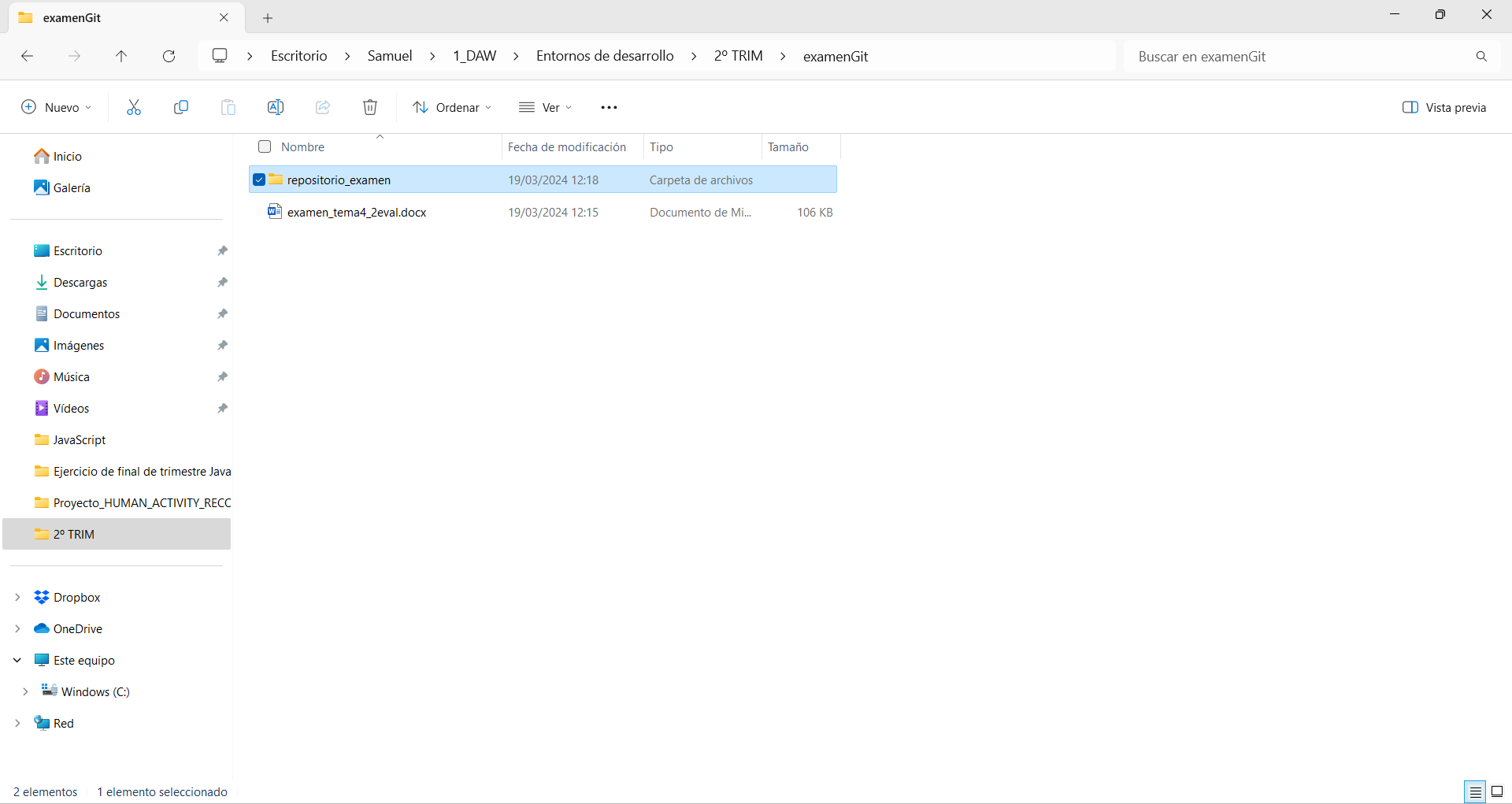
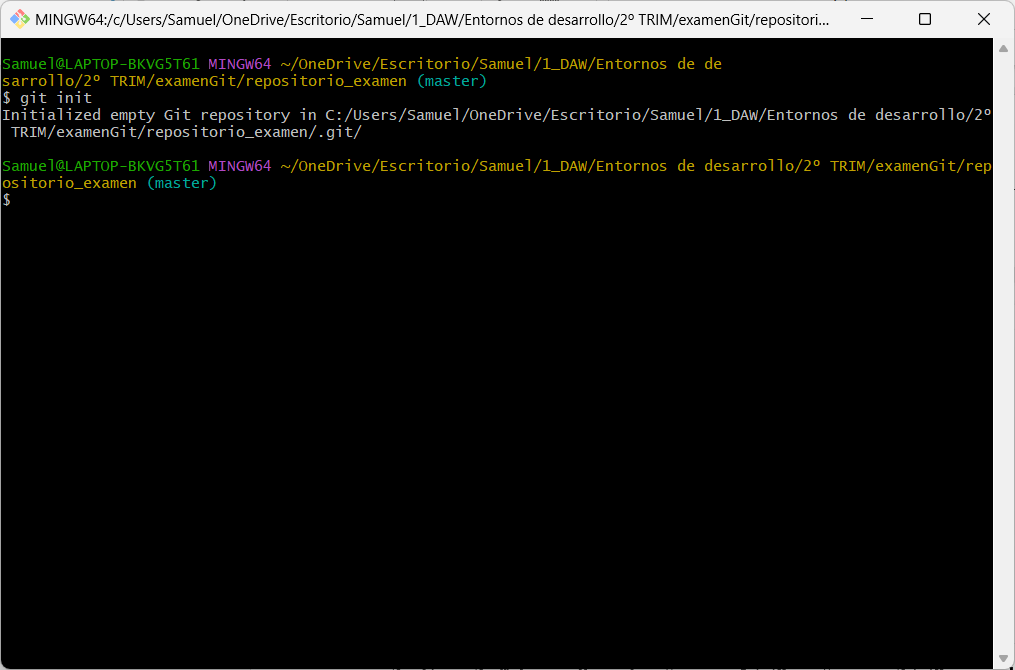
Examen Entornos de Desarrollo

