

TECNICATURA UNIVERSITARIA

EN PROGRAMACIÓN

A DISTANCIA

Práctico 2: Git y GitHub

Pedraza, Joaquín Onésimo

Actividades

1)

1. **¿Qué es GitHub?** GitHub es una plataforma web para alojar y gestionar proyectos de software utilizando Git. Permite a los desarrolladores almacenar, colaborar y compartir código de manera eficiente. GitHub también facilita el control de versiones, la gestión de ramas y la colaboración mediante "pull requests" y "issues".
2. **¿Cómo crear un repositorio en GitHub?** Para crear un repositorio en GitHub, debes iniciar sesión en tu cuenta, ir al perfil y hacer clic en el botón "New" en la sección de repositorios. Luego, llenas los detalles del repositorio como nombre, descripción y el tipo de visibilidad (público o privado) y finalmente haces clic en "Create repository".
3. **¿Cómo crear una rama en Git?** Para crear una rama en Git, utilizas el comando `git branch <nombre-de-la-rama>`. Esto creará una nueva rama basada en la rama actual.
4. **¿Cómo cambiar a una rama en Git?** Para cambiar a una rama en Git, usas el comando `git checkout <nombre-de-la-rama>`. Esto moverá tu entorno de trabajo a la rama indicada.
5. **¿Cómo fusionar ramas en Git?** Para fusionar ramas en Git, primero te posicionas en la rama a la que quieres fusionar los cambios (generalmente la rama principal) usando `git checkout <rama-principal>`, luego ejecutas `git merge <nombre-de-la-rama>`.
6. **¿Cómo crear un commit en Git?** Para crear un commit en Git, primero debes agregar los archivos que has modificado con `git add <archivo>`. Luego, puedes registrar el commit con el comando `git commit -m "Mensaje descriptivo del commit"`.
7. **¿Cómo enviar un commit a GitHub?** Para enviar un commit a GitHub, utilizas el comando `git push origin <nombre-de-la-rama>`. Esto sube tus cambios locales al repositorio remoto en GitHub.

8. **¿Qué es un repositorio remoto?** Un repositorio remoto es una versión de tu repositorio que está alojada en un servidor o en una plataforma como GitHub, GitLab o Bitbucket, permitiendo la colaboración y el acceso desde cualquier lugar.
9. **¿Cómo agregar un repositorio remoto a Git?** Para agregar un repositorio remoto, utilizas el comando `git remote add origin <url-del-repositorio>`, donde `origin` es el nombre del repositorio remoto y `<url-del-repositorio>` es la dirección del repositorio en GitHub u otra plataforma.
10. **¿Cómo empujar cambios a un repositorio remoto?** Para empujar cambios, usas el comando `git push origin <nombre-de-la-rama>`, que envía tus commits locales al repositorio remoto.
11. **¿Cómo tirar de cambios de un repositorio remoto?** Para tirar de cambios de un repositorio remoto, utilizas el comando `git pull origin <nombre-de-la-rama>`, que trae las actualizaciones desde el repositorio remoto a tu entorno local.
12. **¿Qué es un fork de repositorio?** Un fork de repositorio es una copia de un repositorio que se crea en tu cuenta de GitHub para poder modificarlo de manera independiente del original sin afectar al proyecto principal.
13. **¿Cómo crear un fork de un repositorio?** Para crear un fork, navegas al repositorio que deseas copiar en GitHub y haces clic en el botón "Fork" en la parte superior derecha de la página del repositorio.
14. **¿Cómo enviar una solicitud de extracción (pull request) a un repositorio?** Una vez que hayas realizado cambios en tu fork, puedes ir a la página del repositorio original y hacer clic en "New pull request". Luego, seleccionas las ramas que deseas comparar y envías la solicitud de extracción.
15. **¿Cómo aceptar una solicitud de extracción?** Para aceptar una solicitud de extracción, un administrador o colaborador del repositorio puede revisar los cambios y, si están listos, hacer clic en "Merge pull request" para fusionar la solicitud con la rama principal.
16. **¿Qué es una etiqueta en Git?** Una etiqueta en Git es una referencia a un punto específico en el historial de commits, generalmente usada para marcar versiones importantes o lanzamientos del proyecto.
17. **¿Cómo crear una etiqueta en Git?** Para crear una etiqueta, utilizas el comando `git tag <nombre-etiqueta>`, que asigna una etiqueta al commit actual.

18. **¿Cómo enviar una etiqueta a GitHub?** Para enviar una etiqueta a GitHub, utilizas el comando `git push origin <nombre-etiqueta>`, lo cual sube la etiqueta al repositorio remoto.
19. **¿Qué es un historial de Git?** El historial de Git es una secuencia de commits que refleja todos los cambios realizados en un repositorio a lo largo del tiempo.
20. **¿Cómo ver el historial de Git?** Para ver el historial de Git, usas el comando `git log`, que muestra los commits realizados en el repositorio con detalles como el autor, la fecha y el mensaje del commit.
21. **¿Cómo buscar en el historial de Git?** Puedes buscar en el historial de Git usando el comando `git log --grep="palabra-clave"`, lo cual filtra los commits que contienen esa palabra clave en el mensaje.
22. **¿Cómo borrar el historial de Git?** Borrar el historial de Git no es recomendable, pero si es necesario, puedes usar el comando `git reset --hard <commit-id>` para deshacer todos los commits después de un commit específico.
23. **¿Qué es un repositorio privado en GitHub?** Un repositorio privado en GitHub es un repositorio que solo es accesible para los usuarios invitados o autorizados, lo que permite mantener el código en privado.
24. **¿Cómo crear un repositorio privado en GitHub?** Al crear un repositorio en GitHub, puedes seleccionar la opción "Private" en el campo de visibilidad para hacerlo privado.
25. **¿Cómo invitar a alguien a un repositorio privado en GitHub?** Para invitar a alguien a un repositorio privado, debes ir a la sección de "Settings" del repositorio, seleccionar "Manage access" y luego invitar a un colaborador por su nombre de usuario o correo electrónico.
26. **¿Qué es un repositorio público en GitHub?** Un repositorio público en GitHub es un repositorio accesible para cualquiera, lo que permite que cualquier usuario vea el código y contribuya al proyecto.
27. **¿Cómo crear un repositorio público en GitHub?** Al crear un repositorio en GitHub, seleccionas la opción "Public" en el campo de visibilidad para hacerlo público.
28. **¿Cómo compartir un repositorio público en GitHub?** Para compartir un repositorio público, simplemente copias la URL del repositorio y la compartes con quien desees. También puedes usar la opción de "Share" en la interfaz de GitHub.

2) Enlace al repositorio creado para realizar la actividad:

<https://github.com/JoaquinPedraza/Trabajo-Colaborativo-act2.git>

3) Enlace al repositorio creado para realizar la actividad:

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