

SERGIO MORENO FERNANDEZ

2ªDAW

Práctica 1 - Uso de Git y GitHub

Diseño de Interfaces Web

Pedro Prieto Alarcón

23/09/2019

Índice

1. Fundamentos Git I	2
2. Fundamentos Git II	3
3. GitHub	5
4. Entrega de la práctica	6

1. Fundamentos GitI

- 1.Instala Git en tu sistema operativo. Haz una captura de pantalla en la que aparezca el resultado de la ejecución del comando `git -version`.

```
moreno20212@SergioMoreno MINGW64 ~  
$ git --version  
git version 2.28.0.windows.1
```

- 2.Realiza la configuración de Git según lo indicado en el tema (nombre, correo electrónico, editor de preferencia y almacenamiento de credenciales). Haz una captura de pantalla con el resultado de la ejecución de los comandos de configuración.

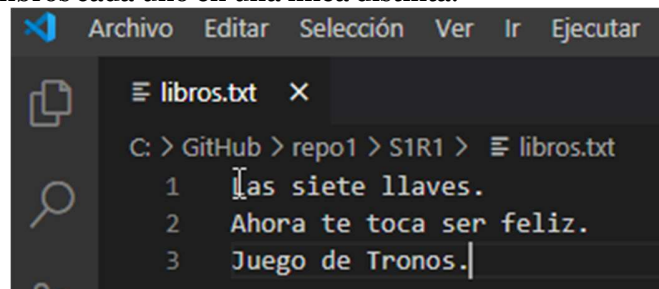
```
moreno20212@SergioMoreno MINGW64 ~  
$ git config --list  
diff.astextplain.textconv=astextplain  
filter.lfs.clean=git-lfs clean -- %f  
filter.lfs.smudge=git-lfs smudge -- %f  
filter.lfs.process=git-lfs filter-process  
filter.lfs.required=true  
http.sslbackend=openssl  
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt  
core.autocrlf=true  
core.fscache=true  
core.symlinks=false  
pull.rebase=false  
credential.helper=manager  
core.editor='C:\User\coneue\AppData\Local\Programs\Microsoft VS\Code.exe' -multiInst -notabbar -  
nosession -noPlugin  
user.name=Sergio Moreno  
user.email=moreno20212@iesmarenostrum.com  
credential.helper=wincred
```

- 3.Crea una carpeta denominada S1R1. Realiza las siguientes acciones en ella:

a)Crea un repositorio Git.

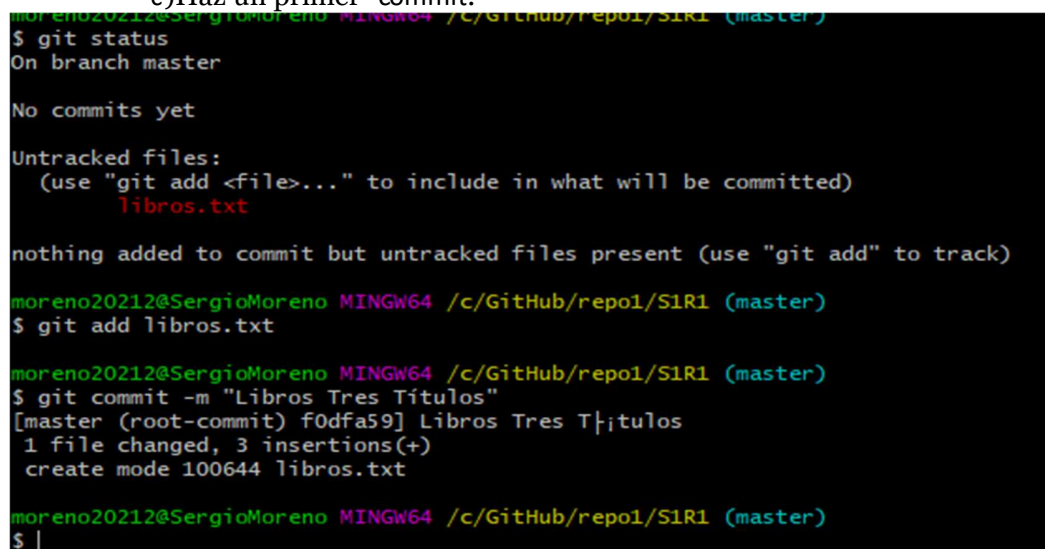
```
moreno20212@SergioMoreno MINGW64 /c/GitHub/repo1/S1R1  
$ git init  
Initialized empty Git repository in C:/GitHub/repo1/S1R1/.git/  
moreno20212@SergioMoreno MINGW64 /c/GitHub/repo1/S1R1 (master)  
$ |
```

- b) Crea un fichero denominado libros.txt. Añade tres títulos de libros cada uno en una línea distinta.



A screenshot of a text editor window titled 'libros.txt'. The editor shows three lines of text: '1 Las siete llaves.', '2 Ahora te toca ser feliz.', and '3 Juego de Tronos.' The cursor is at the end of the third line.

