Control de versiones

LÁZARO JAVIER HERNÁNDEZ

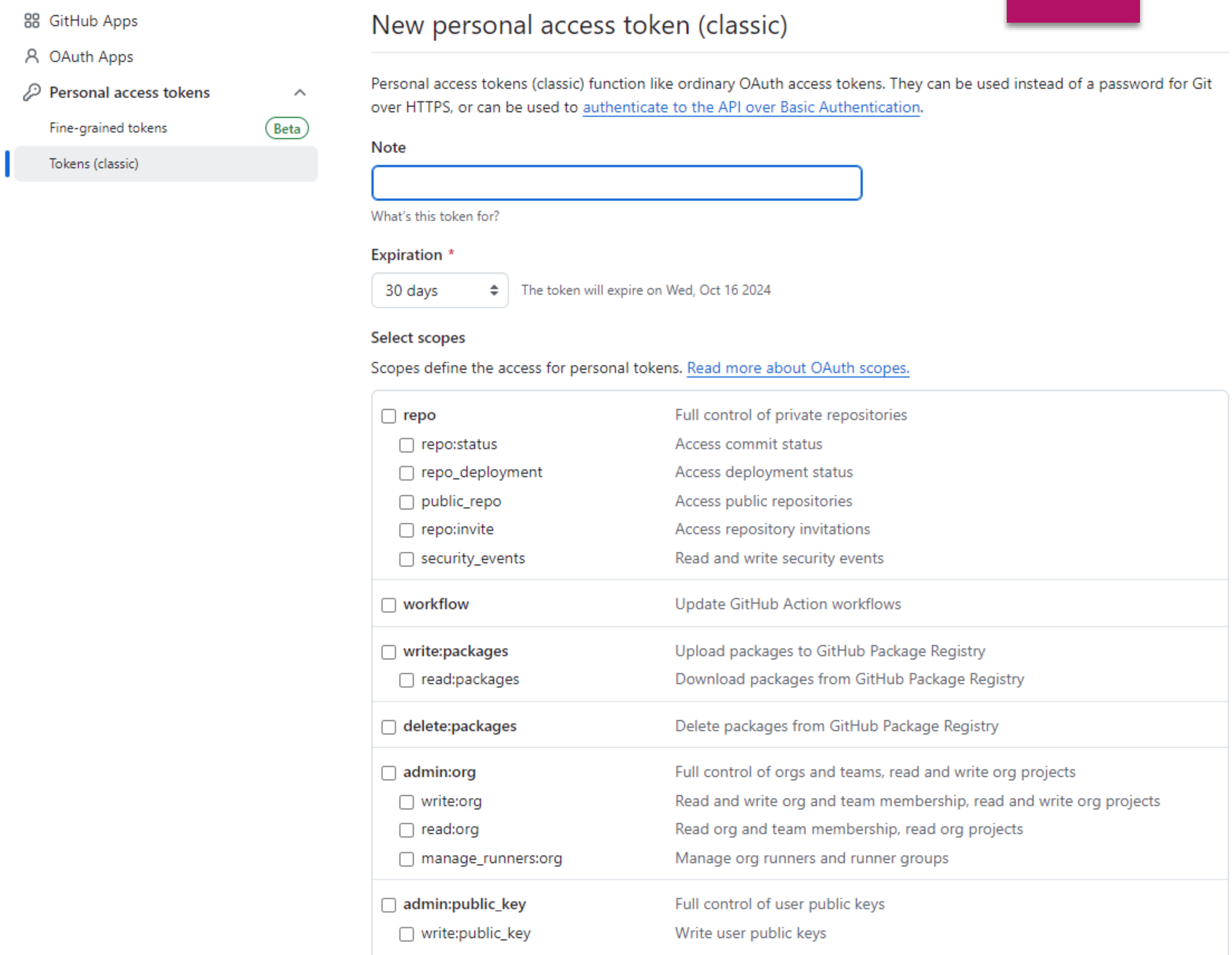
Formas de autenticarse con Git

GitHub authentication

El método de autenticación con **usuario y contraseña** en GitHub ha sido reemplazado debido a políticas de seguridad más estrictas. Ahora, GitHub usa principalmente **Tokens de Acceso Personal (PAT)** o **autenticación SSH** para clonar repositorios, especialmente cuando son **privados**.

