

Práctico 2: Git y GitHub

Actividades

1) Contestar las siguientes preguntas utilizando las guías y documentación proporcionada

- ¿Qué es GitHub?

GitHub es una plataforma de alojamiento de código fuente basada en la web que utiliza el sistema de control de versiones Git. Permite a los desarrolladores colaborar en proyectos, gestionar repositorios, controlar versiones del código, y realizar tareas de revisión de código. GitHub es ampliamente utilizado por programadores de todo el mundo para gestionar proyectos tanto públicos como privados, compartir código y trabajar en equipo.

- ¿Cómo crear un repositorio en GitHub?

Para crear un repositorio en GitHub:

1. Inicia sesión en tu cuenta de GitHub.
2. Haz clic en el botón **"New"** (Nuevo) en la parte superior derecha de la página de tu perfil.
3. Rellena los campos solicitados:
 - **Repository name** (Nombre del repositorio)
 - **Description** (Opcional, descripción del repositorio)
 - Elige si el repositorio será **público** o **privado**.
4. Haz clic en **"Create repository"**.

- ¿Cómo crear una rama en Git?

Para crear una nueva rama en Git:

`git branch <nombre_de_la_rama>`

Esto creará una nueva rama en tu repositorio local.

- ¿Cómo cambiar a una rama en Git?

Si deseas cambiarte a esa rama inmediatamente, utiliza el siguiente comando:

`git checkout <nombre_de_la_rama>`

- ¿Cómo fusionar ramas en Git?

Para fusionar una rama en la rama actual, usa el siguiente comando:

`git merge <nombre_de_la_rama>`

Asegúrate de estar en la rama a la que quieres fusionar antes de ejecutar este comando.

- ¿Cómo crear un commit en Git?

Para crear un commit, sigue estos pasos:

1. Añade los cambios al área de preparación (staging area):
 - `git add <archivo>` # Para archivos específicos
 - `git add .` # Para añadir todos los cambios
2. Realiza el commit:
`git commit -m "Mensaje descriptivo del commit"`

- ¿Cómo enviar un commit a GitHub?

Para enviar (empujar) un commit a GitHub:

1. Asegúrate de tener un repositorio remoto configurado.
2. Usa el comando git push:

`git push origin <nombre_de_la_rama>`

Este comando envía los cambios locales a GitHub.

- ¿Qué es un repositorio remoto?

Un **repositorio remoto** es una versión de tu repositorio que se encuentra en un servidor externo, como GitHub. Se utiliza para almacenar y compartir el código con otros colaboradores. Los repositorios remotos permiten trabajar de forma colaborativa, manteniendo el código sincronizado entre múltiples usuarios.

- ¿Cómo agregar un repositorio remoto a Git?

Para agregar un repositorio remoto, utiliza el siguiente comando:

```
git remote add origin <url_del_repositorio>
```

Esto vincula tu repositorio local con el repositorio remoto.

- ¿Cómo empujar cambios a un repositorio remoto?

Para empujar los cambios locales a un repositorio remoto, usa el siguiente comando:

```
git push origin <nombre_de_la_rama>
```

Este comando envía tus cambios a la rama correspondiente en el repositorio remoto.

- ¿Cómo tirar de cambios de un repositorio remoto?

Para descargar los cambios del repositorio remoto, usa:

```
git pull origin <nombre_de_la_rama>
```

Este comando trae los cambios desde el repositorio remoto y los fusiona con tu rama local.

- ¿Qué es un fork de repositorio?

Un **fork** es una copia de un repositorio en tu propia cuenta de GitHub. Te permite hacer cambios o mejoras sin afectar el repositorio original. Es comúnmente utilizado para contribuir a proyectos de código abierto.

- ¿Cómo crear un fork de un repositorio?

Para crear un fork:

1. Visita la página del repositorio en GitHub.
2. Haz clic en el botón **"Fork"** en la parte superior derecha de la página del repositorio.
3. GitHub creará una copia del repositorio en tu cuenta.