- c) Haz un primer commit.



```
moreno20212@SergioMoreno MINGW64 /c/GitHub/repo1/S1R1 (master)
$ git status
On branch master

No commits yet

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

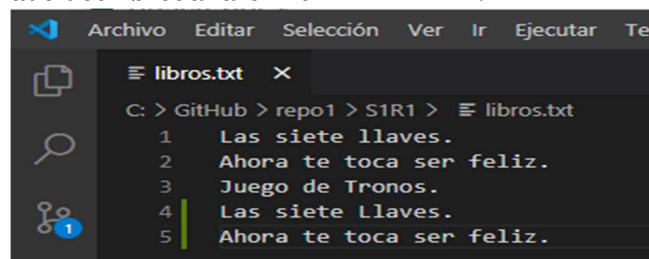
nothing added to commit but untracked files present (use "git add" to track)

moreno20212@SergioMoreno MINGW64 /c/GitHub/repo1/S1R1 (master)
$ git add libros.txt

moreno20212@SergioMoreno MINGW64 /c/GitHub/repo1/S1R1 (master)
$ git commit -m "Libros Tres Títulos"
[master (root-commit) f0dfa59] Libros Tres Títulos
1 file changed, 3 insertions(+)
create mode 100644 libros.txt

moreno20212@SergioMoreno MINGW64 /c/GitHub/repo1/S1R1 (master)
$
```

- d) Añade dos libros al archivo libros.txt.



A screenshot of a text editor window titled 'libros.txt'. The editor shows five lines of text: '1 Las siete llaves.', '2 Ahora te toca ser feliz.', '3 Juego de Tronos.', '4 Las siete llaves.', and '5 Ahora te toca ser feliz.' The cursor is at the end of the fifth line.

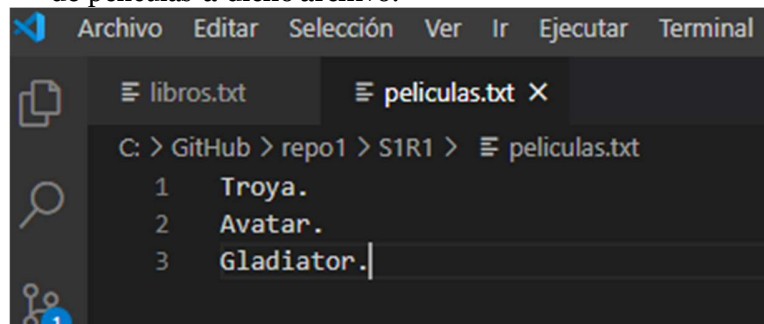
e) Haz un segundo commit.

```
moreno20212@SergioMoreno MINGW64 /c/GitHub/repo1/S1R1 (master)
$ git add libros.txt

moreno20212@SergioMoreno MINGW64 /c/GitHub/repo1/S1R1 (master)
$ git commit -m "Se añaden dos libros mas"
[master 4de43d6] Se añaden dos libros mas
1 file changed, 3 insertions(+), 1 deletion(-)

moreno20212@SergioMoreno MINGW64 /c/GitHub/repo1/S1R1 (master)
$
```

f) Crea un fichero denominado peliculas.txt. Añade tres títulos de películas a dicho archivo.



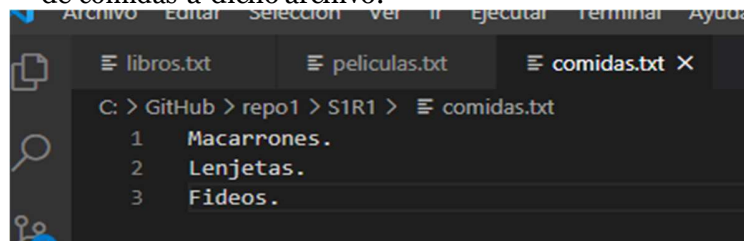
g) Haz una captura de pantalla del comando git status.

```
moreno20212@SergioMoreno MINGW64 /c/GitHub/repo1/S1R1 (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    peliculas.txt

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

moreno20212@SergioMoreno MINGW64 /c/GitHub/repo1/S1R1 (master)
$
```