Pasos para generar un Token de Acceso Personal:

- Acceder a GitHub:Los estudiantes deben iniciar sesión en su cuenta de GitHub.
- Ir a Configuración:Hacer clic en su foto de perfil (arriba a la derecha) y seleccionar Settings.
- Seleccionar "Developer Settings":En la barra lateral izquierda, hacer clic en Developer settings.
- Crear un Token:Ir a Personal Access Tokens > Tokens (classic) y hacer clic en Generate new token.
- Definir permisos:Los estudiantes deben asignar permisos adecuados al token, como acceso a repositorios privados (repo). Luego, copiar el token generado.
- Usar el Token:Al clonar el repositorio, cuando se les solicite usuario y contraseña, los estudiantes deben usar su nombre de usuario como el usuario, y el token generado como contraseña.



New personal access token (classic)

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

Note

What's this token for?

Expiration *

30 days — The token will expire on Wed, Oct 16 2024

Select scopes

Scopes define the access for personal tokens. Read more about OAuth scopes.

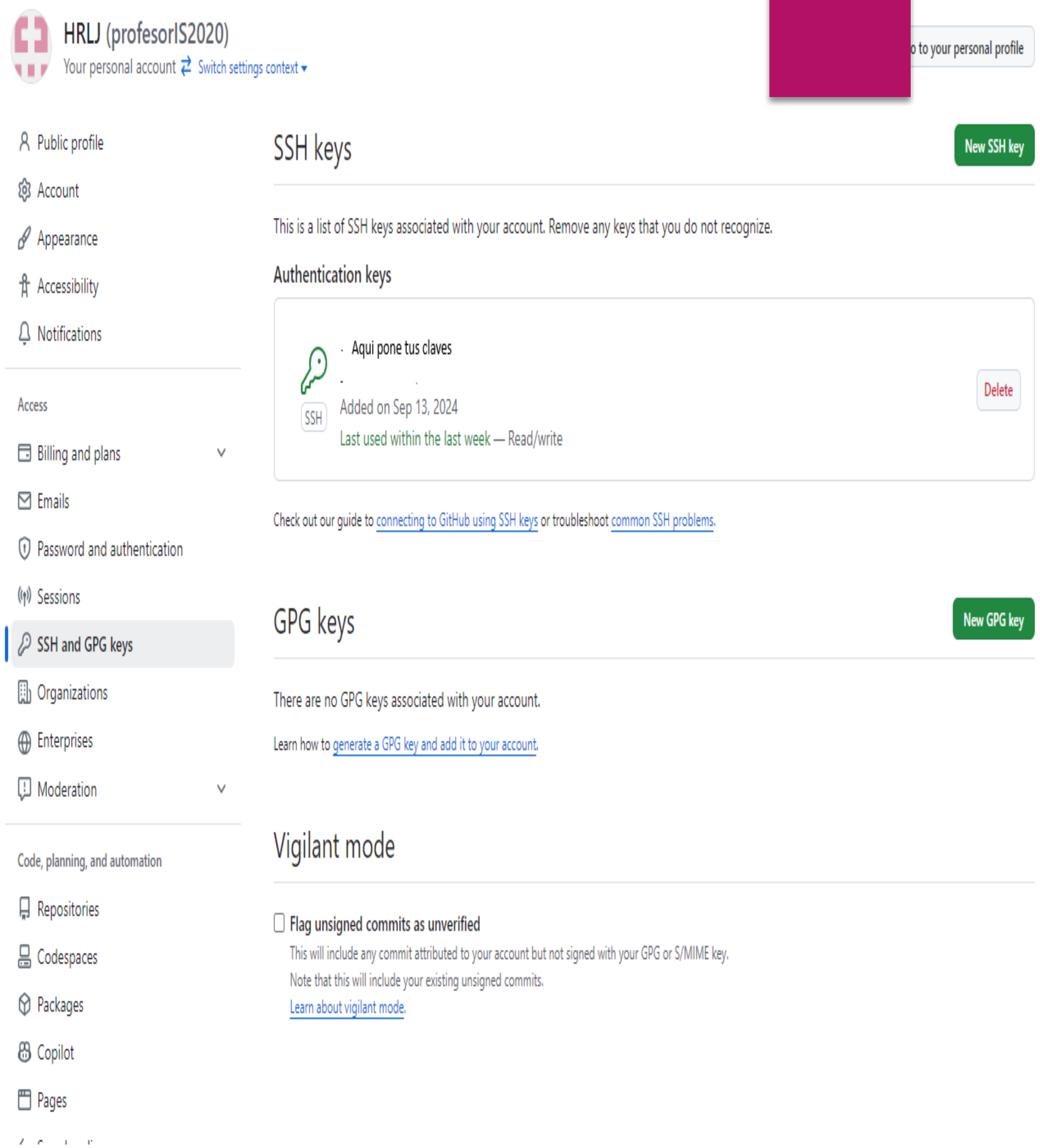
☐ **repo**	Full control of private repositories
☐ repo:status	Access commit status
☐ repo_deployment	Access deployment status
☐ public_repo	Access public repositories
☐ repo:invite	Access repository invitations
☐ security_events	Read and write security events
☐ **workflow**	Update GitHub Action workflows
☐ **write:packages**	Upload packages to GitHub Package Registry
☐ read:packages	Download packages from GitHub Package Registry
☐ **delete:packages**	Delete packages from GitHub Package Registry
☐ **admin:org**	Full control of orgs and teams, read and write org projects
☐ write:org	Read and write org and team membership, read and write org projects
☐ read:org	Read org and team membership, read org projects
☐ manage_runners:org	Manage org runners and runner groups
☐ **admin:public_key**	Full control of user public keys
☐ write:public_key	Write user public keys

