



**UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE
PRIMER PARCIAL**

ESTUDIANTE:

Milton Joseph Yuxi Chicaiza

MATERIA:

PROGRAMACIÓN ORIENTADA A OBJETOS

PARALELO:

NRC-1323

DOCENTE:

Ing. Luis Enrique Jaramillo.

SANGOLQUI – ECUADOR

1. INTRODUCCIÓN

En este informe se dará un seguimiento al proceso realizado para crear un archivo de código básico y subirlo a un repositorio en GitHub utilizando la línea de comandos a través de la consola CMD. Cada paso realizado será descrito de manera clara y sencilla para facilitar el entendimiento y el seguimiento del proceso. Este ejercicio tiene como propósito ayudar a adquirir y fortalecer las habilidades básicas en el uso de Git y GitHub, estas dos herramientas son la clave para gestionar versiones y colaborar eficazmente en proyectos con un grupo de trabajo para el desarrollo de un software en sí.

En este control de lectura 1 nos permite familiarizarnos con el entorno de trabajo en la línea de comandos con una herramienta importante que usan los desarrolladores en sus proyectos. Al aprender a interactuar con Git y GitHub desde la terminal se refuerzan conceptos fundamentales lo que sienta mejor nuestras bases para un manejo eficiente de proyectos colaborativos a futuro.

2. MARCO TEÓRICO

¿QUÉ ES EL CMD?

El CMD (Símbolo del Sistema) es una parte tan integrada en el sistema operativo Windows que en Windows 10 parece estar casi oculto. Sin embargo, esto no implica que Microsoft haya decidido abandonarlo; simplemente ha dado mayor protagonismo a otras funciones de su sistema operativo, como la interfaz gráfica de usuario (GUI), los comandos por voz y las tecnologías de procesamiento de lenguaje natural (Navas, 2023).

¿QUÉ ES UN GIT?

Git es un sistema de control de versiones distribuido, lo que implica que cada clon local de un proyecto actúa como un repositorio completo de control de versiones. Gracias a esto, es posible trabajar de manera eficiente tanto sin conexión como de forma remota, sin depender directamente de un servidor central (Mijacobs, 2023).

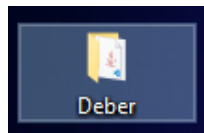
GITHUB

GitHub es una plataforma en la nube que ofrece alojamiento para el sistema de control de versiones Git. Esta herramienta facilita la colaboración entre desarrolladores, permitiéndoles realizar cambios en proyectos compartidos mientras se mantiene un registro detallado del avance y las modificaciones realizadas (B & B, 2023).

3. PROCESO REALIZADO PARA SUBIR ARCHIVOS EN GITHUB

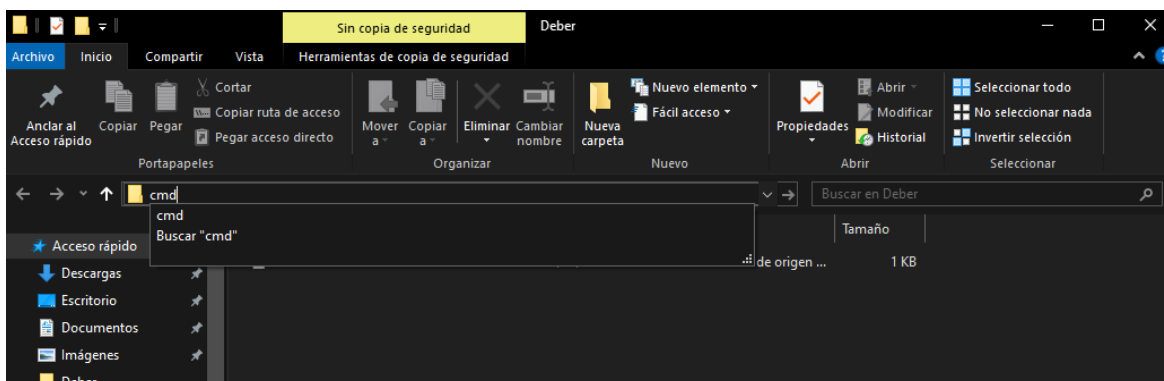
1.1 Creamos una carpeta.

