

Práctico 2: Gity GitHub

Objetivo:

El estudiante desarrollará competencias para trabajar con Git y GitHub, aplicando conceptos fundamentales de control de versiones, colaboración en proyectos y resolución de conflictos, en un entorno simulado y guiado.

Resultados de aprendizaje:

1. Comprender los conceptos básicos de Git y GitHub: Identificar y explicar los principales términos y procesos asociados con Git y GitHub, como repositorios, ramas, commits, forks, etiquetas y repositorios remotos.
2. Manejar comandos esenciales de Git: Ejecutar comandos básicos para crear, modificar, fusionar y gestionar ramas, commits y repositorios, tanto en local como en remoto.
3. Aplicar técnicas de colaboración en GitHub: Configurar y utilizar repositorios remotos, realizar forks, y gestionar pull requests para facilitar el trabajo colaborativo.
4. Resolver conflictos en un entorno de control de versiones: Identificar, analizar y solucionar conflictos de merge generados en un flujo de trabajo con múltiples ramas.

Actividades

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

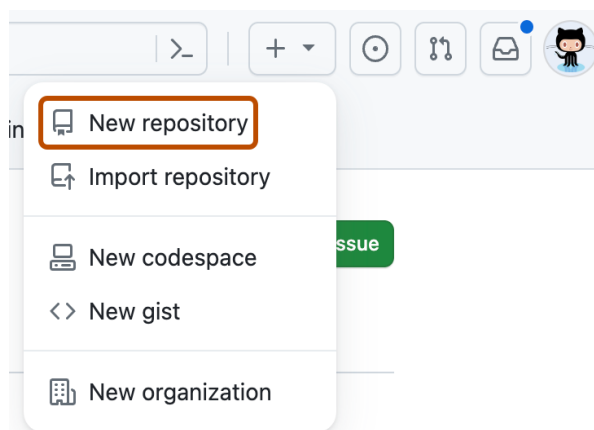
¿Qué es GitHub?

Es un repositorio remoto, comunidad donde se puede compartir los repositorios en forma pública o privada. Plataforma de desarrollo colaborativo basada en la web que permite a los desarrolladores almacenar y gestionar su código fuente.

¿Cómo crear un repositorio en GitHub?



Los repositorios de GitHub almacenan una variedad de proyectos. En esta guía, creará un repositorio y confirmará el primer cambio.

1. En la esquina superior derecha de cualquier página, selecciona y luego haz clic en **Nuevo repositorio**.



2. Escriba un nombre corto y fácil de recordar para el repositorio. Por ejemplo: "hola-mundo".

Owner * Repository name *

 octocat / hello-world 

Great repository names are short and memorable. Need inspiration? How about [vigilant-meme?](#)

Description (optional)

3. Opcionalmente, puede agregar una descripción del repositorio. Por ejemplo, "Mi primer repositorio en GitHub".
4. Elige la visibilidad del repositorio.
5. Seleccione **Initialize this repository with a README** (Inicializar este repositorio con un archivo Léame).
6. Haga clic en **Create repository** (Crear repositorio).

¿Cómo crear una rama en Git?

Utilizando el comando: `git branch <branch name>` para crea una nueva rama

¿Cómo cambiar a una rama en Git?

El comando `git checkout` cambia entre ramas en un proyecto.

¿Cómo fusionar ramas en Git?

La herramienta `git merge` se utiliza para fusionar uno o más ramas dentro de la rama que tienes activa.

¿Cómo crear un commit en Git?

Realizar el commit con el comando `git commit -m «Mensaje descriptivo»`. Es importante que el mensaje sea descriptivo y explique los cambios realizados en este commit.

¿Cómo enviar un commit a GitHub?

Con el comando: `git push origin nombre_de_la_rama`

¿Qué es un repositorio remoto?

Un repositorio remoto es una copia de tu proyecto alojada en un servidor.

¿Cómo agregar un repositorio remoto a Git?

Para agregar un repositorio remoto, use el comando `Git Remote Add`, seguido del nombre y URL del repositorio remoto. Por ejemplo:

```
git remote agregar origen https://github.com/usuario/repositorio.git
```

Reemplazar '**origen**' con el nombre que desea asignar al repositorio remoto y reemplazar '**https://github.com/usuario/repositorio.git**' con la URL del repositorio remoto real. El nombre '**origen**' es una convención para el repositorio remoto predeterminado, pero puede elegir cualquier nombre que se adapte a sus necesidades.

