

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?

Es una plataforma que ofrece alojamiento de repositorios de control de versiones, que permite a los desarrolladores almacenar y gestionar sus proyectos de software. Es una herramienta gratuita y de código abierto que permite el trabajo en equipo en proyectos, el intercambio de código y el trabajo conjunto de forma eficiente.

- ¿Cómo crear un repositorio en GitHub?

Pueden hacerlo con el comando `git init`

- ¿Cómo crear una rama en Git?

Se crean utilizando el comando `git checkout -b nombre-rama`

- ¿Cómo cambiar a una rama en Git?

Con el comando de `git checkout`

- ¿Cómo fusionar ramas en Git?

Se fusionan utilizando el comando `git merge`

- ¿Cómo crear un commit en Git?

Se crean mediante el comando `git commit -m "mensaje"`

- ¿Cómo enviar un commit a GitHub?

`git add .`

- ¿Qué es un repositorio remoto?

Un repositorio remoto es una versión del proyecto que se encuentra alojada en un servidor externo (por ejemplo, GitHub), permitiendo compartirlo con otros desarrolladores.

- ¿Cómo agregar un repositorio remoto a Git?

Se realiza mediante el comando `git remote add origin URL-del-repositorio`

- ¿Cómo empujar cambios a un repositorio remoto?

Con el comando `git push origin nombre-de-la-rama`

- ¿Cómo tirar de cambios de un repositorio remoto?

`git pull origin nombre-de-la-rama`

- ¿Qué es un fork de repositorio?

Un fork es una copia de un repositorio público que se hace en la cuenta personal de GitHub, permitiendo realizar modificaciones sin afectar el original.

- ¿Cómo crear un fork de un repositorio?

- Ingresar al repositorio original en GitHub.
- Hacer clic en el botón “Fork” (arriba a la derecha).
- Seleccionar la cuenta de destino.

- ¿Cómo enviar una solicitud de extracción (pull request) a un repositorio?
 - Ingresar al repositorio original.
 - Hacer clic en “Pull request”.
 - Comparar ramas y enviar solicitud describiendo los cambios.
- ¿Cómo aceptar una solicitud de extracción?
 - Revisar la pull request.
 - Hacer clic en “Merge pull request”.
 - Confirmar la fusión.
- ¿Qué es un etiqueta en Git?

Una etiqueta (tag) en Git se utiliza para marcar versiones específicas o importantes dentro de un repositorio, como por ejemplo una versión estable.
- ¿Cómo crear una etiqueta en Git?

Se utiliza el siguiente comando `git tag nombre-de-la-etiqueta`
- ¿Cómo enviar una etiqueta a GitHub?

para subir la etiqueta:

`git push origin nombre-de-la-etiqueta`

O todas las etiquetas:

`git push origin --tags`

- ¿Qué es un historial de Git?

Es el registro de todos los commits realizados en un repositorio, permitiendo ver los cambios, las fechas y los autores.
- ¿Cómo ver el historial de Git?

Con el siguiente comando: `git log`
- ¿Cómo buscar en el historial de Git?

Se busca con `git log --grep="palabra-clave"`

 - ¿Cómo borrar el historial de Git?

Utilizando los siguientes comando puedes borrar el historial local pero no el remoto:

```
rm -rf .git
```

```
git init
```
- ¿Qué es un repositorio privado en GitHub?

Es un repositorio que sólo puede ser visto y accedido por las personas que el dueño autorice.
- ¿Cómo crear un repositorio privado en GitHub?

Al crear un nuevo repositorio, se selecciona la opción “Private” antes de crearlo.

- ¿Cómo invitar a alguien a un repositorio privado en GitHub?
 - Entrar al repositorio.
 - Ir a “Settings” → “Manage access”.
 - Invitar por nombre de usuario o mail.

- ¿Qué es un repositorio público en GitHub?

Es un repositorio que puede ser visto y clonado por cualquier persona en internet.

- ¿Cómo crear un repositorio público en GitHub?

En el proceso de creación, elegir la opción “Public”.

- ¿Cómo compartir un repositorio público en GitHub?

Simplemente se comparte el enlace del repositorio que aparece en la barra de direcciones del navegador.

2) Realizar la siguiente actividad: <https://github.com/IgnacioRoveres/tp-n2-ejercicio2.git>

3) Realizar la siguiente actividad: <https://github.com/IgnacioRoveres/conflict-exercise.git>