Cambiamos el nombre de la carpeta por un nombre de nuestra elección en este caso “Deber”.



Dentro de esta carpeta será donde subiremos los diferentes archivos al GitHub.

1.2 Accedemos al CMD dentro de la carpeta



1.3 Dentro de la consola ponemos los siguientes comandos:

- `git config --global user.email "@gmail.com"`
- `git config --global user.name "Your user Name"`

```
Selecc...
C:\Windows\System32\cmd.exe
Microsoft Windows [Versión 10.0.19045.5131]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Joseph\Desktop\Deber>git config --global user.email "@gmail.com"

C:\Users\Joseph\Desktop\Deber>git config --global user.name "Your user Name"
```

1.4 EJERCUTAR LOS SIGUIENTES COMANDOS

- `git init`
- `git branch -M main`

Este comando nos dirá si tenemos un repositorio previamente obtenido.

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Versión 10.0.19045.5131]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Joseph\Desktop\Deber>git init
Reinitialized existing Git repository in C:/Users/Joseph/Desktop/Deber/.git/

C:\Users\Joseph\Desktop\Deber>git branch -M main

C:\Users\Joseph\Desktop\Deber>
```

- `git remote add origin https://github.com/MiltonYuxi/ControlLectura1.git`

Con este comando hacemos la conexión de nuestro repositorio con nuestro archivo de la carpeta que creamos anteriormente.

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Versión 10.0.19045.5131]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Joseph\Desktop\Deber>git init
Reinitialized existing Git repository in C:/Users/Joseph/Desktop/Deber/.git/

C:\Users\Joseph\Desktop\Deber>git remote add origin https://github.com/MiltonYuxi/ControlLectura1.git
```

- git status

Para comprobar los archivos que tenemos y que los cuales queremos agregar.

```
C:\Users\Joseph\Desktop\Deber>Git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    DeberPO.java

nothing added to commit but untracked files present (use "git add" to track)
```

- git add .
- git status

Con el primer comando agregamos el archivo a subir y con “git status” verificamos.

```
C:\Users\Joseph\Desktop\Deber>git add .

C:\Users\Joseph\Desktop\Deber>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   DeberPO.java
```

- git commit -m "Control de Lectura 1"

Con esto agregamos una pequeña descripción cuando subimos el archivo

```
C:\Users\Joseph\Desktop\Deber>git commit -m "Control de Lectura 1"
[master (root-commit) a218a20] Control de Lectura 1
1 file changed, 6 insertions(+)
create mode 100644 DeberPO.java
```

