

Práctico 2: Git y GitHub

¿Qué es GitHub?

GitHub es una plataforma para programadores donde puedes almacenar y colaborar en proyectos de código. Permite gestionar versiones, trabajar en equipo y compartir proyectos, especialmente en desarrollo de software.

¿Cómo crear un repositorio en GitHub?

1. Inicia sesión en tu cuenta.
2. Haz clic en "New" (Nuevo).
3. Completa el nombre, descripción, y ajusta las configuraciones.
4. Haz clic en "Create repository" (Crear repositorio).

¿Cómo crear una rama en Git?

`git branch nombre-de-la-rama`

¿Cómo cambiar a una rama en Git?

`git checkout nombre-de-la-rama`

¿Cómo fusionar ramas en Git?

`git merge nombre-de-la-rama`

¿Cómo crear un commit en Git?

`git commit -m "Mensaje descriptivo sobre los cambios realizados"`

¿Cómo enviar un commit a GitHub?

1. Conecta tu repositorio local con el remoto (si aún no lo has hecho):

`git remote add origin URL-del-repositorio`

Reemplaza URL-del-repositorio con el enlace del repositorio en GitHub.

2. Envía los commits al repositorio remoto:

`git push origin nombre-de-la-rama`

¿Qué es un repositorio remoto?

Es donde guardas tu código en línea, como en GitHub, para compartir y colaborar. Conectas tu repositorio local al remoto para sincronizar cambios usando `git push` y `git pull`.

¿Cómo agregar un repositorio remoto a Git?

`git remote add origin URL-del-repositorio`

¿Cómo empujar cambios a un repositorio remoto?

`git push origin nombre-de-la-rama`

¿Cómo tirar de cambios de un repositorio remoto?

`git pull origin nombre-de-la-rama`

¿Qué es un fork de repositorio?

Un fork es una copia de un repositorio en tu cuenta de GitHub para trabajar en él sin afectar el original. Ideal para colaborar en proyectos de código abierto.

¿Cómo crear un fork de un repositorio?

1. Encuentra el repositorio en GitHub que deseas copiar.
2. Haz clic en el botón **"Fork"** en la esquina superior derecha de la página.
3. Selecciona tu cuenta o la organización donde deseas crear el fork.

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

1. Haz un fork y clona el repositorio: Trabaja en tu copia del proyecto y realiza cambios necesarios.
2. Confirma los cambios:
Usa `git add`, `git commit`, y `git push` para enviar los cambios a tu repositorio remoto (tu fork).
3. Inicia la solicitud: Ve al repositorio original en GitHub, haz clic en "Pull Request" y selecciona los cambios que deseas fusionar desde tu fork.
4. Añade detalles: Escribe un mensaje claro explicando tus cambios.
5. Envía la solicitud: Haz clic en "Create Pull Request".

¿Cómo aceptar una solicitud de extracción?

1. Controlar que los cambios funcionan correctamente y no introducen problemas.
2. Aprueba la solicitud: Haz clic en "Approve" (Aprobar).
3. Fusiona los cambios: Haz clic en el botón "Merge Pull Request" y selecciona el método de fusión que prefieras (por ejemplo, "Merge commit" o "Squash and merge").

¿Qué es una etiqueta en Git?

Una etiqueta en Git es un marcador fijo usado para identificar puntos importantes en el historial, como lanzamientos de versiones. No cambian ni se mueven como las ramas.

¿Cómo crear una etiqueta en Git?

1. Elige el commit que deseas etiquetar:
2. Usa el comando para crear la etiqueta:
3. `git tag nombre-de-la-etiqueta`
4. Si deseas incluir un mensaje:
5. `git tag -a nombre-de-la-etiqueta -m "Descripción de la etiqueta"`
6. Empuja la etiqueta al repositorio remoto (opcional):
7. `git push origin nombre-de-la-etiqueta`

¿Cómo enviar una etiqueta a GitHub?

1. Crea la etiqueta localmente:
2. `git tag nombre-de-la-etiqueta`
3. Envía la etiqueta al repositorio remoto:
4. `git push origin nombre-de-la-etiqueta`

¿Qué es un historial de Git?

El historial de Git es el registro de todos los cambios y commits realizados en tu proyecto. Permite rastrear modificaciones y revertir versiones anteriores.

¿Cómo ver el historial de Git?

`git log`

Esto mostrará los commits con detalles como autor, fecha y mensaje. Para un historial más compacto:

`git log --oneline`

¿Cómo buscar en el historial de Git?

1. Buscar por palabras clave en mensajes de commit:

2. `git log --grep="palabra-clave"`

3. Filtrar por autor:

4. `git log --author="nombre-del-autor"`

5. Ver cambios en un archivo específico:

6. `git log -- archivo`

¿Cómo borrar el historial de Git?

Para borrar o reiniciar el historial de Git: elimina la carpeta `.git` o usa `git checkout --orphan nueva-rama` para empezar desde cero con un nuevo historial. Estas acciones son drásticas, así que úsalas con cuidado.

¿Qué es un repositorio privado en GitHub?

Un repositorio privado en GitHub es un espacio donde almacenas tu código de forma oculta al público. Solo las personas a quienes invites pueden acceder y colaborar en él.

¿Cómo crear un repositorio privado en GitHub?

1. Ve a tu cuenta de GitHub y haz clic en el botón "New" para crear un repositorio.
2. Completa el formulario con el nombre y descripción del repositorio.
3. Selecciona la opción "Private" para que solo personas invitadas puedan acceder.
4. Haz clic en "Create repository".

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

Ve a Settings > Collaborators, agrega el usuario y envía la invitación.

¿Qué es un repositorio público en GitHub?

Un repositorio público en GitHub es un proyecto visible para cualquier persona en la plataforma. Esto permite que otros puedan ver el código, descargarlo y, si decides permitirlo, colaborar mediante issues o pull requests.

¿Cómo crear un repositorio público en GitHub?

1. Ve a tu cuenta y haz clic en "New" para iniciar la creación de un nuevo repositorio.
2. Llena los campos como el nombre y la descripción del proyecto.
3. Selecciona la opción "Public" para hacerlo visible al público.
4. Haz clic en "Create repository"

¿Cómo compartir un repositorio público en GitHub?

Para compartir un repositorio público en GitHub, simplemente copia la URL del repositorio desde la barra de direcciones de tu navegador o desde el botón "Code" en la página del repositorio. Luego, envíala a la persona o publícala donde quieras que otros accedan a tu proyecto.

Manuel Andres Vizone