

tutorial paso a paso con los comandos para que tus alumnos puedan subir un proyecto a GitHub, incluyendo archivos grandes con Git LFS.

---

## Tutorial: Subir proyecto a GitHub con archivos grandes

Este tutorial te guiará para subir tu carpeta de proyecto a un nuevo repositorio de GitHub, configurando correctamente Git LFS para manejar archivos grandes como videos.

---

### Paso 1: Preparar tu carpeta de proyecto

Abre la línea de comandos (CMD) en la carpeta de tu proyecto. Asegúrate de que no contenga una carpeta oculta llamada **.git**. Si existe, bórrala.

### Paso 2: Inicializar Git y configurar Git LFS

Ejecuta estos comandos para iniciar un nuevo repositorio en tu carpeta local y configurarlo para manejar archivos grandes de video.

Bash

```
# Inicializa Git en la carpeta actual
```

```
git init
```

```
# Configura Git LFS para que funcione en tu computadora  
git lfs install
```

```
# Le dice a Git LFS que rastree los archivos .mp4 y .wmv  
git lfs track "*.mp4"  
git lfs track "*.wmv"
```

### Paso 3: Agregar y confirmar tus archivos

Ahora, agrega todos los archivos de tu proyecto, incluyendo el nuevo archivo `.gitattributes` que contiene la configuración de Git LFS, y haz tu primera confirmación.

Bash

```
# Agrega todos los archivos a la cola de confirmación
```

```
git add .
```

```
# Crea el primer commit con un mensaje descriptivo
```

```
git commit -m "Initial commit with LFS configured"
```

### Paso 4: Vincular con GitHub y subir los archivos

Finalmente, crea un nuevo repositorio **vacío** en GitHub (sin README, `.gitignore`, ni licencia). Luego, vincula tu carpeta local y sube todos los archivos.

Bash

```
# Vincula tu carpeta local con el repositorio remoto
```

```
git remote add origin https://github.com/profejavierc/MaterialDid-ctico.git
```

```
# Sube los archivos a la rama principal de GitHub
```

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

Una vez que el push termine, tus archivos estarán en línea en el repositorio.

# trabajar en el repositorio

## Paso 1: Acceso al repositorio

Primero, cada estudiante necesita acceso al repositorio. Tienes dos opciones principales:

- **Opción 1: Repositorio público** (más fácil). Si el repositorio es público, cualquiera puede clonarlo y trabajar en él.
  - **Opción 2: Repositorio privado** (más seguro). Si es privado, debes invitar a cada estudiante como colaborador. Para ello, ve a **Settings** -> **Collaborators** en GitHub y agrega a cada alumno por su nombre de usuario de GitHub.
- 

## Paso 2: Clonar el repositorio

Cada alumno debe descargar una copia del proyecto a su computadora. Deben abrir su línea de comandos y ejecutar el siguiente comando.

Bash

# Reemplaza la URL con la del repositorio

```
git clone https://github.com/profejavierec/MaterialDid-ctico.git
```

Esto creará una copia local del proyecto en su máquina.

---

## Paso 3: Realizar cambios y subirlos

Una vez que un alumno haya hecho cambios en los archivos (por ejemplo, ha editado un archivo de Word o agregado un nuevo video), debe seguir estos pasos para subir sus cambios.

### Agrega los archivos modificados:

Bash

```
git add .
```

1.

### Confirma los cambios con un mensaje:

Bash

```
git commit -m "Descripción de los cambios realizados"
```

2.

**Sincroniza los cambios con el repositorio en línea:** Antes de subir, es vital que descarguen los cambios que otros compañeros hayan subido.

Bash

```
# Descarga y fusiona los cambios de GitHub
```

```
git pull origin main
```

3. Si hay conflictos (cuando dos personas cambian la misma línea en un archivo), deberán resolverlos manualmente.

**Sube los cambios:**

Bash

```
git push origin main
```

4.

Este proceso de **pull**, **commit** y **push** les permite trabajar de forma colaborativa sin sobrescribir el trabajo de los demás.