Generate ssh key

Generar ssh key

- Step by Step:
  - Generar la clave ssh
    - ssh-keygen -t rsa -b 4096 -C "lazaro.hernandez@uneatlantico.com"
  - Start the SSH agent in the background:
    - eval "$(ssh-agent -s)"
  - Add Your SSH Key to the Agent
    - ssh-add ~/.ssh/id_rsa

    ….(virguilla alt+ñ)
  - Copy the SSH public key to your clipboard:
    - cat ~/.ssh/id_rsa.pub
  - Now, go to your GitHub account settings:
    - Go to Settings> SSH and GPG keys> New SSH key.Paste the SSH key into the key field and give it a recognizable title (like "Ubuntu VM"))



HRLJ (profesorIS2020)

Your personal account ⇄ Switch settings context ▾

o to your personal profile

SSH keys

New SSH key

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

Authentication keys

Aqui pone tus claves

SSH

Added on Sep 13, 2024

Last used within the last week — Read/write

Delete

Check out our guide to connecting to GitHub using SSH keys or troubleshoot common SSH problems.

GPG keys

New GPG key

There are no GPG keys associated with your account.

Learn how to generate a GPG key and add it to your account.

Vigilant mode

□ Flag unsigned commits as unverified

This will include any commit attributed to your account but not signed with your GPG or S/MIME key.

Note that this will include your existing unsigned commits.

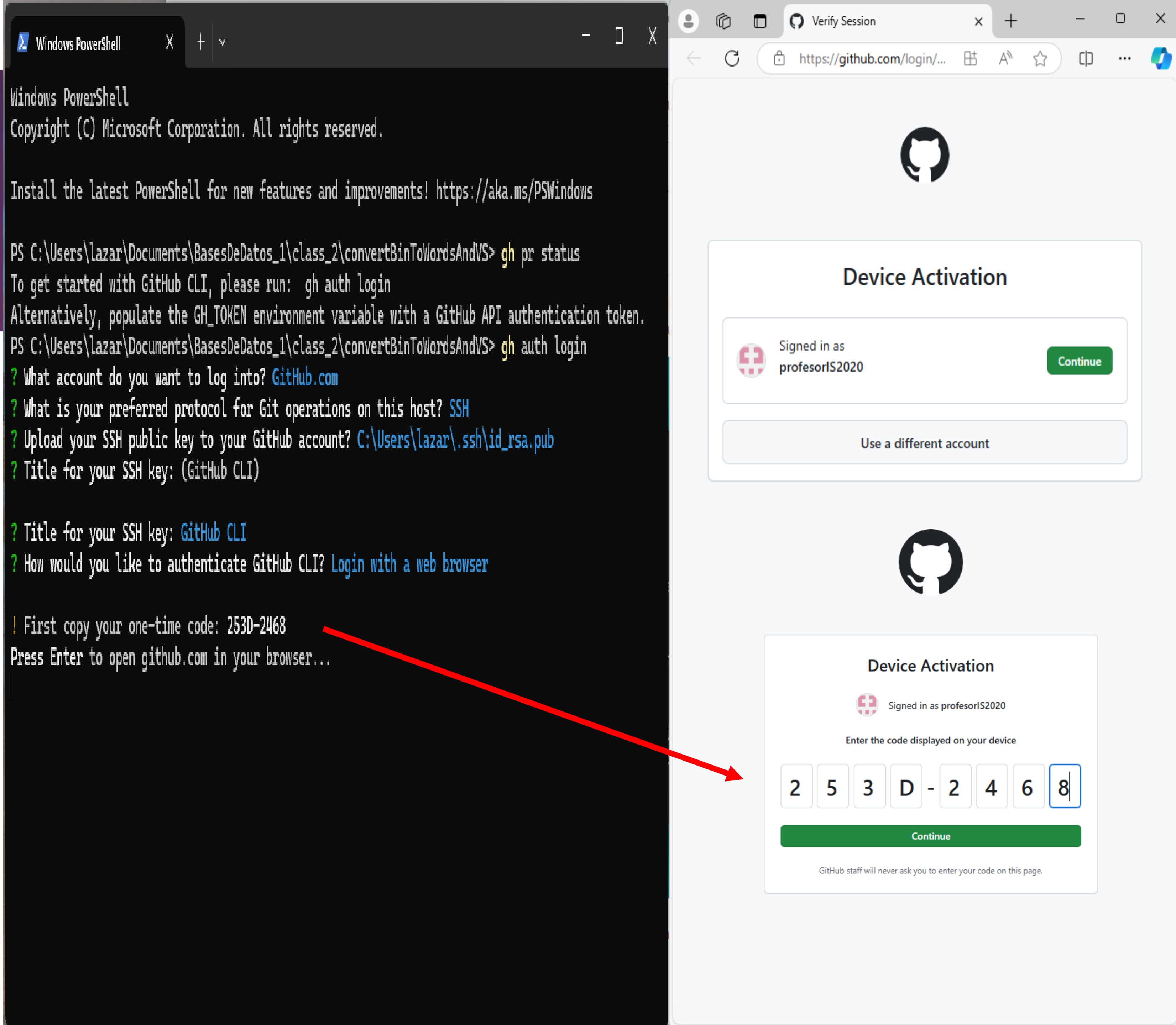
Learn about vigilant mode.

