

---

## Práctico 2: Git y GitHub

### Actividades:

---

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

- **¿Qué es GitHub?**

Github es una plataforma en línea que permite a los desarrolladores almacenar, colaborar y gestionar proyectos de software. Es útil para trabajos en equipo.

- **¿Cómo crear un repositorio en GitHub?**

Se debe ingresar a la plataforma de Github con tu usuario y ingresar a la parte de repositorios, allí seleccionar la opción de nuevo repositorio.

- **¿Cómo crear una rama en Git?**

Para crear una rama en Git debemos usar el comando `GIT BRANCH 'nombre de la rama'`

- **¿Cómo cambiar a una rama en Git?**

Se cambia de rama utilizando el comando `GIT CHECKOUT 'rama'`

- **¿Cómo fusionar ramas en Git?**

Se utiliza el `GIT MERGE` para fusionar ramas

- **¿Cómo crear un commit en Git?**

Para crear un commit debemos utilizar `GIT COMMIT -M`

- **¿Cómo enviar un commit a GitHub?**

Utilizamos el comando `PUSH`

- **¿Qué es un repositorio remoto?**

Un repositorio remoto es una versión de tu repositorio local que está alojada en internet o en una red

- **¿Cómo agregar un repositorio remoto a Git? • ¿Cómo empujar cambios a un repositorio remoto?**

Primero debemos conectar el repositorio remoto con el local con el comando `GIT REMOTE ADD ORIGIN` junto con el url del repositorio. Para empujar los cambios debemos utilizar `GIT PUSH ORIGIN` junto con el nombre de la rama.

- **¿Cómo tirar de cambios de un repositorio remoto?**

Debemos utilizar el comando `PULL` es decir `git pull origin (nombre de la rama)`. Esto descargará los cambios más recientes del repositorio remoto y los fusionará con la rama local.

- **¿Qué es un fork de repositorio?**

Un fork en un repositorio es una copia independiente de un repositorio existente

- **¿Cómo crear un fork de un repositorio?**

Primero debemos ir al repositorio que queremos realizar el fork en github. Allí tocar el botón de fork.

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

Primero debemos hacer un fork del repositorio original, luego clonarlo para crear una nueva rama donde realizaremos nuestros cambios en el archivo. Debemos subir nuestra rama al repositorio remoto así poder iniciar la solicitud de extracción en GitHub: Tocamos el botón de COMPARE PULL REQUEST, hacemos una descripción de los cambios y hacemos clic en CREATE PULL REQUEST.

- **¿Cómo aceptar una solicitud de extracción?**

Debemos seleccionar la solicitud que queremos aceptar y clickear en el botón MERGE PULL REQUEST y luego confirmamos la acción con CONFIRM MERGE. Los cambios de la solicitud de extracción se integrarán en la rama base.

- **¿Qué es un etiqueta en Git?**

Una etiqueta en Git (tag) es una forma de marcar un punto específico en el historial del proyecto. Las etiquetas facilitan la gestión de versiones en proyectos grandes.

- **¿Cómo crear una etiqueta en Git?**

Se crea con el comando `GIT TAG 'nombre'`

También con el comando `GIT TAG -A 'nombre' -M 'mensaje'` creamos una etiqueta con un mensaje

- **¿Cómo enviar una etiqueta a GitHub?**

Con el comando `GIT PUSH --TAGS` subimos los cambios incluyendo todos los tags

- **¿Qué es un historial de Git?**

El historial de Git es el registro completo de todos los cambios realizados en un repositorio. Este historial documenta cada commit realizado por los colaboradores, lo que permite rastrear qué se cambió, cuándo y quién lo hizo. Es fundamental para el desarrollo colaborativo, ya que facilita la resolución de problemas y la revisión del progreso del proyecto.

- **¿Cómo ver el historial de Git?**

Hay diferentes comandos para ver el historial en git:

El básico es `GIT LOG` para obtener una vista completa del historial luego `GIT LOG --ONELINE` mostrara una línea por commit.

`GIT LOG 'nombre del archivo'` para ver el historial de un archivo en particular.

Se pueden filtrar por fecha y por autor con `GIT LOG --AUTHOR= 'nombre'` y `GIT LOG --SINCE= 'fecha'`

Con `GIT LOG --GRAPH --ONELINE --ALL` se puede ver el historial en forma de árbol que es útil en repositorios con ramas y colaboraciones.

- **¿Cómo buscar en el historial de Git?**

Para cambios específicos y ver las diferencias de cada commit utilizamos GIT LOG -P

- **¿Cómo borrar el historial de Git?**

Para borrar el historial de Git localmente, puedes borrar la carpeta .git. También puedes reescribir la historia del repositorio para eliminar un archivo o datos confidenciales.

- **¿Qué es un repositorio privado en GitHub?**

Es un espacio en el cual se puede almacenar un proyector y controlar quien tiene acceso.

- **¿Cómo crear un repositorio privado en GitHub?**

Es igual a crear cualquier repositorio pero debemos seleccionar la opción de PRIVATE

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

Con el botón INVITE A COLLABORATOR. Debemos escribir el usuario o el correo de la persona que deseamos invitar. Se puede elegir el nivel de acceso que se desea otorgar. Hacemos click en ADD y la persona recibirá una notificación en github o correo para aceptar la invitación.

- **¿Qué es un repositorio público en GitHub?**

Un repositorio publico esta accesible para cualquier persona en internet y son ideales [ara proyectos de código abierto, colaboración global y compartir conocimiento.

- **¿Cómo crear un repositorio público en GitHub?**

Se crea como cualquier repositorio y elegimos la opción PUBLIC

- **¿Cómo compartir un repositorio público en GitHub?**

Seleccionamos el botón verde que dice CODE y ahí seleccionamos el url del repositorio para compartir.