19 Marzo, 2024

**Nombre y apellidos: \_\_\_Samuel Plaza Sáez\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

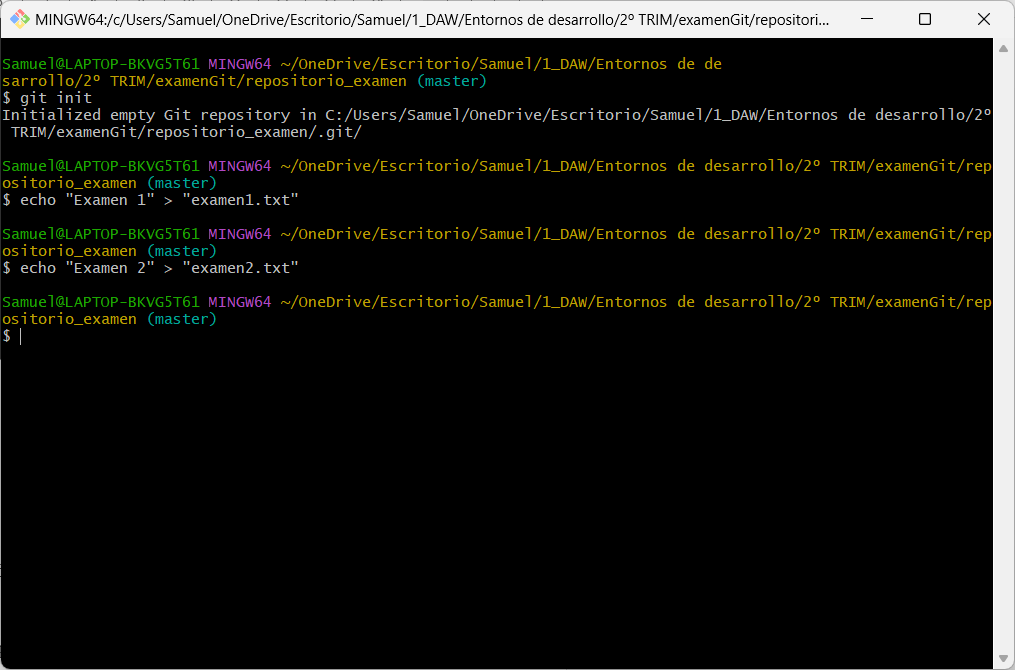
**1. Crea un repositorio local llamado repositorio\_examen.**



