

Práctico 2: Git y GitHub

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

- ¿Qué es GitHub?

Es una plataforma, un servidor, que utiliza Git para almacenar código en un repositorio. GitHub permite a desarrolladores colaborar entre sí, generando un clima de discusiones y aportes, así como también de descargas de programas y aplicaciones.

- ¿Cómo crear un repositorio en GitHub?

Podríamos crear un repositorio en GitHub accediendo a nuestro perfil en la opción “nuevo repositorio”.

- ¿Cómo crear una rama en Git?

Podríamos crear una rama en Git a través del comando ‘git branch’ incluyendo el nombre de la rama posterior a un espacio.

- ¿Cómo cambiar a una rama en Git?

Podríamos cambiar una rama en Git a través del comando ‘git checkout’ incluyendo el nombre de la rama posterior a un espacio.

- ¿Cómo fusionar ramas en Git?

Para fusionar ramas en Git deberíamos utilizar el comando ‘git merge’ incluyendo el nombre de la rama posterior a un espacio.

- ¿Cómo crear un commit en Git?

Para crear un commit deberíamos de ejecutar el comando ‘git commit -m “mensaje”’, claro está que antes deberíamos de haber agregado el archivo al Stage.

- ¿Cómo enviar un commit a GitHub?

Luego de agregar de forma provisoria un archivo en el repositorio local con el comando ‘git add “nombre del archivo”’ y de confirmar a través de consola con el comando ‘git commit -m “nombre del archivo”’. Por último, a través del comando ‘git push -u origin master’ pasaríamos el archivo a nuestro repositorio remoto.

- ¿Qué es un repositorio remoto?

Es una copia de un proyecto que se encuentra alojado es un servidor, como ser el de GitHub.

- ¿Cómo agregar un repositorio remoto a Git?

Podríamos agregar un repositorio remoto a Git a través del comando 'git remote add origin "url"'

- ¿Cómo empujar cambios a un repositorio remoto?

A través del comando 'git push -u origin master', o bien 'git push' luego de la primera vez.

- ¿Cómo tirar de cambios de un repositorio remoto?

Podríamos tomar esos cambios utilizando primero el comando 'git fetch' para obtener esos cambios del repositorio remoto, y luego a través del comando 'git pull origin master' podríamos aplicar los cambios.

- ¿Qué es un fork de repositorio?

Es una copia exacta de un repositorio original donde podemos escribir, haciendo públicos nuestros propios cambios, generando una participación abierta en post de la programación colaborativa.

- ¿Cómo crear un fork de un repositorio?

Partiendo de un repositorio publico podríamos copiar dicho repositorio en nuestra propia cuenta a través de la solapa Fork en GitHub, teniendo la posibilidad de copiar la rama principal y así como también renombrar el repositorio.

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

Luego de realizar un Fork del proyecto respectivo, así como también implementar los cambios que querríamos llevar a cabo, y crear la rama respectiva, podríamos realizar el pull request del repositorio.

- ¿Cómo aceptar una solicitud de extracción?

En el nombre del repositorio, haciendo clic en "solicitudes de incorporación de cambios", y luego, en haciendo clic en la que querríamos revisar, siguiendo en "archivos cambiados", revisando claro esta los cambios en la solicitud de cambios, con posterioridad haciendo clic en "revisar los cambios", podríamos luego aprobar la combinación de los cambios propuestos en la solicitud de incorporación de cambios, haciendo clic luego en "enviar revisión".

- ¿Qué es un etiqueta en Git?

En Git una etiqueta es un proceso mediante el cual podemos marcar o crear puntos específicos en el historial de nuestro proyecto o repositorio y esto generalmente se hace para marcar algunos puntos de lanzamientos limpios.

- ¿Cómo crear una etiqueta en Git?

Para crear una etiqueta tenemos que hacer uso del comando 'git tag' pudiendo identificar con el número de versión.

- ¿Cómo enviar una etiqueta a GitHub?

Podríamos enviar una etiqueta a GitHub a través del comando "git push origin tag <tag_name>".

- ¿Qué es un historial de Git?

Es el registro de las confirmaciones realizadas en un repositorio de código, permite ver quiénes son los autores de los cambios, averiguar donde se introdujeron los errores, revertiendo los cambios problemáticos.

El comando para ver el historial de Git es "git log"

- ¿Cómo ver el historial de Git?

Podríamos ver el historial de Git a través del comando "git log".

- ¿Cómo buscar en el historial de Git?

Podríamos buscar en el historial de Git a través de los comandos "git log -follow <nombre archivo>".

- ¿Cómo borrar el historial de Git?

Localmente, deberíamos de borrar la carpeta oculta .git

- ¿Qué es un repositorio privado en GitHub?

Es un espacio en el servidor donde se puede guardar código, archivos y revisiones de forma privada, solo el propio usuario podría visualizarlo, o quienes nosotros permitamos.

- ¿Cómo crear un repositorio privado en GitHub?

Podemos cambiar la visibilidad del repositorio desde el menú de configuración en GitHub, haciendo clic en cambiar la visibilidad.

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

Podríamos invitar a alguien agregando su nombre en el repositorio, de esa manera le llegaría un correo electrónico para que acepte la invitación. De esta manera tendrá acceso de colaborador en nuestro repositorio.

- ¿Qué es un repositorio público en GitHub?

Es un repositorio en el que se puede almacenar, compartir y trabajar en código de forma colaborativa.

- ¿Cómo crear un repositorio público en GitHub?

Así como podemos cambiar la visibilidad del repositorio de forma privada, lo podemos poner de forma pública eligiendo dicha visibilidad desde GitHub.

- ¿Cómo compartir un repositorio público en GitHub?

Podemos compartir nuestro repositorio escribiendo el nombre de la persona o equipo a invitar desde GitHub o bien, compartiendo la URL del repositorio.

LINK repositorio: <https://github.com/pablonahuel81/primer-repo-git>