Github CLI

GitHub CLIC

Pasos para usar GitHub CLI:

- **Instalar GitHub CLI**:
  - Los estudiantes deben instalar la CLI desde GitHub CLI.
- **Autenticarse**:
  - Ejecutar el comando gh `auth login`, que abrirá el navegador y les permitirá autenticarse.
- **Clonar el repositorio**:
  - Una vez autenticados, pueden usar gh `repo clone` para clonar cualquier repositorio que tengan permitido ver:
  - bash
  - Copiar código
  - `gh repo clone organizacion/nombre-del-repo`



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\lazar\Documents\BasesDeDatos_1\class_2\convertBinToWordsAndVS> gh pr status
To get started with GitHub CLI, please run:  gh auth login
Alternatively, populate the GH_TOKEN environment variable with a GitHub API authentication token.
PS C:\Users\lazar\Documents\BasesDeDatos_1\class_2\convertBinToWordsAndVS> gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations on this host? SSH
? Upload your SSH public key to your GitHub account? C:\Users\lazar\.ssh\id_rsa.pub
? Title for your SSH key: (GitHub CLI)

? Title for your SSH key: GitHub CLI
? How would you like to authenticate GitHub CLI? Login with a web browser

! First copy your one-time code: 253D-2468
Press Enter to open github.com in your browser...
```

Device Activation

Signed in as profesorIS2020

Device Activation

Signed in as profesorIS2020

Enter the code displayed on your device

2 5 3 D - 2 4 6 8

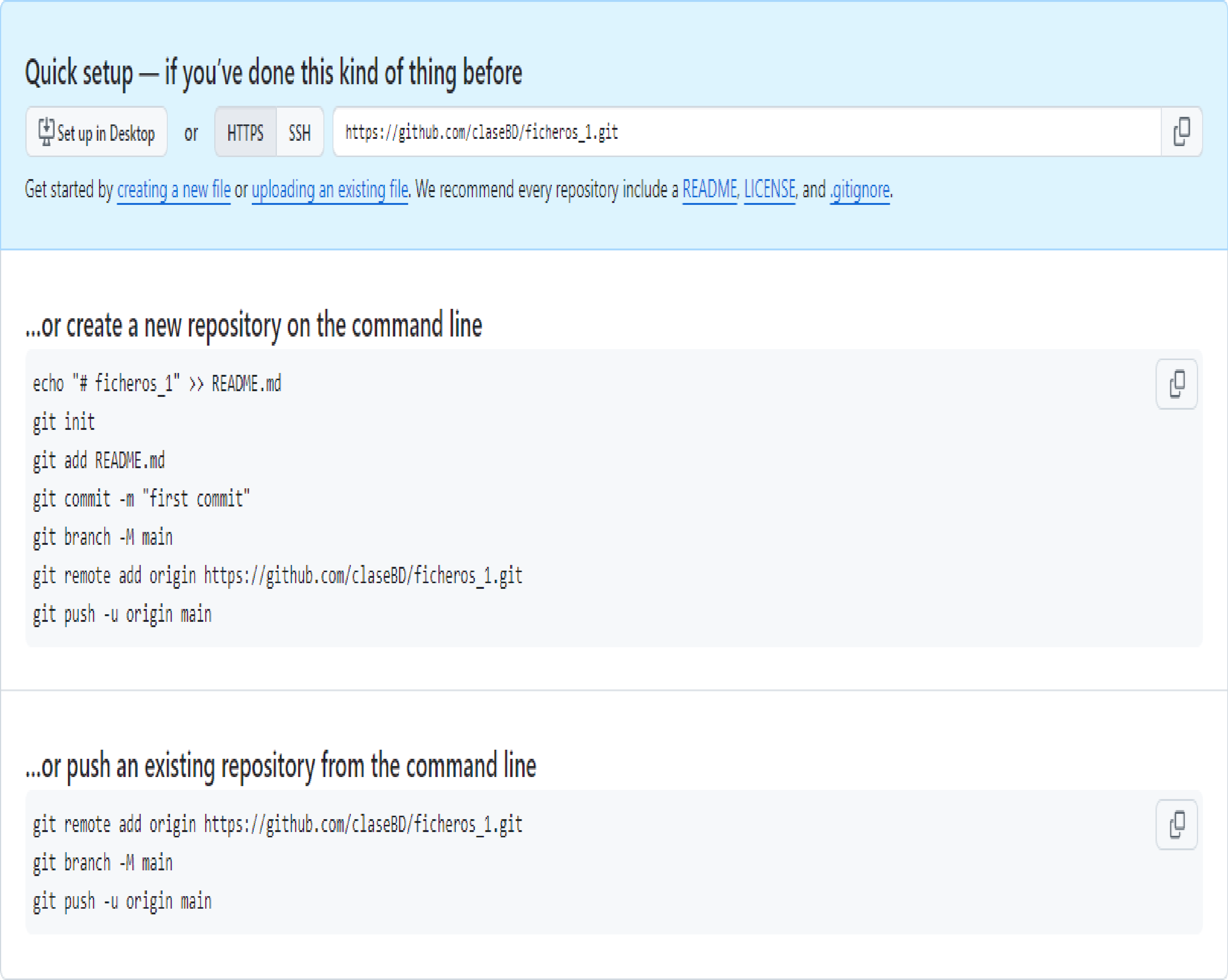
GitHub staff will never ask you to enter your code on this page.

Inicio con git y github

Construir un repositorio remoto

Step by Step:

- Entrar a tu cuenta de github.
  - Crear un nuevo repositorio con el nombre "ficheros_1".
  - No crear archivo README.md
- Configurar git local y aplicar "Quick setup"



Quick setup — if you've done this kind of thing before

```
https://github.com/claseBD/ficheros_1.git
```

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

…or create a new repository on the command line

```
echo "# ficheros_1" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/claseBD/ficheros_1.git
git push -u origin main
```

…or push an existing repository from the command line

