INFORME: PROCESO DE CONTROL DE VERSIONES CON GIT

Fecha: 22/11/2024

Asignatura: Programación Orientada a Objetos

Tema: Control de versiones con Git

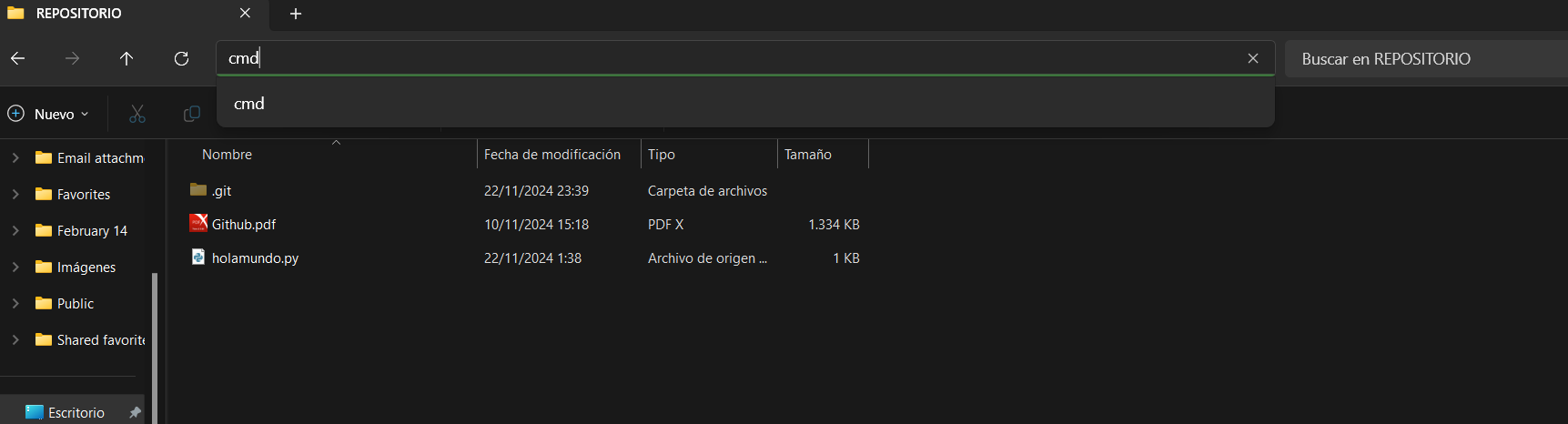
1. ANÁLISIS DE LA ESTRUCTURA DEL REPOSITORIO

En la primera imagen se observa la estructura del repositorio local que contiene:

- Una carpeta .git (repositorio Git)

- Un archivo Github.pdf de 1.334 KB

- Un archivo holamundo.py de 1 KB



2. PROCESO DE CONTROL DE VERSIONES

2.1 Comandos Utilizados y su Función

1. git commit -m "prueba subir pdf"

- Función: Crea una nueva confirmación con los cambios realizados

- Resultado: Se registró 1 archivo modificado (Github.pdf)

- Explicación: El comando commit guarda los cambios en el repositorio local con un mensaje descriptivo

2. git push

- Función: Sube los cambios confirmados al repositorio remoto

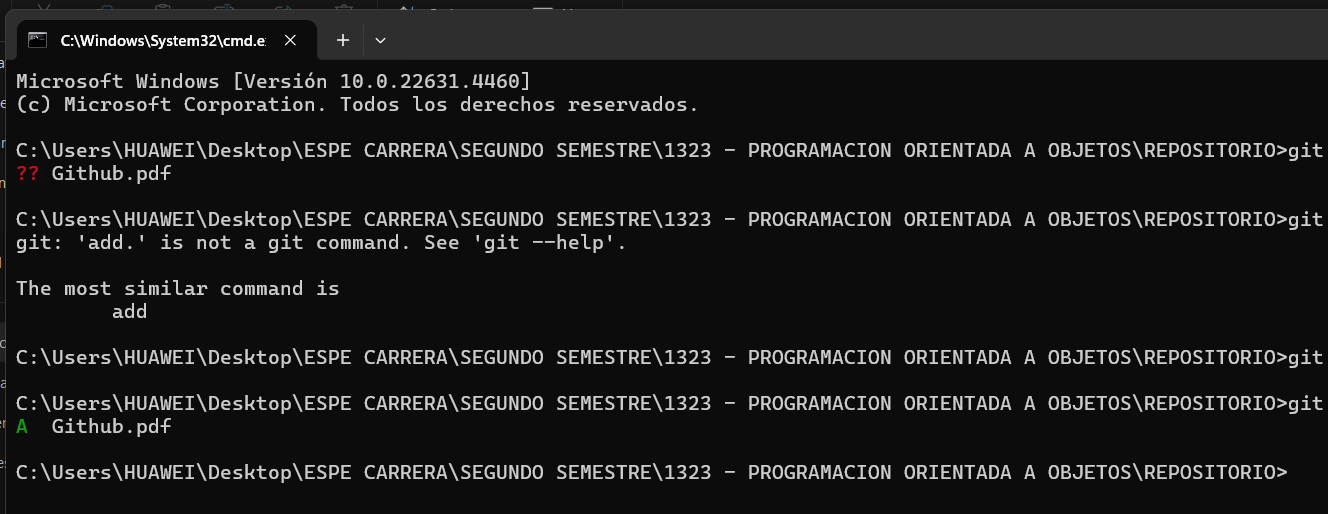
- Resultado:

- Subió exitosamente los objetos (4/4)

- Realizó compresión delta usando 8 hilos

- Transfirió 1.18 MiB a 11.31 MiB/s

- Explicación: Push envía los commits locales al repositorio remoto en GitHub



3. git ?? Github.pdf y git add Github.pdf

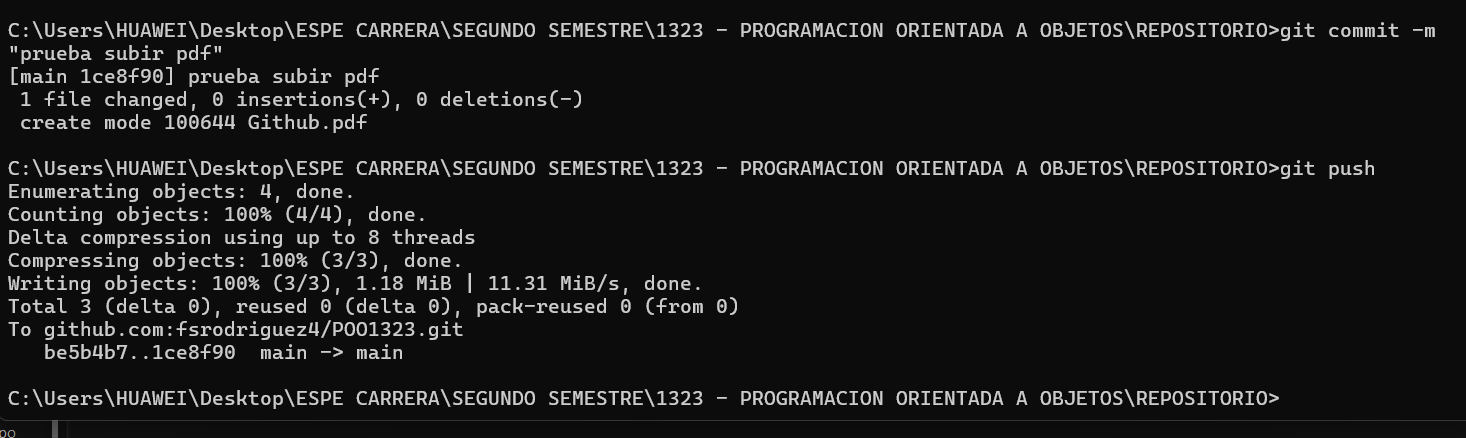
- Función:

- ?? verifica el estado del archivo

- add agrega el archivo al área de preparación

- Resultado: El comando ?? mostró error pero se corrigió usando add

- Explicación: Estos comandos son parte del flujo de trabajo para preparar archivos antes de confirmarlos



3. OBSERVACIONES IMPORTANTES

- La ruta del proyecto indica que es un trabajo académico: ESPE CARRERA\SEGUNDO SEMESTRE\1323 - PROGRAMACION ORIENTADA A OBJETO

- El repositorio remoto está alojado en: github.com:fsrodriguez4/POO1323.git

- Se manejaron correctamente los estados de Git: sin seguimiento → preparación → confirmación → subida

4. CONCLUSIONES

1. El proceso demuestra un flujo de trabajo básico pero efectivo con Git

2. Se evidencia la importancia de usar mensajes descriptivos en los commits

3. La estructura del repositorio mantiene una organización clara para el proyecto académico

5. RECOMENDACIONES

1. Utilizar siempre mensajes de commit descriptivos

2. Verificar el estado de los archivos antes de confirmar cambios

3. Mantener una estructura organizada del repositorio

4. Asegurar que los archivos estén en el área de preparación antes de hacer commit

Este informe documenta el proceso básico de control de versiones realizado, mostrando la secuencia correcta de comandos Git para gestionar cambios en un proyecto académico.