¿Cómo empujar cambios a un repositorio remoto?

Utilizando el comando `git pull`

¿Cómo tirar de cambios de un repositorio remoto?

Utilizando el comando: `git commit -m "nombre"`

¿Qué es un fork de repositorio?

La función de fork en Git se define como una operación usual en el sistema de Git que **se encarga de la creación de una copia de un repositorio** en la cuenta de usuario.

¿Cómo crear un fork de un repositorio?

Pasos para hacer un fork en GitHub:

1. **Seleccionar el repositorio:** Inicia sesión en GitHub y navega hasta el repositorio que deseas bifurcar.
2. **Hacer clic en "Fork":** En la parte superior derecha de la página del repositorio, verás un botón que dice "Fork". Al hacer clic en este botón, GitHub creará una copia del repositorio en tu cuenta.
3. **Clonar el fork:** Una vez que has bifurcado el repositorio, puedes clonarlo en tu máquina local usando el comando `git clone` y la URL de tu fork. Esto te permitirá trabajar en el código de manera local.
4. **Hacer cambios y comitear:** Ahora puedes realizar los cambios que desees en el código. Una vez que hayas terminado, puedes hacer commits de tus modificaciones.
5. **Push y pull request:** Finalmente, sube tus cambios a tu repositorio en GitHub usando el comando `git push`. Si deseas contribuir al proyecto original, puedes crear un *pull request*, lo que permitirá a los administradores del repositorio original revisar y, potencialmente, aceptar tus cambios en su código.

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

En primer lugar, necesitas tener una cuenta de GitHub y un repositorio donde realizarás cambios. Si estás contribuyendo a otro proyecto, necesitarías bifurcar ese repositorio a tu cuenta.

A continuación, necesitas crear una nueva rama en tu repositorio. Esta rama contendrá los cambios que quieres proponer. Para crear una nueva rama, utiliza el siguiente comando en tu terminal:

```
git checkout -b mi-nueva-rama
```

Una vez que la nueva rama está lista, realiza tus cambios. Estos cambios podrían ser cualquier. Después de realizar cambios, necesitas prepararlos y confirmarlos. La preparación es el proceso de preparar cambios para una confirmación. Para preparar tus cambios, utiliza el comando:

5. `git add .`

Después de la preparación, confirmas los cambios con un mensaje que describe lo que has hecho. El mensaje de confirmación debe ser conciso e informativo. Para confirmar tus cambios, utiliza el comando:

6. `git commit -m "Tu mensaje de confirmación informativo"`

Ahora, necesitas empujar tus cambios al repositorio de GitHub. Utiliza el comando:

7. `git push origin mi-nueva-rama`

Después de empujar los cambios, ve al sitio web de GitHub. Navega a tu repositorio y verás un botón 'Comparar y solicitar extracción'. Haz clic en él.

En la siguiente página, escribe un título y descripción para tu solicitud de extracción y haz clic en 'Crear solicitud de extracción'.

¿Cómo aceptar una solicitud de extracción?

Después de que tu solicitud de extracción haya sido revisada y aprobada, puede fusionarse en el repositorio haciendo clic en "Fusionar solicitud de extracción".

¿Qué es una etiqueta en Git?

Git Tag es una poderosa característica de Git que permite a los usuarios capturar un punto en la historia y marcarlo con una referencia que puede ser utilizada para comparaciones futuras o restauraciones.

¿Cómo crear una etiqueta en Git?

Para crear etiquetas en Git, vamos a usar el siguiente comando:

```
git tag -a 0.5 -m 'Versión 0.5'
```

¿Cómo enviar una etiqueta a GitHub?

Enviar una etiqueta al repositorio remoto es un proceso sencillo pero que debemos hacer explícitamente para que otros desarrolladores tengan acceso a la etiqueta. Usamos el comando "git push" con la opción "--tags" para enviar todas las etiquetas:

```
$ git push --tags
```

También podemos enviar una etiqueta específica agregando el nombre de la etiqueta al comando anterior:

```
$ git push origin v1.0
```

¿Qué es un historial de Git?

Son los cambios que sufre el proyecto

¿Cómo ver el historial de Git?