```
git remote add origin https://github.com/claseBD/ficheros_1.git
git branch -M main
git push -u origin main
```

Setup y Configuración

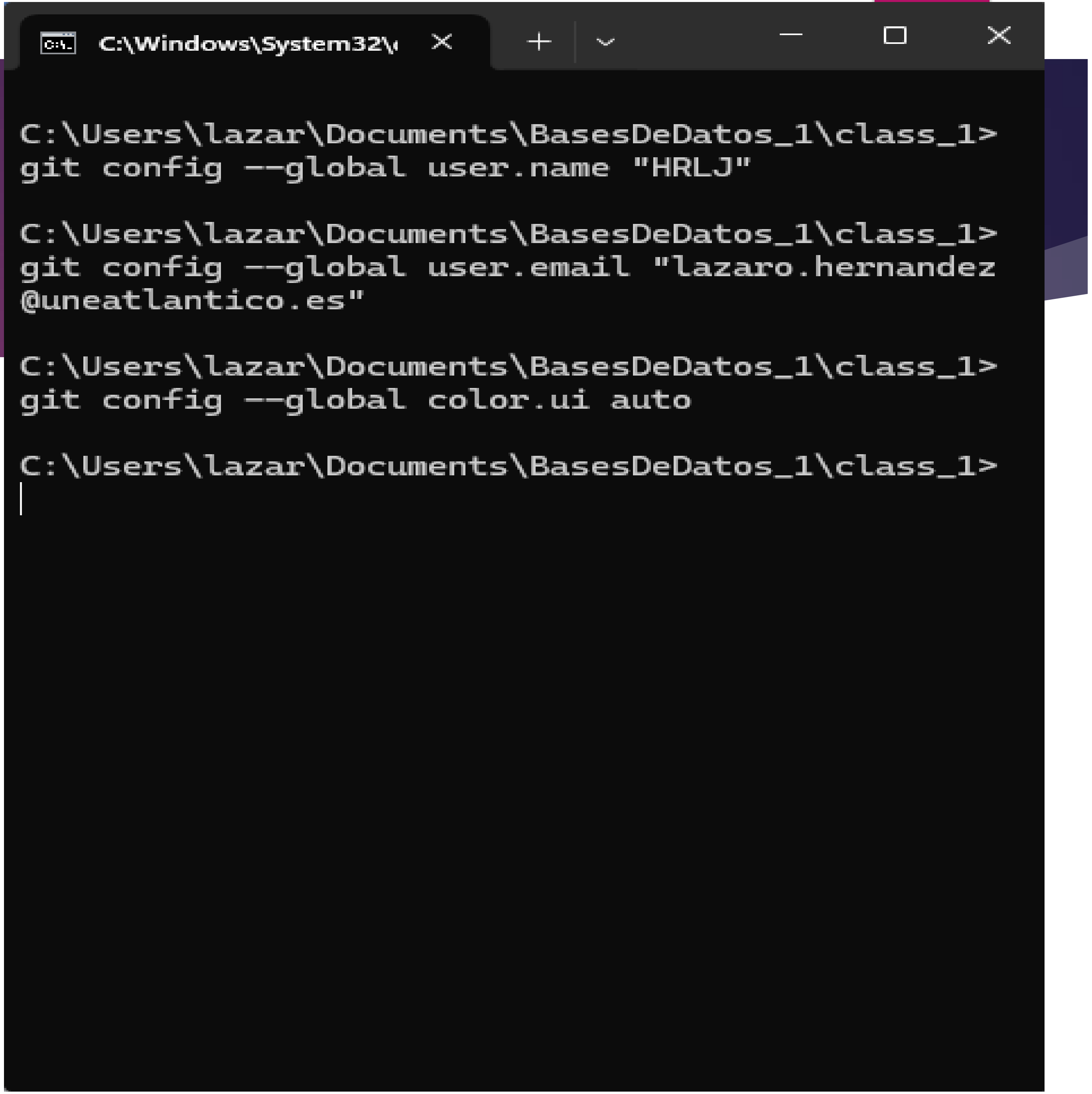
Configuración:

git config - -global user.name “HRLJ”

git config - - global user.email lazaro.hernandez@uneatlantico.es

git config - - global init.defaultBranch <name>

git config –global color.ui auto

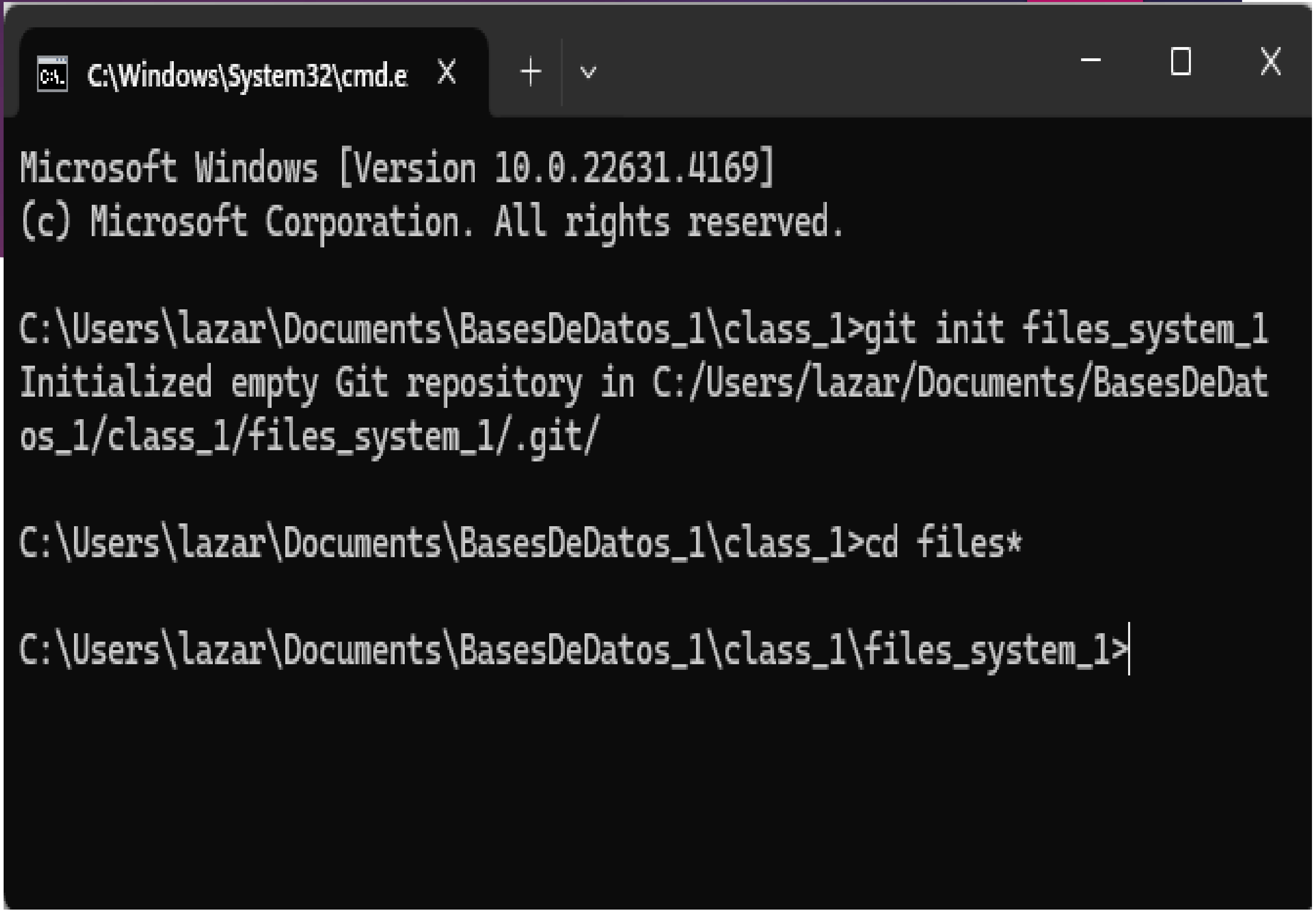