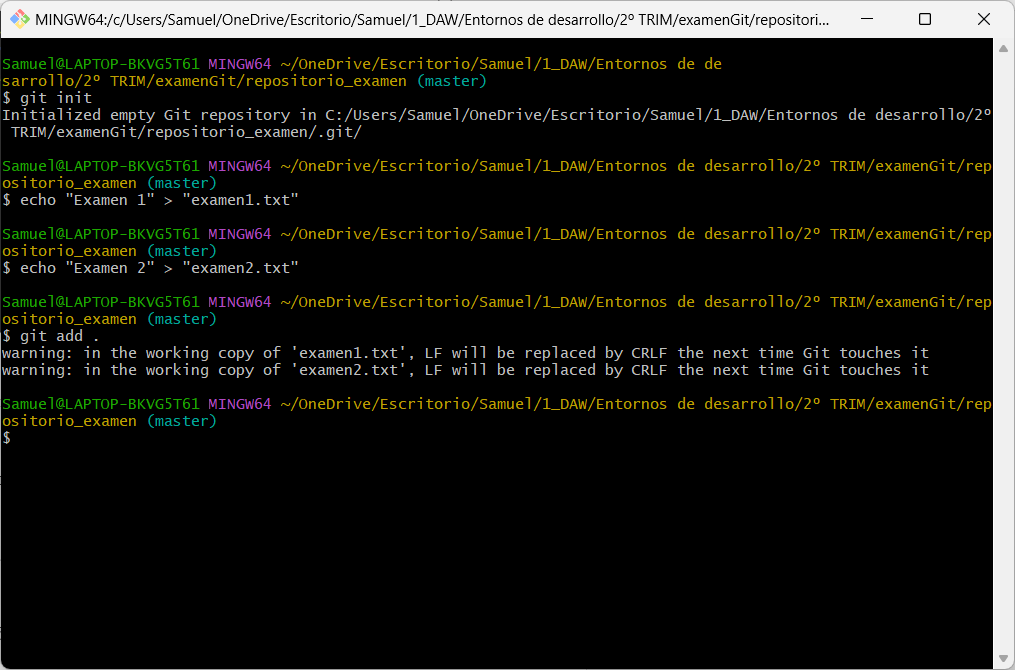


**2. Crea 2 ficheros llamados “examen1.txt” y “examen2.txt” con el siguiente contenido:**

* Examen 1
* Examen 2



**3. Añade los ficheros a la carpeta de preparación (staging).**

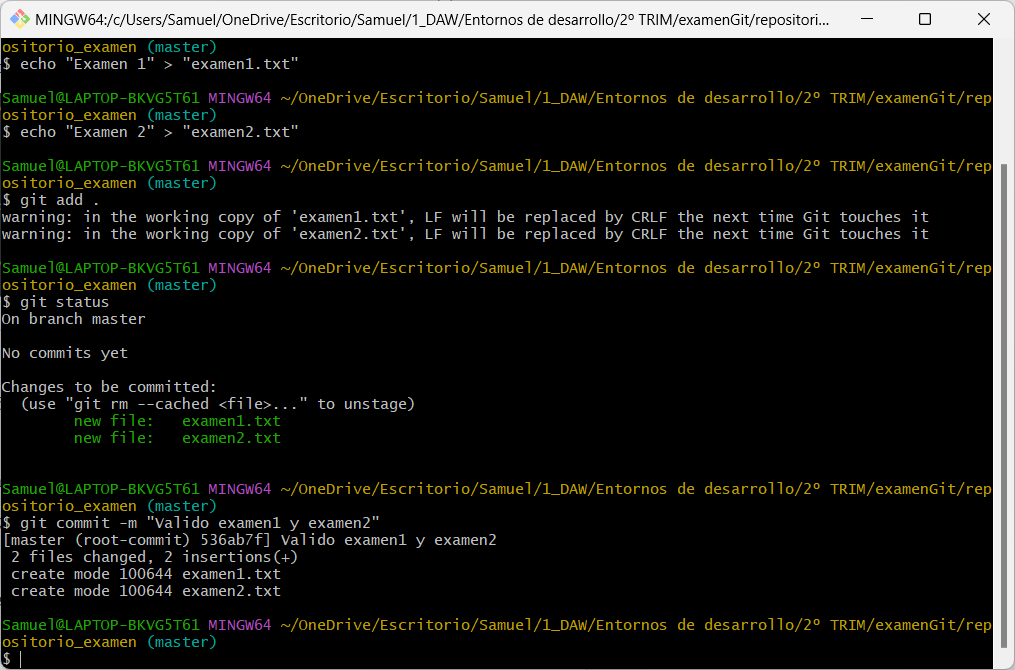


**4. Visualiza el estado actual del repositorio.**

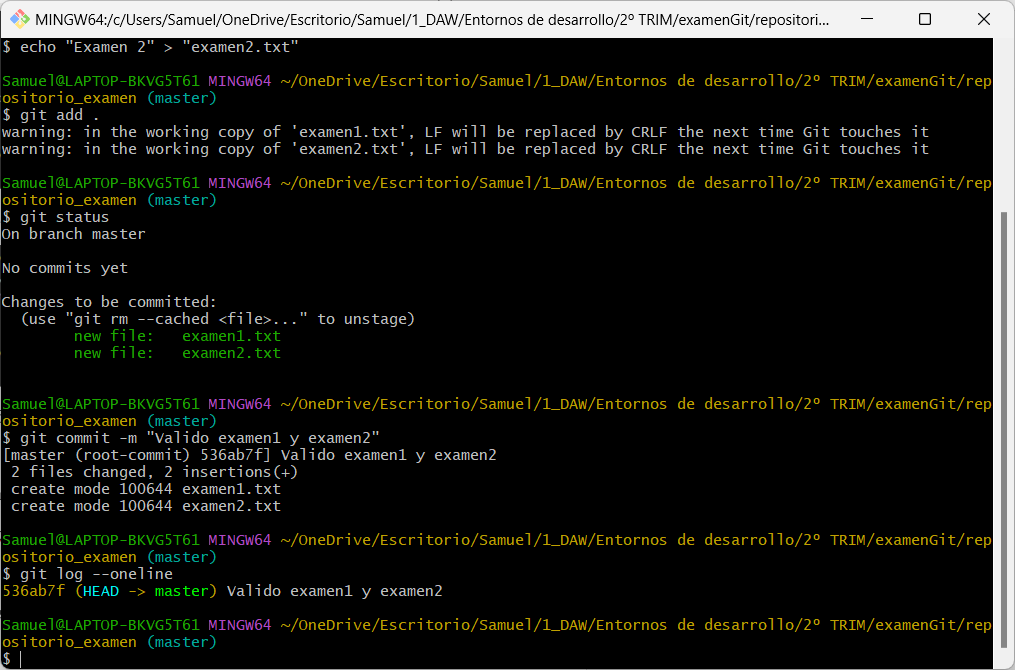
Texto

Descripción generada automáticamente

**5. Valida los cambios realizados con los mensajes “Valido examen1 y examen2”.**

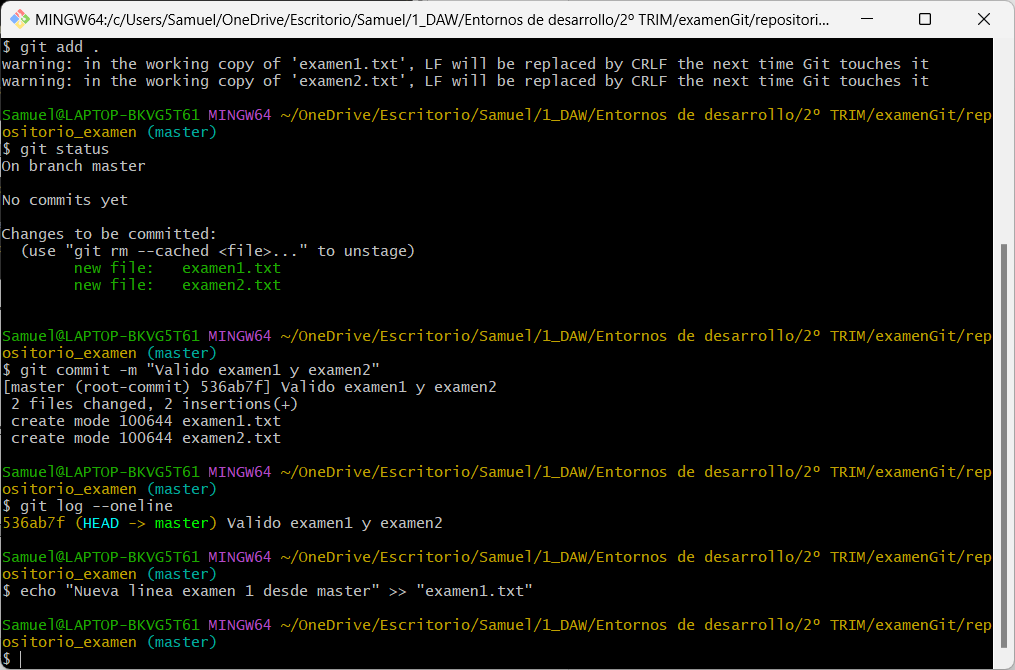


