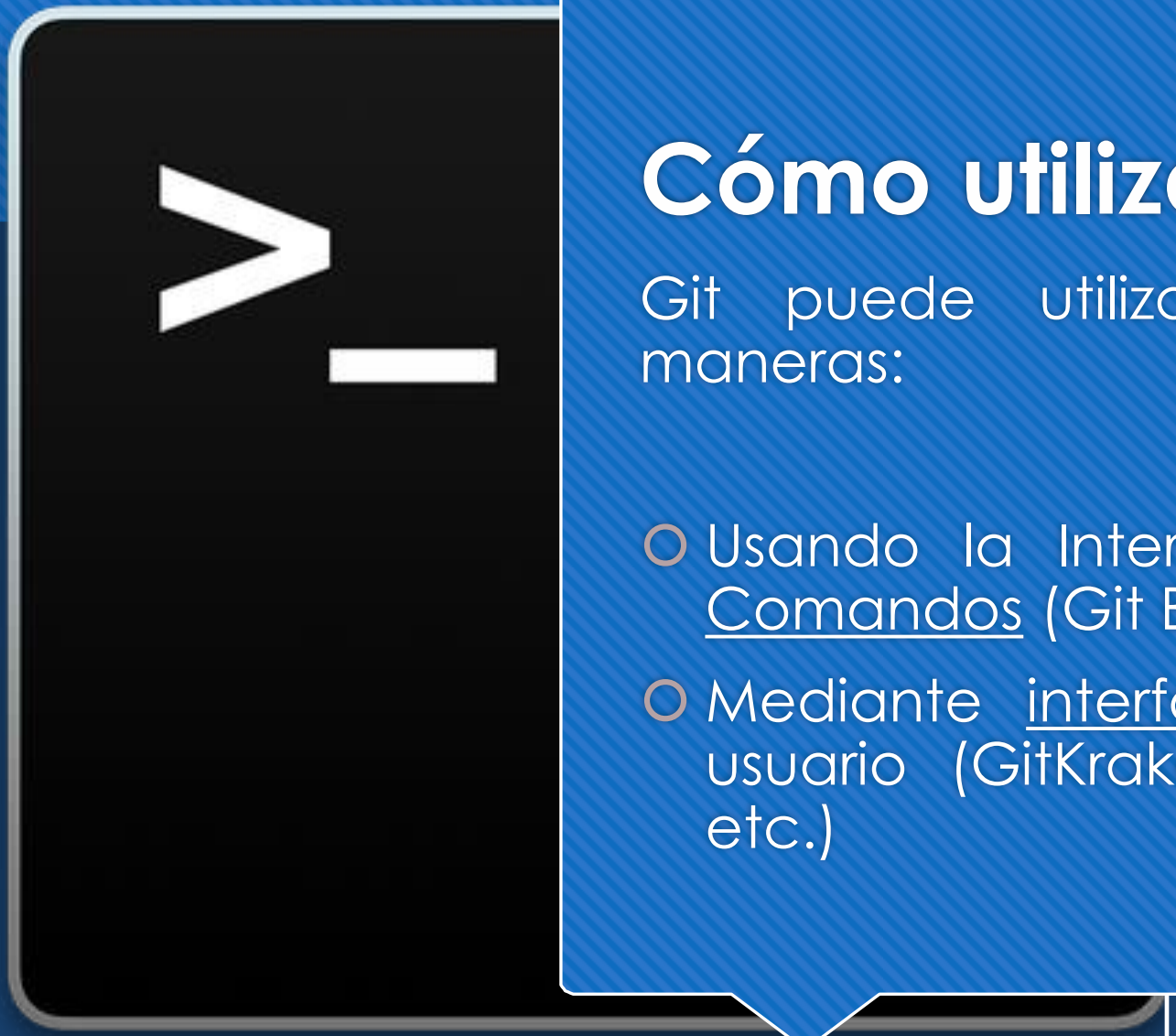
Git	GitHub
• Es un sistema de control de versiones. • Por sí solo puede utilizarse como SCV local.	• Es una plataforma de Internet en donde se alojan proyectos. • Incorpora características de Git para trabajar con repositorios de manera colaborativa.

Terminología

CONCEPTO	DESCRIPCIÓN
Repositorio (Repository)	Almacén digital para gestionar y almacenar versiones de código fuente, permitiendo control sobre cambios y colaboración.
Cambio (Commit)	Registro de cambios confirmados en el repositorio, cada uno con un identificador único, un mensaje descriptivo, hora de realización y quién hizo el cambio.
Puntero (HEAD)	Es un puntero que hace referencia al commit actual, usualmente es el último commit realizado en la rama pero este puede moverse a commits antiguos para ver el código en diferentes puntos históricos del proyecto.

Terminología

CONCEPTO	DESCRIPCIÓN
Rama (Branch)	Versión independiente del código base, usada para desarrollar funcionalidades o correcciones sin afectar la línea principal.
Unión (Merge)	Proceso de integrar o fusionar cambios de una rama a otra, combinando dos historiales de desarrollo.
Conflicto (Conflict)	Ocurre cuando el sistema no puede manejar adecuadamente cambios realizados por dos o más usuarios en un mismo archivo.



Cómo utilizarlo

Git puede utilizarse de distintas maneras:

- Usando la Interfaz de Línea de Comandos (Git Bash).
- Mediante interfaces gráficas de usuario (GitKraken, Visual Studio, etc.)

Git Bash

Git Bash es una aplicación para entornos de Microsoft Windows que ofrece una capa de emulación para una experiencia de líneas de comandos de Git.

Comandos de Git Bash

COMANDO	LO QUE HACE
git init	Inicializa un repositorio el cual utilizará GIT para el control de versiones.
git status	Muestra el estado actual del repositorio, incluyendo archivos nuevos, archivos eliminados, cambios en archivos, etc.
git log	Permite ver el historial de los *commits* realizados en el proyecto.
git config	Sirve para establecer o modificar la configuración de git global o de un repositorio especifico. La configuración puede ser la información del usuario, colores de consola, etc.

Comandos de Git Bash

COMANDO	LO QUE HACE
git add	Añade los archivos con cambios del *working directory* al *staging area* preparandolos para el próximo *commit*.
git commit	Guarda los cambios realizados del *staging area* al repositorio local con un mensaje descriptivo.
git clone	Se encarga de descargar un repositorio remoto completo a un repositorio local.
git pull	Descarga las últimas actualizaciones del repositorio remoto al repositorio local.
git push	Carga los cambios del repositorio local al repositorio remoto.

Comandos de Git Bash

COMANDO	LO QUE HACE
git branch	Crea una nueva rama en el repositorio.
git merge	Permite hacer la fusión de ramas. Por lo general se aplica para fusionar una rama normal con la rama master.
Git checkout	Se utiliza para cambiar de una rama a otra, para viajar a *commits* anteriores, etc.

Tarea

- Crearse una cuenta en GitHub.
- Buscar repositorios en la plataforma y seleccionar un proyecto que les interese.
- Clonar el repositorio en su computadora e inspeccionar el código.
- Hacer un reporte con los pasos que hicieron, el repositorio que les gusto, explicar por qué seleccionaron ese repositorio y que opinan del código de ese proyecto ¿Lo entienden? ¿Es sencillo? ¿Podrían ustedes hacer algo similar?.