```
C:\Users\lazar\Documents\BasesDeDatos_1\class_1>
git config --global user.name "HRLJ"

C:\Users\lazar\Documents\BasesDeDatos_1\class_1>
git config --global user.email "lazaro.hernandez
@uneatlantico.es"

C:\Users\lazar\Documents\BasesDeDatos_1\class_1>
git config --global color.ui auto

C:\Users\lazar\Documents\BasesDeDatos_1\class_1>
```

Crear repositorio local

Step by Step:

- Crear repositorios
  - git init [project name]
  - git clone [url]
- Efectuar cambios
  - git status
  - git diff
  - git add [file]
  - git diff - -stage
  - git reset - - file
  - git commit -m "[mensaje descriptivo]"



```
Microsoft Windows [Version 10.0.22631.4169]
(c) Microsoft Corporation. All rights reserved.

C:\Users\lazar\Documents\BasesDeDatos_1\class_1>git init files_system_1
Initialized empty Git repository in C:/Users/lazar/Documents/BasesDeDat
os_1/class_1/files_system_1/.git/

C:\Users\lazar\Documents\BasesDeDatos_1\class_1>cd files*

C:\Users\lazar\Documents\BasesDeDatos_1\class_1\files_system_1>
```

Crear repositorio remoto

Configurar git para repositorio remoto

git remote add origin https://github.com/claseBD/files_system_1.git

git remote -v

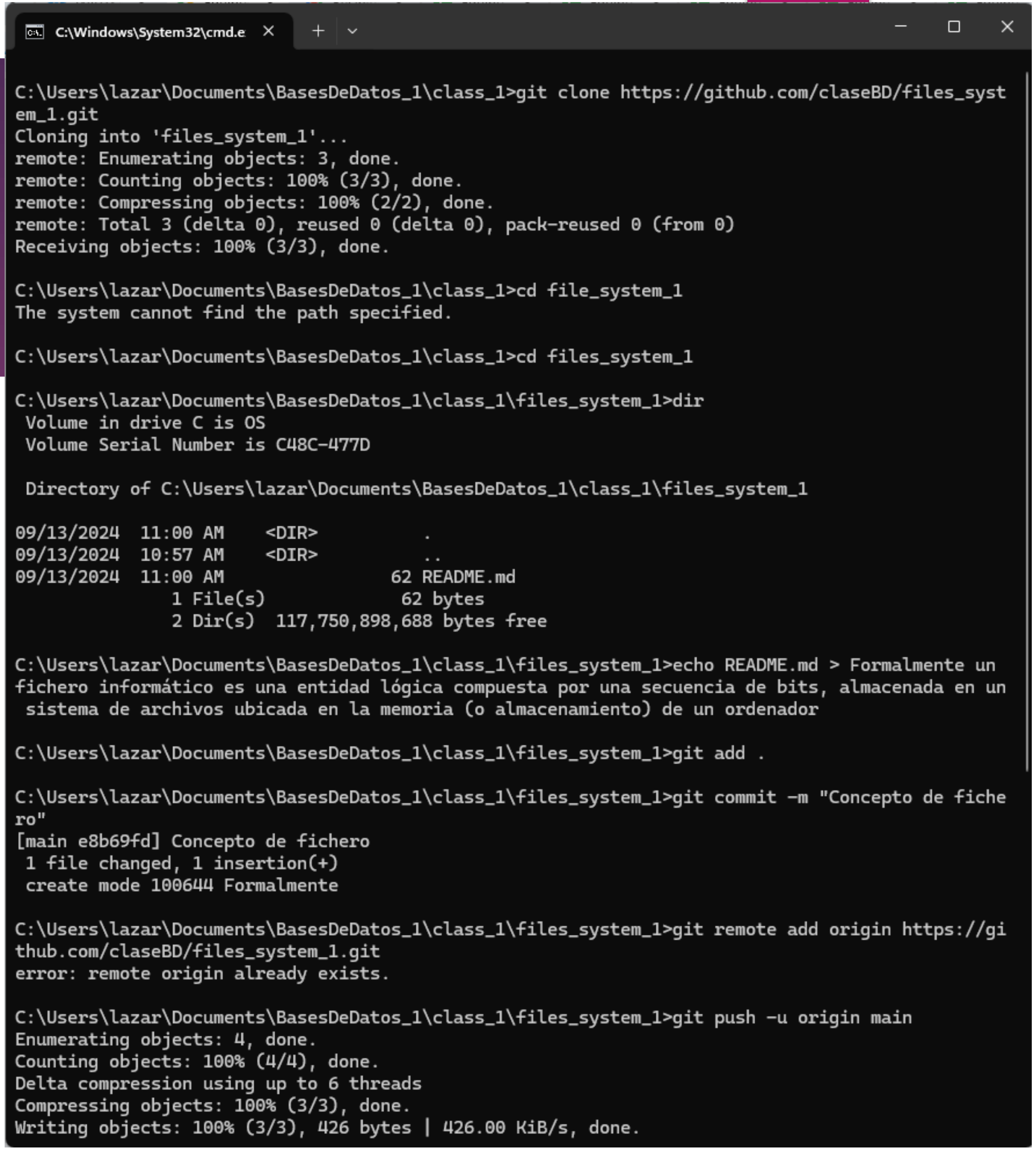
Verificar git remote

origin https://github.com/OWNER/REPOSITORY.git (fetch)

origin

https://github.com/OWNER/RE

POSITORY.git (push)



```
C:\Users\lazar\Documents\BasesDeDatos_1\class_1>git clone https://github.com/claseBD/files_syst
em_1.git
Cloning into 'files_system_1'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.

C:\Users\lazar\Documents\BasesDeDatos_1\class_1>cd file_system_1
The system cannot find the path specified.

C:\Users\lazar\Documents\BasesDeDatos_1\class_1>cd files_system_1