**6. Muestra todas las revisiones del repositorio.**



**7. Añade una nueva línea al fichero “examen1.txt”:**

* Nueva línea examen1 desde master

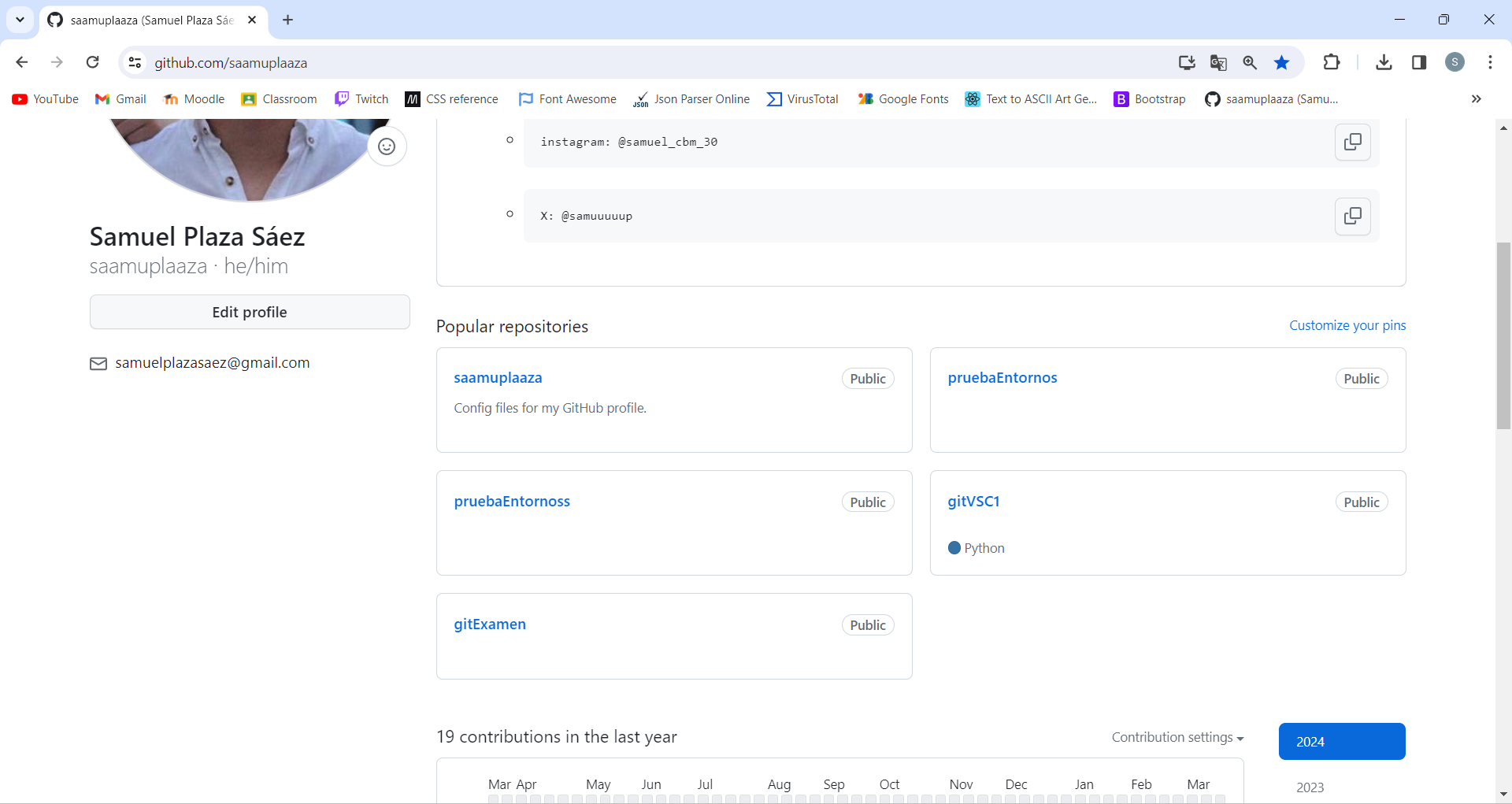


**8. Valida los cambios realizados con el mensaje “Agrego nueva línea examen1 desde master”.**

Texto

Descripción generada automáticamente

**9. Crea un repositorio en GitHub llamado “gitExamen”.**



**10. Sube lo que hay validado en el repositorio local al repositorio remoto.**

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