Utilizando el comando git log

¿Cómo buscar en el historial de Git?

Se utiliza el siguiente comando: `registro de git--grep="archivo"--una línea`

¿Cómo borrar el historial de Git?

Se utiliza el siguiente comando:

Paso 1: Acceder al repositorio

Para borrar un repositorio en Git, lo primero que debemos hacer es acceder a él desde nuestra línea de comandos. Utilizamos el siguiente comando:

```
“`cd nombre_del_repositorio”
```

Recuerda asegurarte de que te encuentras en la ubicación correcta antes de ejecutar este comando.

Paso 2: Borrar el repositorio local

Una vez que estemos dentro del repositorio, podemos borrarlo utilizando el siguiente comando:

```
“`rm -rf .git”
```

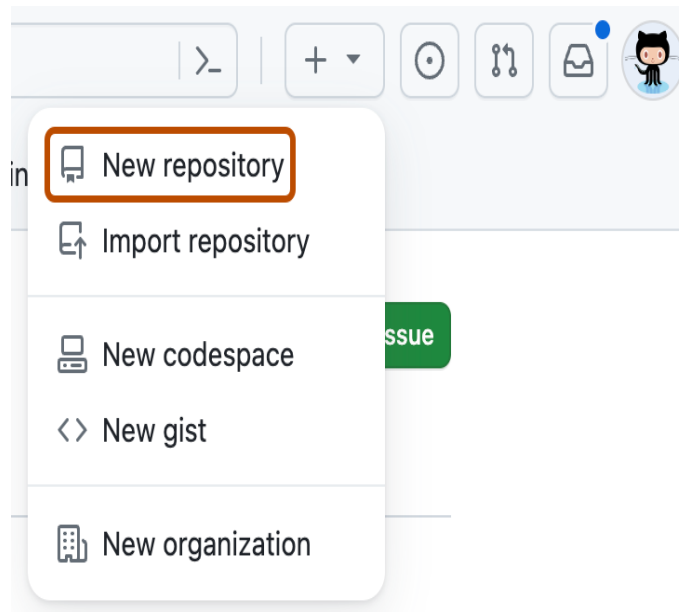
Este comando borrará de manera definitiva la carpeta .git, que contiene toda la información del repositorio

¿Qué es un repositorio privado en GitHub?

Es un repositorio que no se puede visualizar

¿Cómo crear un repositorio privado en GitHub?


En la esquina superior derecha de cualquier página, selecciona y luego haz clic en **Nuevo repositorio**.



1. Otra opción para crear un repositorio con la estructura de directorios y los archivos de un repositorio existente es seleccionar el menú desplegable **Escoger una plantilla** y hacer clic un repositorio de plantilla. Verás repositorios de plantillas que te pertenecen a ti y a las organizaciones de las que eres miembro o bien repositorios de plantillas que has usado anteriormente. Para más información, consulta [Crear un repositorio desde una plantilla](#).
2. Si decidió elegir una plantilla, también puede incluir la estructura de directorios y los archivos de todas las ramas en la plantilla, y no únicamente aquellos de la rama predeterminada, seleccione **Include all branches**.
3. Usa el menú desplegable **Propietario** para seleccionar la cuenta que quieres que sea propietaria del repositorio.

Owner * Repository name *

 octocat ▼ /

✓  octocat

 github

emorable. Need inspiration?

4. Teclea el nombre de tu repositorio, y una descripción opcional.

Owner * Repository name *

 octocat ▾ / hello-world ✓

Great repository names are short and memorable. Need inspiration? How about [vigilant-meme?](#)

Description (optional)

5. Elige la visibilidad del repositorio. Para más información, consulta [Acerca de los repositorios](#).

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

Realiza la invitación desde github por invitación a colaboradores que le llega por el mail

¿Qué es un repositorio público en GitHub?

Es un repositorio que permite realizar git commit sin pedir permiso al dueño del repositorio

¿Cómo crear un repositorio público en GitHub?

Al momento de crear el repositorio puedo elegir si lo quiero publico o privado

¿Cómo compartir un repositorio público en GitHub?

Para conectarse al repositorio desde un entorno local, es imprescindible clonar el repositorio utilizando el comando git clone en la terminal. Por ejemplo: «`bash git clone`»