C:\Users\lazar\Documents\BasesDeDatos_1\class_1\files_system_1>dir
 Volume in drive C is OS
 Volume Serial Number is C48C-477D

 Directory of C:\Users\lazar\Documents\BasesDeDatos_1\class_1\files_system_1

09/13/2024  11:00 AM    <DIR>          .
09/13/2024  10:57 AM    <DIR>          ..
09/13/2024  11:00 AM                62 README.md
               1 File(s)             62 bytes
               2 Dir(s)  117,750,898,688 bytes free

C:\Users\lazar\Documents\BasesDeDatos_1\class_1\files_system_1>echo README.md > Formalmente un
fichero informático es una entidad lógica compuesta por una secuencia de bits, almacenada en un
 sistema de archivos ubicada en la memoria (o almacenamiento) de un ordenador

C:\Users\lazar\Documents\BasesDeDatos_1\class_1\files_system_1>git add .

C:\Users\lazar\Documents\BasesDeDatos_1\class_1\files_system_1>git commit -m "Concepto de fiche
ro"
[main e8b69fd] Concepto de fichero
 1 file changed, 1 insertion(+)
 create mode 100644 Formalmente

C:\Users\lazar\Documents\BasesDeDatos_1\class_1\files_system_1>git remote add origin https://gi
thub.com/claseBD/files_system_1.git
error: remote origin already exists.

C:\Users\lazar\Documents\BasesDeDatos_1\class_1\files_system_1>git push -u origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 6 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 426 bytes | 426.00 KiB/s, done.
```

Instalar Jupyter

Step 1: Actualizar e instalar dependencias

```
sudo apt update && sudo apt upgrade -y
sudo apt install python3 python3-pip
python3-venv -y
```

Step 2: Crear un entorno virtual

```
python3  -m venv MiProyecto

cd MiProyecto

source bin/activate
```

Paso 2: Instalar Jupyter

```
pip install notebook
```

Step 3: Ejecución de Jupyter Notebook

```
jupyter notebook
```

GRACIAS