**11. Crea una carpeta en local llamada proyecto\_examen (abre la carpeta en Visual Studio Code y guarda el workspace como proyecto\_examenVSC). Crea el repositorio desde la terminal.**

Captura de pantalla de computadora

Descripción generada automáticamente

**12. Desde Visual Studio Code crea los ficheros “examen1.py” y “examen2.py” con el siguiente contenido respectivamente:**

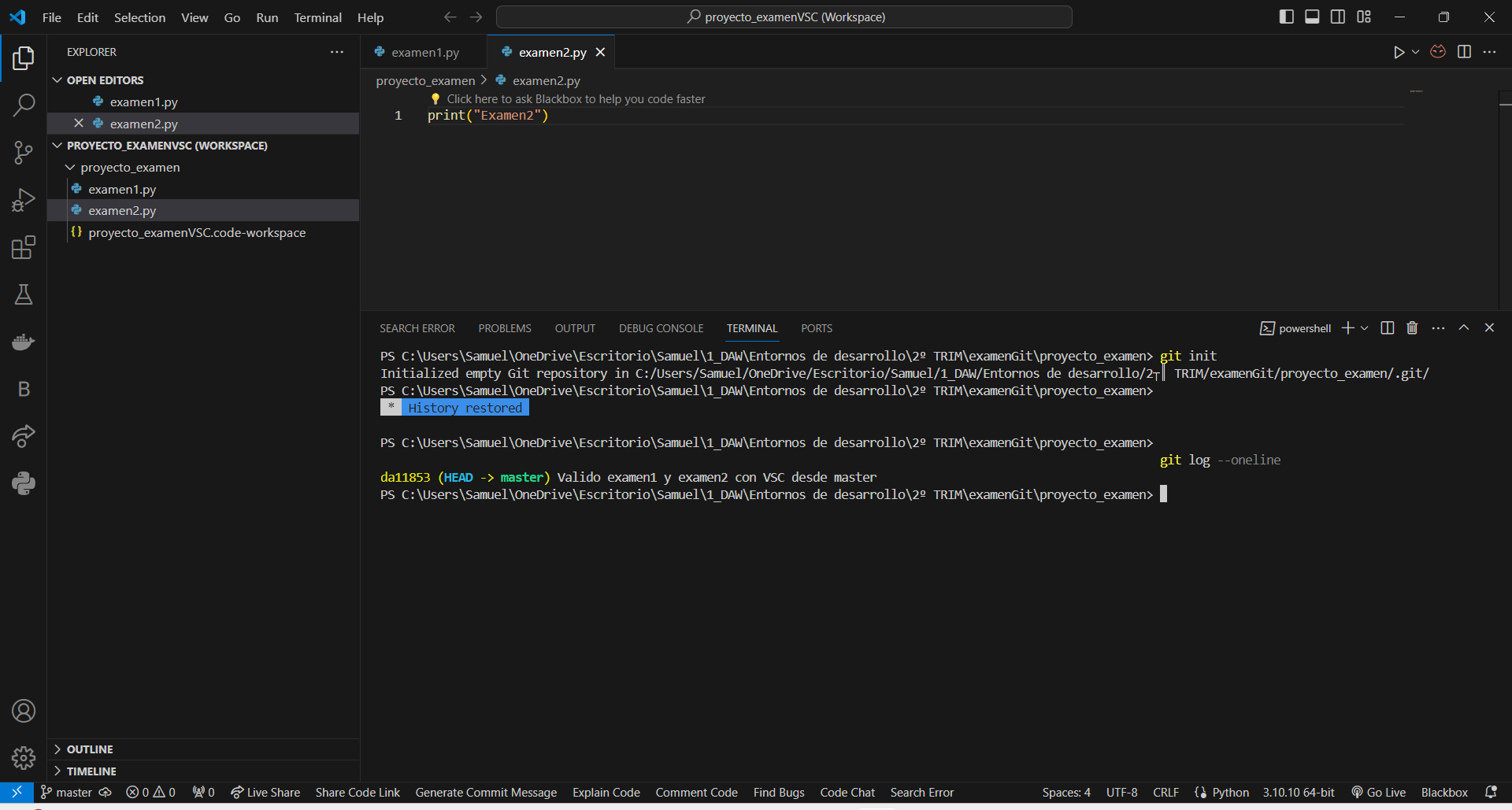
* print (“Examen1”)
* print (“Examen2”)