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

Para enviar un pull request (PR):

1. Realiza cambios en tu fork y súbelos a tu repositorio en GitHub.
2. Ve al repositorio original y haz clic en **"Pull requests"**.
3. Haz clic en **"New pull request"**.
4. Selecciona tu rama y compara tus cambios con la rama principal del repositorio original.
5. Haz clic en **"Create pull request"** y agrega un mensaje describiendo los cambios.

- ¿Cómo aceptar una solicitud de extracción?

Para aceptar un pull request, un mantenedor del repositorio original debe:

1. Revisar el PR.
2. Si todo está bien, hacer clic en **"Merge pull request"** para fusionar los cambios.
3. Confirma la fusión.

- ¿Qué es una etiqueta en Git?

Una **etiqueta** (tag) en Git es una referencia específica a un punto en el historial del repositorio. Las etiquetas se usan principalmente para marcar versiones importantes o puntos de lanzamiento.

- ¿Cómo crear una etiqueta en Git?

Para crear una etiqueta:

```
git tag <nombre_de_la_etiqueta>
```

Si deseas etiquetar una versión específica, utiliza:

```
git tag <nombre_de_la_etiqueta> <hash_del_commit>
```

- ¿Cómo enviar una etiqueta a GitHub?

Para enviar una etiqueta a GitHub, usa:

```
git push origin <nombre_de_la_etiqueta>
```

- ¿Qué es un historial de Git?

El **historial de Git** es una secuencia de commits que muestra todos los cambios realizados en el repositorio a lo largo del tiempo. Puedes ver qué cambios se han realizado y cuándo.

- ¿Cómo ver el historial de Git?

Para ver el historial de commits, utiliza:

`git log`

Este comando muestra una lista de commits con información sobre el autor, fecha y mensaje de cada commit.

- ¿Cómo buscar en el historial de Git?

Para buscar un commit específico en el historial, puedes usar:

`git log --grep="palabra_clave"`

- ¿Cómo borrar el historial de Git?

Para borrar el historial de Git, puedes usar el siguiente comando:

`git reset --hard <commit_id>`

Este comando restablece tu repositorio al estado de un commit anterior, eliminando los commits posteriores.

- ¿Qué es un repositorio privado en GitHub?

Un **repositorio privado** es un repositorio cuyo acceso está restringido solo a los colaboradores que se inviten explícitamente. Solo las personas con permisos pueden ver o contribuir al contenido del repositorio.

- ¿Cómo crear un repositorio privado en GitHub?

Para crear un repositorio privado:

1. Haz clic en **"New"** en GitHub.
2. En el formulario de creación del repositorio, selecciona **"Private"**.
3. Haz clic en **"Create repository"**.

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

Para invitar a alguien:

1. Ve al repositorio privado en GitHub.
2. Haz clic en **"Settings"**.
3. En la sección de **"Manage access"**, haz clic en **"Invite a collaborator"**.
4. Ingresa el nombre de usuario de la persona y haz clic en **"Add"**.

- ¿Qué es un repositorio público en GitHub?

Un **repositorio público** es un repositorio cuyo contenido está disponible para cualquier persona en GitHub. Cualquier persona puede ver, descargar y contribuir al repositorio.

- ¿Cómo crear un repositorio público en GitHub?

Para crear un repositorio público:

1. Haz clic en **"New"** en GitHub.
2. En el formulario de creación, selecciona **"Public"**.
3. Haz clic en **"Create repository"**.

- ¿Cómo compartir un repositorio público en GitHub?

Para compartir un repositorio público:

1. Copia la URL del repositorio desde la barra de direcciones del navegador.
2. Comparte la URL con otros para que puedan ver o clonar el repositorio.

2)

<https://github.com/Titacele/TP2-Git-y-GitHub>

3)

<https://github.com/Titacele/TP2-conflict-exercise>