- `git push --set-upstream origin master`
- `git push`

Con estos comandos aceptamos y lo guardamos en la plataforma de GitHub.

```
C:\Users\Joseph\Desktop\Deber>git push
fatal: The current branch master has no upstream branch.
To push the current branch and set the remote as upstream, use

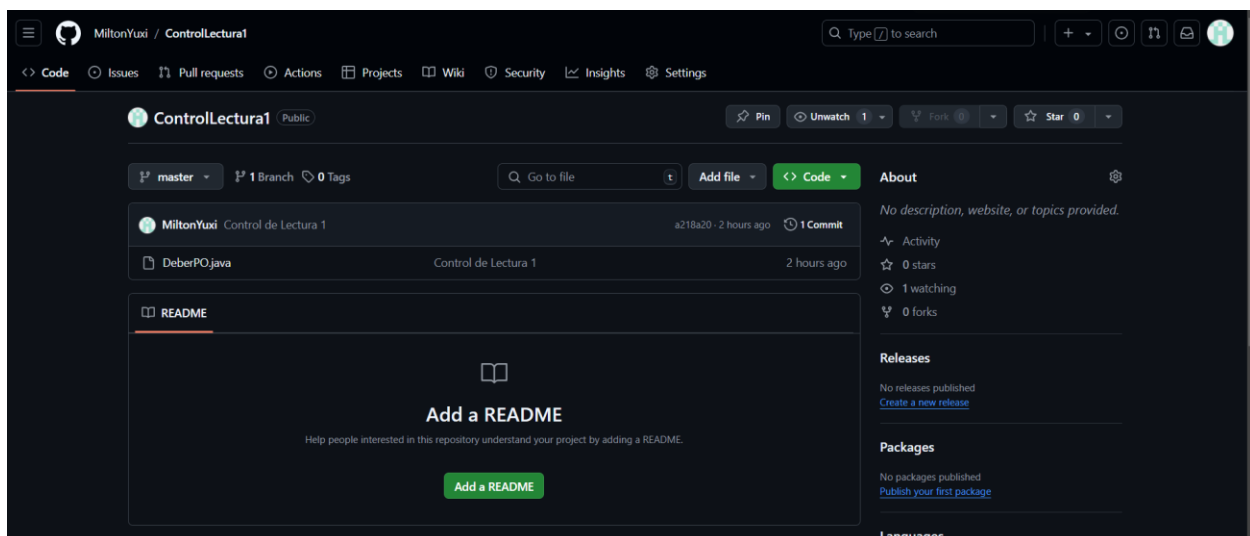
    git push --set-upstream origin master

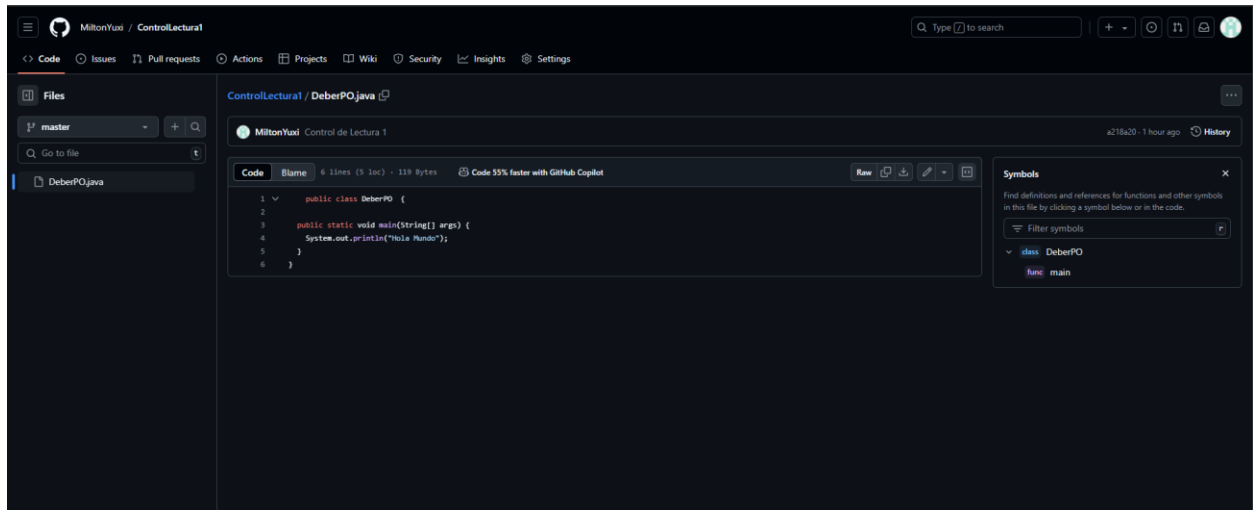
To have this happen automatically for branches without a tracking
upstream, see 'push.autoSetupRemote' in 'git help config'.

C:\Users\Joseph\Desktop\Deber>git push --set-upstream origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 326 bytes | 326.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/MiltonYuxi/ControlLectura1.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

4. RESULTADOS

Los resultados después de la aplicación de la serie de comandos son muy exitosos ya que todos los cambios realizados en el repositorio local se sincronizaron correctamente y se reflejaron de manera automática en la plataforma de GitHub. Esto nos confirma que el proceso fue llevado a cabo de forma correcta desde la inicialización del repositorio local hasta la conexión y subida al repositorio remoto.





5. CONCLUSIÓN