Captura de pantalla de computadora

Descripción generada automáticamente

**13. Valida los cambios realizados desde VSC con el mensaje “Valido examen1 y examen2 con VSC desde master”.**

**He hecho la captura después de darle al botón “commit”, por lo que adjunto captura de la terminal con el comando “git log --oneline” para ver que se hahecho el commit.**

****

**14. Crea la rama “desarrollador1” en VSC sin utilizar el terminal.**

Captura de pantalla de computadora

Descripción generada automáticamente

Captura de pantalla de computadora

Descripción generada automáticamente

**15. Añade la siguiente línea al fichero “examen1.py” desde la rama desarrollador1:**

* Nueva línea examen1 en VSC desde desarrollador1.

Captura de pantalla de computadora

Descripción generada automáticamente

**16. Comprueba con qué rama estás trabajando en VSC sin utilizar el terminal.**

**Captura de pantalla de computadora

Descripción generada automáticamente**

**17. Valida los cambios realizados con el mensaje “Agrego nueva línea examen1 en VSC desde desarrollador1”.**

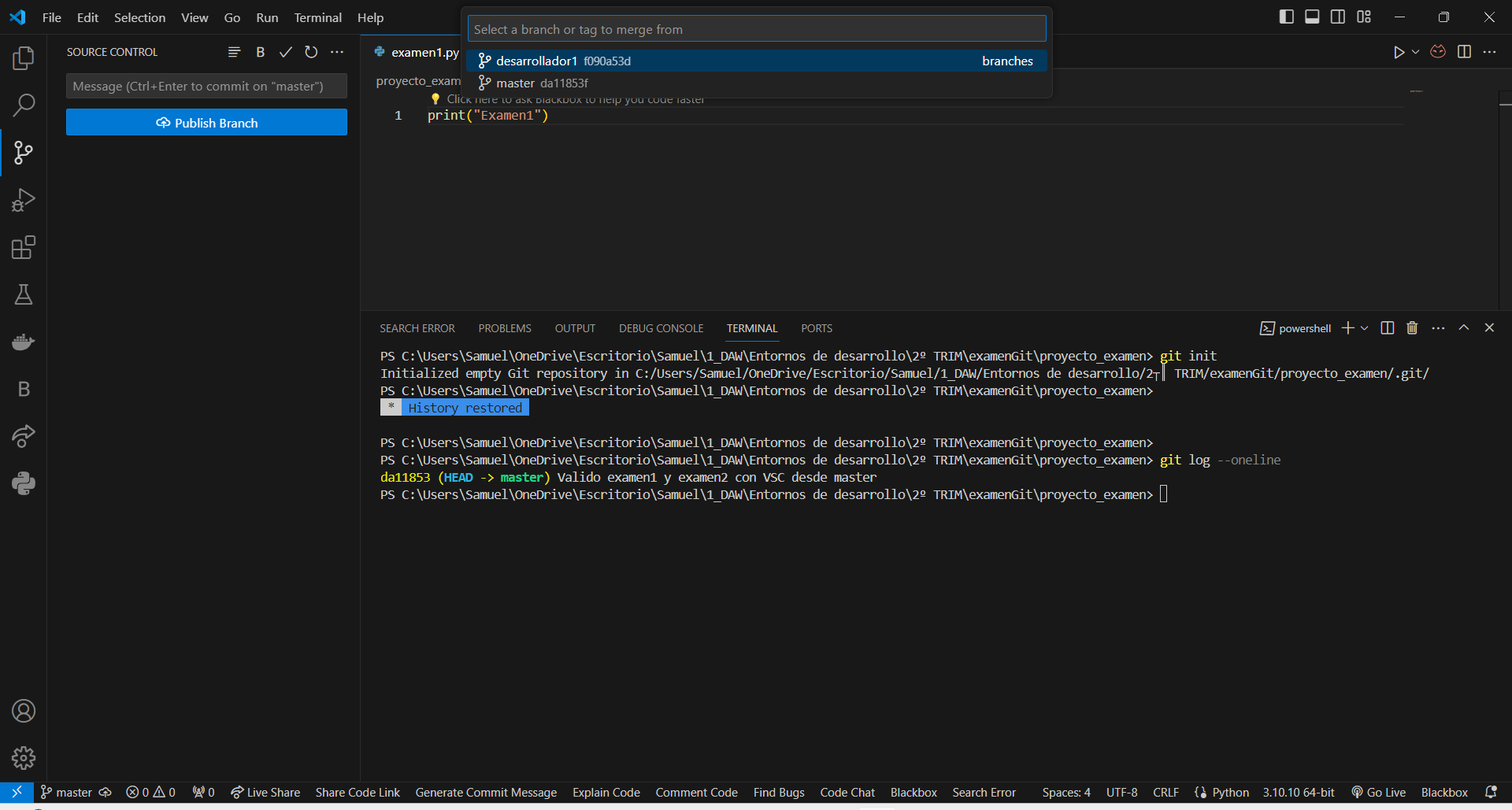
