

Documentación de Procedimientos para la Creacion de un repositorio en Git

Julio Cesar Callañaupa Sallo

August 30, 2024

1 Paso 1: Instalar Git

Para comenzar a utilizar Git, primero necesitamos instalarlo en nuestra máquina. En el sistema Ubuntu, esto se realiza con el siguiente comando:

```
sudo apt update
sudo apt install git
```

Este comando actualizará la lista de paquetes disponibles e instalará Git en tu sistema. Una vez completada la instalación, puedes verificar la versión instalada para asegurarte de que Git está correctamente instalado:

```
git --version
```

Este comando debería mostrar la versión de Git que se ha instalado.

2 Paso 2: Configurar Git

Después de instalar Git, el siguiente paso es configurarlo con tu información personal. Esto permite que Git registre tus cambios con tu nombre y correo electrónico.

Primero, configura tu nombre de usuario:

```
git config --global user.name "Tu Nombre"
```

Luego, configura tu dirección de correo electrónico:

```
git config --global user.email "tuemail@example.com"
```

Estos datos serán utilizados para todos los commits que realices. Puedes verificar tu configuración con:

```
git config --list
```

Este comando muestra la configuración actual de Git, incluyendo tu nombre de usuario y correo electrónico.

3 Paso 3: Crear un Repositorio Local

Para crear un nuevo repositorio local, primero debes seleccionar la carpeta donde deseas almacenar el repositorio. Luego, inicializa el repositorio con el siguiente comando:

```
git init
```

Este comando crea un nuevo directorio `‘.git’` en la carpeta actual, que contiene todos los archivos necesarios para el repositorio de Git. No es necesario especificar un nombre para el repositorio en este paso, ya que Git utilizará la carpeta en la que se encuentra como el nombre del repositorio.

Para verificar que el repositorio se ha inicializado correctamente, puedes usar el comando:

```
ls -a
```

Este comando lista todos los archivos, incluyendo el directorio oculto `‘.git’`, que indica que el repositorio ha sido creado.

4 Paso 4: Agregar Archivos y Hacer Commits

Una vez que el repositorio local está creado, puedes agregar archivos al repositorio y hacer commits para guardar los cambios.

1. ****Agregar archivos al área de preparación (staging area):**** Usa el comando ‘git add’ para añadir archivos a la zona de preparación. Por ejemplo, para agregar todos los archivos en el directorio actual, usa:

```
git add .
```

Para agregar un archivo específico, usa:

```
git add nombre_del_archivo
```

2. ****Hacer un commit:**** Después de agregar archivos al área de preparación, puedes hacer un commit para guardar estos cambios en el historial del repositorio. Usa el siguiente comando, incluyendo un mensaje descriptivo entre comillas:

```
git commit -m "Mensaje del commit"
```

Este comando registra los archivos añadidos y el mensaje proporcionado en el historial del repositorio. Para verificar los commits realizados, usa:

```
git log
```

Este comando muestra el historial de commits con detalles como el ID del commit, el autor, la fecha y el mensaje del commit.

5 Paso 5: Crear un Repositorio en GitHub

Para compartir tu repositorio local en GitHub, primero necesitas crear un repositorio en la plataforma GitHub:

1. ****Accede a GitHub:**** Inicia sesión en tu cuenta de GitHub y dirígete a la página principal.
2. ****Crear un nuevo repositorio:**** - Haz clic en el botón **“New”** o **“New repository”** en la página principal de tu perfil. - Rellena los detalles del repositorio, como el nombre del repositorio y una breve descripción. - Puedes elegir si deseas que el repositorio sea público o privado. - No marques la opción de inicializar el repositorio con un README, ya que tu repositorio local ya está inicializado. - Haz clic en **“Create repository”** para crear el repositorio en GitHub.

Una vez creado, GitHub te proporcionará la URL del repositorio, que se usará para vincular tu repositorio local con el repositorio en la nube.

6 Paso 6: Vincular el Repositorio Local con GitHub

Para conectar tu repositorio local con el repositorio que creaste en GitHub, debes añadir una referencia remota y luego empujar tus cambios. Sigue estos pasos:

1. ****Añadir la URL del repositorio remoto:**** Usa el siguiente comando para añadir el repositorio de GitHub como un remoto llamado ‘origin’. **Sustituye url del repositorio con la URL proporcionada por GitHub - generalmente termina en ‘.git’.**

```
git remote add origin URL_DEL_REPOSITORIO
```

2. ****Verificar la configuración remota:**** Para asegurarte de que la URL remota se ha añadido correctamente, puedes usar:

```
git remote -v
```

Este comando mostrará las URLs configuradas para las operaciones de ‘fetch’ y ‘push’.

3. ****Subir los cambios al repositorio remoto:**** Una vez que hayas añadido el remoto, puedes subir tus commits al repositorio de GitHub con:

```
git push -u origin main
```

Sustituye ‘main’ con el nombre de tu rama principal si es diferente (por ejemplo, ‘master’). El parámetro ‘-u’ establece ‘origin’ como el remoto por defecto para futuros ‘push’ y ‘pull’.

7 Paso 7: Subir Cambios a GitHub

Después de vincular tu repositorio local con el repositorio en GitHub, puedes subir tus cambios al repositorio remoto. Para hacerlo, sigue estos pasos:

1. ****Subir los cambios al repositorio remoto:**** Usa el siguiente comando para enviar tus commits al repositorio de GitHub. Asegúrate de reemplazar ‘main’ con el nombre de tu rama principal si es diferente (por ejemplo, ‘master’):

```
git push origin main
```

Este comando empuja tus cambios al repositorio remoto configurado en ‘origin’ bajo la rama ‘main’. puedes saber si tu rama principal esta usando main o master con el siguiente comando:

```
git branch
```

y cambiar segun sea el caso

2. ****Autenticación:**** Si es la primera vez que estás empujando cambios al repositorio remoto, Git te pedirá que ingreses tu nombre de usuario y token de acceso de GitHub en lugar de una contraseña. Usa tu nombre de usuario de GitHub y el token generado en lugar de una contraseña para la autenticación.

- ****Nombre de usuario:**** Tu nombre de usuario de GitHub. - ****Token:**** El token de acceso personal generado desde tu cuenta de GitHub.

Después de realizar estos pasos, tus cambios locales estarán reflejados en el repositorio remoto en GitHub.