

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?
- ¿Cómo crear un repositorio en GitHub?
- ¿Cómo crear una rama en Git?
- ¿Cómo cambiar a una rama en Git?
- ¿Cómo fusionar ramas en Git?
- ¿Cómo crear un commit en Git?
- ¿Cómo enviar un commit a GitHub?
- ¿Qué es un repositorio remoto?
- ¿Cómo agregar un repositorio remoto a Git?
- ¿Cómo empujar cambios a un repositorio remoto?
- ¿Cómo tirar de cambios de un repositorio remoto?
- ¿Qué es un fork de repositorio?
- ¿Cómo crear un fork de un repositorio?

- ¿Cómo enviar una solicitud de extracción (pull request) a un repositorio?
- ¿Cómo aceptar una solicitud de extracción?
- ¿Qué es un etiqueta en Git?
- ¿Cómo crear una etiqueta en Git?
- ¿Cómo enviar una etiqueta a GitHub?
- ¿Qué es un historial de Git?
- ¿Cómo ver el historial de Git?
- ¿Cómo buscar en el historial de Git?
 - ¿Cómo borrar el historial de Git?
- ¿Qué es un repositorio privado en GitHub?
- ¿Cómo crear un repositorio privado en GitHub?
- ¿Cómo invitar a alguien a un repositorio privado en GitHub?
- ¿Qué es un repositorio público en GitHub?
- ¿Cómo crear un repositorio público en GitHub?
- ¿Cómo compartir un repositorio público en GitHub?

Nombre: Ayrton Andrés caldo

✓ el principal objetivo de git es llevar a cabo un control de un proyecto que estemos realizando y git hub es una comunidad que nos permite compartir nuestro proyecto con los demás de forma pública o privada.

✓ Seguir a otros usuarios agregar otros repositorios en favoritos es como una red social.

✓ Lo primero es crear un carpeta donde queremos que se guarden los archivos luego abrir git ingresar (git init) se inicializa el repositorio local .

✓ Colocar (git add .) agregara todos los archivos necesarios

✓ (git commit -m " descripción de lo que se realizo en ese momento")

✓ luego iniciar en github en el perfil se creara el repositorio se pedirá el nombre se elige entre privado / publico.

✓ Crear el repositorio remoto y el código que nos manda pegarlo en

git .

✓ Para crear una rama se ejecutara el comando (git branch" más el nombre de la rama "

✓ Para cambiar de rama se colocara el comando git checkout "y el nombre de la rama"

✓ Posicionarse en la rama principal y realizar un git checkout mainLuego git merge "el nombre de la rama" y se fusionaran las ramas.

✓ Para realizar un commit primero se escribe git add . luego git commit -m " y los cambios que se realizaron"

✓ Se realiza escribe git add . luego git commit -m " y los cambios realizados" luego git push origin master.

✓ Seria la copia de nuestro código o proyecto que se encuentra subida en git hub como un respaldo.

✓ Para crear un repositorio remoto en git hub en la pantalla de inicio en el (+) crear repositorio no pedirá en nombre, descripción y si lo queremos de manera pública o privada.

✓ Se hace mediante el comando git push en git.

✓ Primero colocar el comando git log --oneline nos dará el historial de los commit realizados.

o Luego git reset --soft head~"y el número de commit a deshacer"

✓ El fork genera una copia del archivo original primero

✓ ir a github crear fork colocar un nombre y descripción y crear fork.

✓ seleccionar la rama que contiene los cambios, Comparar y solicitud de extracción poner una descripción, titulo y Crear solicitud de incorporación de cambios.

✓ Ir a nuestro perfil de github a la parte de solicitudes de extracción.

✓ Es para marcar en un punto específico de una rama o escribir algo.

✓ Se utiliza el comando git tag "nombre"

✓ Utilizando el comando git tag --tags

✓ El historial de git guarda todos los cambios que se fueron haciendo

- ✓ Usando el comando git log , git log “nombre del archivo”
- Para eliminar el historial de git se utiliza git reset –hard
- ✓ Un repositorio privado es donde solo tienen acceso el creado o un grupo de personas.
- ✓ Para crear un repositorio privado en github en el (+) crear nuevo repositorio y colocar la opción privado.
- ✓ Para invitar a alguien a un repositorio privado al crearlo nos da una URL que debemos compartir hacia esa persona.
- ✓ Es donde toda la gente que visite tu perfil puede verlo y descargar tu código y modificarlo.
- ✓ Para crear un repositorio privado en github en el (+) crear nuevo repositorio y colocar la opción publico.
- ✓ Para invitar a alguien a un repositorio público al crearlo nos da una URL que debemos compartir hacia esa persona.

2) Realizar la siguiente actividad:

LINK DEL REPOSITORIO:

https://github.com/Ayrtonandrescaldo/Repositorio_1.git

- 3) Realizar la siguiente actividad:

LINK DEL REPOSITORIO:

<https://github.com/Ayrtonandrescaldo/conflict-exercise.git>