

Práctico 2: Git y GitHub

Objetivo:

El estudiante desarrollará competencias para trabajar con Git y GitHub, aplicando conceptos fundamentales de control de versiones, colaboración en proyectos y resolución de conflictos, en un entorno simulado y guiado.

Resultados de aprendizaje:

1. Comprender los conceptos básicos de Git y GitHub: Identificar y explicar los principales términos y procesos asociados con Git y GitHub, como repositorios, ramas, commits, forks, etiquetas y repositorios remotos.
2. Manejar comandos esenciales de Git: Ejecutar comandos básicos para crear, modificar, fusionar y gestionar ramas, commits y repositorios, tanto en local como en remoto.
3. Aplicar técnicas de colaboración en GitHub: Configurar y utilizar repositorios remotos, realizar forks, y gestionar pull requests para facilitar el trabajo colaborativo.
4. Resolver conflictos en un entorno de control de versiones: Identificar, analizar y solucionar conflictos de merge generados en un flujo de trabajo con múltiples ramas.

Actividades

- 1) Contestar las siguientes preguntas utilizando las guías y documentación proporcionada (Desarrollar las respuestas) :

- ¿Qué es GitHub?

GitHub es una plataforma en línea para almacenar proyectos que utilizan el sistema de control de versiones Git. Es el servicio más popular para colaborar en proyectos de software tanto públicos como privados. Permite compartir código, colaborar entre desarrolladores, revisar cambios, organizar tareas y mucho más.

- ¿Cómo crear un repositorio en GitHub?

1. Ingresar a <https://github.com>.
2. Hacer clic en el botón de crear nuevo repositorio.
3. Completar nombre, descripción y otras configuraciones como visibilidad.

- ¿Cómo crear una rama en Git?

En la carpeta donde se encuentra el proyecto, escribimos en línea de comandos:
git branch <nombre_branch>

- ¿Cómo cambiar a una rama en Git?

En la carpeta donde está el proyecto, escribimos en línea de comandos:
git checkout <nombre_branch>

- ¿Cómo fusionar ramas en Git?

En la carpeta donde está el proyecto, escribimos en línea de comandos **git merge <nombre_rama>**, lo que ocasiona que realiza una fusión o merge con la rama actual en la que estamos trabajando con la rama <nombre_rama>

- ¿Cómo crear un commit en Git?

Agregar los archivos al área de preparación (staging):

git add archivo

Crear el commit:

git commit -m "Mensaje descriptivo"

- ¿Cómo enviar un commit a GitHub?

Luego de tener el repositorio remoto vinculado:

git push origin main

Esto sube los commits locales al repositorio en GitHub.

- ¿Qué es un repositorio remoto?

Es un repositorio que está almacenado en un servidor externo (como GitHub). Permite compartir código entre distintos desarrolladores o equipos.

- ¿Cómo agregar un repositorio remoto a Git?

Se ejecuta el comando `git clone <repositorio_remoto>` hay varias formas en las cuales se puede utilizar, que dependen de las capacidades o funcionalidades del sitio donde esté alojado el repositorio, en el caso de github tenemos tres formas, utilizando HTTPS `git clone <url_proyecto>` requiere usar usuario y contraseña, con SSH para lo que se necesita vincular la cuenta con las Keys locales o bien utilizar el cliente oficial de github.

- ¿Cómo empujar cambios a un repositorio remoto?

En este caso, asumiendo que ya tenemos vinculado nuestro repositorio local con github, se ejecuta el comando `git push origin <nombre_de_rama>`, previamente agregando los cambios con el comando `add` y ejecutando luego el comando `commit`, combinando ambas instrucciones en una sola línea `git commit -am "mensaje"`

- ¿Cómo tirar de cambios de un repositorio remoto?

Para ello se utiliza el comando `pull`, parado en el repositorio local, puedo ejecutar por ejemplo `git pull origin main`, donde `origin` es el nombre asignado al repositorio remoto y `main` el nombre de la rama remota.

- ¿Qué es un fork de repositorio?

Un fork es una **copia completa de un repositorio de GitHub** dentro de tu propia cuenta. Te permite hacer cambios sin afectar el proyecto original.

- ¿Cómo crear un fork de un repositorio?

1. Ir al repositorio que se desea forkear.
2. Hacer clic en el botón **Fork** en la esquina superior derecha.
3. Elegir tu cuenta.

- ¿Cómo enviar una solicitud de extracción (pull request) a un repositorio?

1. Subir los cambios a tu fork.
2. Ir al repositorio original.
3. Hacer clic en "New Pull Request".

4. Comparar tus cambios y proponer la fusión.

- ¿Cómo aceptar una solicitud de extracción?

Se realiza al revisar el Pull Request y decidir si se acepta o no.

- ¿Qué es una etiqueta en Git?

Las etiquetas (tags) son marcas que se crean para referenciar de manera especial ciertos commits. Generalmente se usan para marcar versiones específicas.

- ¿Cómo crear una etiqueta en Git?

Hay dos tipos de etiquetas, notada o ligeras, la diferencia entre la segunda y la primera es que la primera es más completa y contiene información, como descripción y datos del autor

- Ligera : `git tag nombre-etiqueta`
- Anotada: `git tag -a nombre-etiqueta -m "Mensaje"`

- ¿Cómo enviar una etiqueta a GitHub?

Para hacerlo se utiliza esta línea de comandos: `git push origin nombre-etiqueta`

- ¿Qué es un historial de Git?

Es la secuencia de cambios registrados en un repositorio

- ¿Cómo ver el historial de Git?

el comando es `git log`

- ¿Cómo buscar en el historial de Git?

`git log` permite buscar con opciones como `--grep` para buscar mensajes de commits.

- ¿Cómo borrar el historial de Git?

Si no hemos subido los cambios al repositorio remoto podemos hacer con el siguiente comando `git reset --hard HEAD~n` siendo `n` al número de commits locales hacia atrás que quiero eliminar

- ¿Qué es un repositorio privado en GitHub?

Un repositorio privado es uno que solo puede ser visto por las personas invitadas, ideal para proyectos privados o comerciales.

- ¿Cómo crear un repositorio privado en GitHub?

1. Crear un repositorio en GitHub.
2. Elegir la opción de visibilidad Private.

- ¿Cómo invitar a alguien a un repositorio privado en GitHub?

Esto se realiza desde GitHub → Settings → Collaborators and Teams → Invitar colaborador.

- ¿Qué es un repositorio público en GitHub?

Un repositorio público es visible para cualquier persona en internet. Ideal para proyectos open source.

- ¿Cómo crear un repositorio público en GitHub?

1. Crear un repositorio en GitHub.
2. Elegir la opción de visibilidad Public.

- ¿Cómo compartir un repositorio público en GitHub?

Compartiendo el enlace del repositorio (URL), cualquier persona podrá acceder a él.