

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 herramienta que permite almacenar repositorios y, por lo tanto, gestionar y compartir código fuente. Al basarse en Git permite, además, contar con un sistema de control de versiones.

- ¿Cómo crear un repositorio en GitHub?

- 1- Acceder a github.com
- 2- Acceder a la opción "New repository"
- 3- Designar un nombre al repositorio
- 4- Indicar si el repositorio será público o privado
- 5- Crear repositorio con la opción "Create Repository"

- ¿Cómo crear una rama en Git?

En la terminal debe ejecutarse el comando `git branch nombreRama`

- ¿Cómo cambiar a una rama en Git?

En la terminal debe ejecutarse el comando `git checkout nombreRama`

- ¿Cómo fusionar ramas en Git?

Para traer el contenido de una determinada rama, estando posicionados en la rama en la que se desea la fusión debe ejecutarse el comando `git merge nombreRama`

- ¿Cómo crear un commit en Git?

En la terminal debe ejecutarse el comando `git commit -m "comentario"`

- ¿Cómo enviar un commit a GitHub?

En la terminal debe ejecutarse el comando `git push origin nombreRama`

- ¿Qué es un repositorio remoto?

Es un repositorio que se encuentra alojado en la nube (servidor de Github en nuestro caso) y que se utiliza para almacenar, compartir y colaborar en proyectos.

- ¿Cómo agregar un repositorio remoto a Git?

- 1- Acceder a github.com
- 2- Acceder a la opción "New repository"
- 3- Designar un nombre al repositorio
- 4- Indicar si el repositorio será público o privado
- 5- Crear repositorio con la opción "Create Repository"

- ¿Cómo empujar cambios a un repositorio remoto?

En la terminal deben ingresarse los siguientes comandos:

`git add .` - permite trabajar sobre los cambios de todos los archivos

`git commit -m "comentario"` – permite registrar y comentar cada cambio

`git push origin nombreRama` – permite enviar los cambios al repositorio remoto.

- ¿Cómo tirar de cambios de un repositorio remoto?

En la terminal debe ejecutarse el comando `git pull origin nombreRama`

- ¿Qué es un fork de repositorio?

Es una copia del repositorio que se crea en una cuenta diferente permitiendo aplicar cambios sin afectar el original.

- ¿Cómo crear un fork de un repositorio?

Se accede al repositorio original y se selecciona la opción “fork”.

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

Una vez realizada una modificación al repositorio a través del comando git push, dentro del repositorio de GitHub debe seleccionarse la opción Pull requests y luego New pull request. Antes de enviar la solicitud deben seleccionarse las ramas que se quieren comparar.

- ¿Cómo aceptar una solicitud de extracción?

Siendo propietario o autorizado, debe accederse a la opción Pull requests y seleccionar la solicitud correspondiente. Luego de revisar las modificaciones propuestas, los cambios pueden aceptarse con la opción Merge pull request.

- ¿Qué es una etiqueta en Git?

Las etiquetas se utilizan para marcar hitos en la historia de un repositorio

- ¿Cómo crear una etiqueta en Git?

Una etiqueta se crea con el comando git tag *nombreEtiqueta*

- ¿Cómo enviar una etiqueta a GitHub?

Con el comando git push origin *nombreEtiqueta*

- ¿Qué es un historial de Git?

Es el listado de commits que fueron realizados a lo largo de la historia del repositorio

- ¿Cómo ver el historial de Git?

Utilizando el comando git log

- ¿Cómo buscar en el historial de Git?

a) git log --author="nombre" – permite ver los commits realizados por un

determinado autor

- b) `git log --since="fecha"` – permite ver los commits realizados desde determinada fecha
- c) `git log --until="fecha"` – permite ver los commits realizados hasta determinada fecha

- ¿Cómo borrar el historial de Git?

Con el comando `git reset --hard <hashPrimerCommit >`

- ¿Qué es un repositorio privado en GitHub?

Un repositorio privado es aquel al que solo puede acceder su propietario y las personas por el él autorizadas.

- ¿Cómo crear un repositorio privado en GitHub?

1. Acceder a github.com
2. Acceder a la opción “New repository”
3. Designar un nombre al repositorio
4. Indicar la opción de repositorio privado
5. Crear repositorio con la opción “Create Repository”

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

Dentro del manejo de configuraciones del repositorio debe seleccionarse la opción “Manage access” y desde allí invitar a los usuarios que se desee con el nivel de autorización que se requiera

- ¿Qué es un repositorio público en GitHub?

Un repositorio público es aquel al que puede acceder cualquier persona, lo que implica que cualquiera puede realizar un fork del proyecto, copiarlo o clonarlo.

- ¿Cómo crear un repositorio público en GitHub?

1. Acceder a github.com
2. Acceder a la opción “New repository”
3. Designar un nombre al repositorio
4. Indicar la opción de repositorio público
5. Crear repositorio con la opción “Create Repository”

- ¿Cómo compartir un repositorio público en GitHub?

[El repositorio público puede compartirse a través del link del repositorio.](#)