En conclusión, este ejercicio práctico me permitió adquirir y reforzar conocimientos fundamentales sobre el uso de Git y GitHub, desde la creación de un archivo básico hasta su correcta subida a un repositorio. Durante el proceso se comprendió la importancia de las herramientas de control de versiones para la gestión eficiente de proyectos y también para cuando se necesite una colaboración en un equipo.

Así mismo, se llevará a cabo el procedimiento correspondiente para subir este archivo de Word a la plataforma de GitHub, donde estarán disponibles ambos archivos.

6. REFERENCIAS BIBLIOGRÁFICAS

B, G., & B, G. (2023, 10 enero). *Qué es GitHub y cómo empezar a usarlo*. Tutoriales Hostinger.

<https://www.hostinger.es/tutoriales/que-es-github>

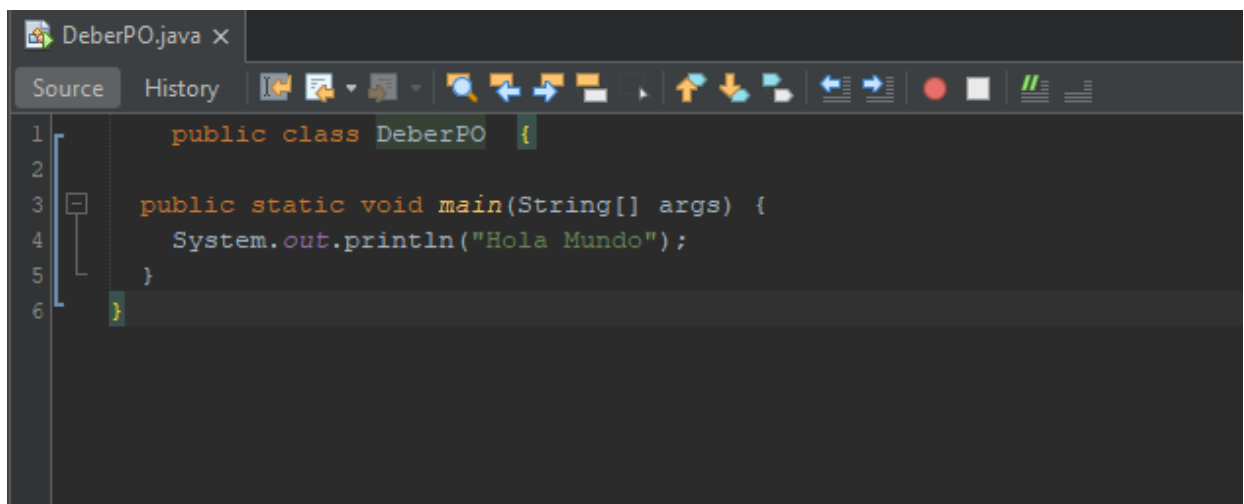
Mijacobs. (2023, 5 octubre). *¿Qué es Git? - Azure DevOps*. Microsoft Learn.

<https://learn.microsoft.com/es-es/devops/develop/git/what-is-git>

Navas, M. Á. (2023, 23 octubre). Que es CMD ¿Que significa y para qué sirve? Profesional Review. <https://www.profesionalreview.com/2018/06/30/que-es-cmd/>

7. ANEXOS

Código “Hola Mundo” previamente realizado en Apache NetBeans IDE 23

A screenshot of the Apache NetBeans IDE interface. The top toolbar shows various icons for file operations, search, and execution. Below the toolbar, the 'Source' tab is active, displaying the code for a Java class named 'DeberPO'. The code is as follows:

```
1 public class DeberPO {  
2  
3     public static void main(String[] args) {  
4         System.out.println("Hola Mundo");  
5     }  
6 }
```