**Captura de pantalla de computadora

Descripción generada automáticamente**

**18. Fusiona la rama “master” con la rama “desarrollador1”.**

**Captura de pantalla de computadora

Descripción generada automáticamente**

****

**19. Crea un nuevo repositorio en GitHub llamado “gitExamen2” y una carpeta en local llamada “gitExamen2”.**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**20. Abre el repositorio creado anteriormente en VSC.**

**Texto

Descripción generada automáticamente con confianza media**

**21. Crea en VSC el fichero “prueba1.py” con el siguiente contenido:**

* print (“Prueba1”)

Captura de pantalla de computadora

Descripción generada automáticamente

**Valida los cambios con el mensaje “Prueba1 con VSC desde master”.**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**22. Sube los cambios a GitHub.**

**Captura de pantalla de computadora

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

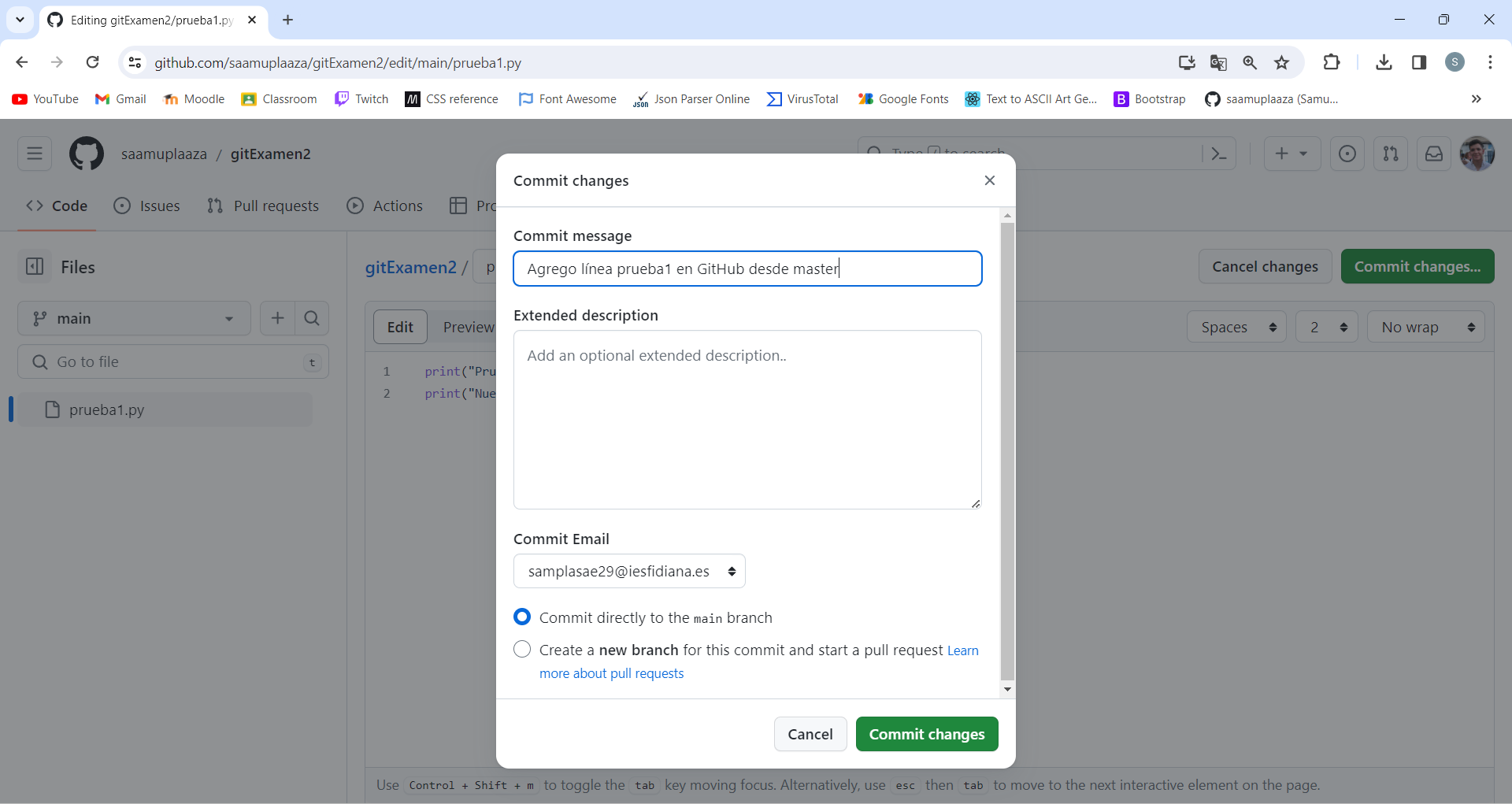
**23. En GitHub, desde la rama “master” agrega la siguiente línea al fichero “prueba1.py”:**

* print (“Nueva línea prueba1 en GitHub desde master”)

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

**Valida los cambios con el mensaje “Agrego línea prueba1 en GitHub desde master”.**

****

**24. Sincroniza los cambios.**

Captura de pantalla de computadora

Descripción generada automáticamente

**25. En VSC, desde la rama “master” agrega la siguiente línea al fichero “prueba1.py”:**

* print (“Nueva línea prueba1 en VSC desde master”)

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**Valida los cambios con el mensaje “Agrego línea prueba1 en VSC desde master”.**

**Interfaz de usuario gráfica

Descripción generada automáticamente**

**26. Desde GitHub crea la rama “desarrollador2”.**

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

**27. En la rama “desarrollador2”, crea el fichero “prueba2.py” con el siguiente contenido:**

* print (“Prueba2”)

Interfaz de usuario gráfica, Texto, Aplicación, Word, Correo electrónico

Descripción generada automáticamente

**Valida los cambios con el mensaje “Prueba2 en GitHub desde desarrollador2”.**

**Una captura de pantalla de una computadora

Descripción generada automáticamente**

**28. Desde GitHub, en la rama “desarrollador2”, añade la siguiente línea al fichero “prueba1.py”:**

* print (“Nueva línea prueba1 en GitHub desde desarrollador2”)

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

**Valida los cambios con el mensaje “Prueba1 en GitHub desde desarrollador2”.**

**Una captura de pantalla de una computadora

Descripción generada automáticamente**

**29. Fusiona ambas ramas en GitHub y sincroniza los cambios con VSC.**

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

**Captura de pantalla de computadora

Descripción generada automáticamente**

**Captura de pantalla de computadora

Descripción generada automáticamente**

**Captura de pantalla de computadora

Descripción generada automáticamente**

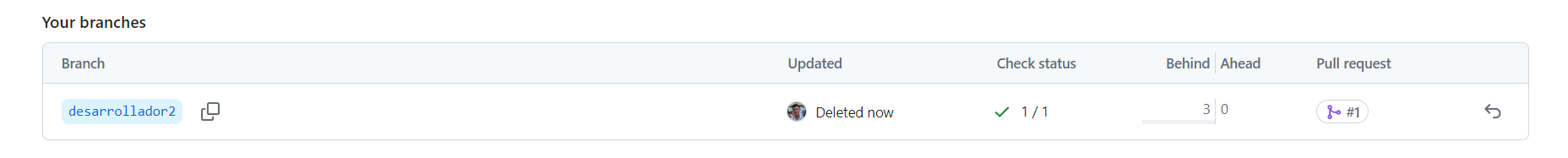
**Captura de pantalla de computadora

Descripción generada automáticamente**

**30. Elimina la rama “desarrollador2” desde GitHub. Crea un repositorio en local llamado repclonado y clona el repositorio remoto gitExamen2.**

**Una captura de pantalla de una computadora

Descripción generada automáticamente**

****

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

**Texto

Descripción generada automáticamente**

**Nota: Del ejercicio 1 al ejercicio 29 (ambos inclusive) valen 0.33 puntos. El ejercicio 30 vale 0,43 puntos. Adjuntar captura de pantalla de cada uno de los ejercicios.**

**Se evalúa el RA4 del módulo.**