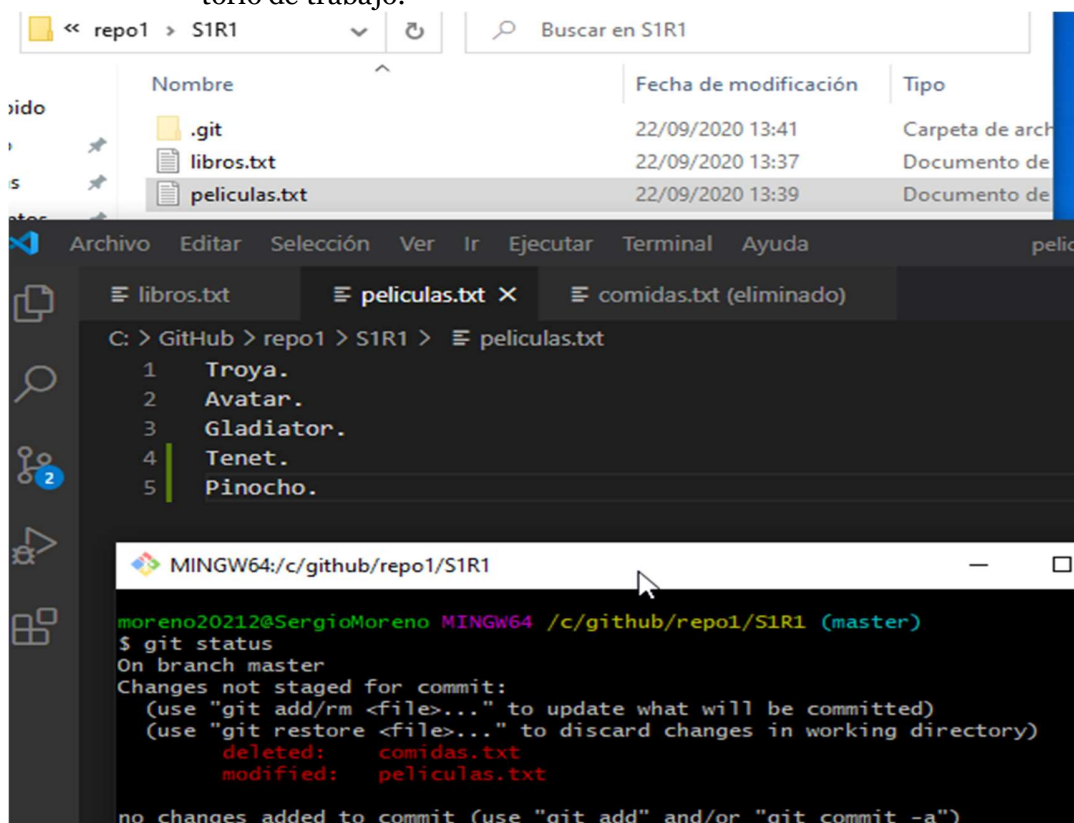
h) Crea un fichero denominado comidas.txt. Añade tres nombres de comidas a dicho archivo.



- i) Haz un tercer commit que incluya los archivos películas.txt y comidas.txt.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/S1R1 (master)
$ git commit -m "Añado Peliculas y comidas"
[master 2e781c0] Añado Peliculas y comidas
2 files changed, 6 insertions(+)
create mode 100644 comidas.txt
create mode 100644 peliculas.txt
```

- j) Elimina el archivo comidas.txt desde el navegador de archivos.
k) Añade dos películas más al archivo películas.txt.
l) Haz una captura de pantalla que muestre los cambios en el directorio de trabajo.



- m) Añade los cambios al área de preparación.

- n) Haz una captura de pantalla del comando `git status`. Debe indicar que se ha borrado el archivo `comidas.txt` y que se ha modificado el archivo `peliculas.txt`.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/S1R1 (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        deleted:    comidas.txt
        modified:   peliculas.txt
```

- ñ) Haz un cuarto commit.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/S1R1 (master)
$ git commit -m "Añados peliculas elimino comidas"
[master 47eee21] Añados peliculas elimino comidas
2 files changed, 3 insertions(+), 4 deletions(-)
delete mode 100644 comidas.txt
```

- o) Crea un archivo denominado `datos.bak`. Añade tres títulos de libros a dicho archivo.
- p) Crea una subcarpeta denominada `output`. Crea un archivo denominado `salida.txt` en su interior. Escribe tu nombre y apellidos en dicho archivo.
- q) Haz una captura de pantalla del comando `git status`. Deben aparecer los archivos `datos.bak` y `output/salida.txt` como archivos nuevos.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/S1R1 (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        datos.bak
        output/
```

- r) Crea un archivo `.gitignore` para que los ficheros con extensión `.bak` y el contenido de la carpeta `output/` no se incluyan en el repositorio.

```
C: > GitHub > repo1 > S1R1 > .gitignore
1 *.bak
2 /output
```


- s) Haz una nueva captura de pantalla del comando `git status`. Ahora no deben aparecer los archivos `datos.bak` y `output/salida.txt` como archivos nuevos, sino que en su lugar debe aparecer únicamente el archivo `.gitignore`.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/S1R1 (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .gitignore
```

- t) Haz un último commit para incluir el archivo `.gitignore` en el repositorio.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/S1R1 (master)
$ git commit -m "añado gitignore"
[master ee693aa] a[redacted]ado gitignore
1 file changed, 2 insertions(+)
create mode 100644 .gitignore
```

- u) Haz una captura de pantalla que muestre el histórico de cambios del repositorio.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/S1R1 (master)
$ git log
commit ee693aa27fe3d2fcd6d7751a634337c291e08383 (HEAD -> master)
Author: Sergio Moreno <moreno20212@iesmarenosttrum.com>
Date: Tue Sep 22 14:06:12 2020 +0200

    añadido gitignore

commit 47eee21838a76fa63fcb5204553306bbc1e97d30
Author: Sergio Moreno <moreno20212@iesmarenosttrum.com>
Date: Tue Sep 22 13:49:07 2020 +0200

    Añados películas elimino comidas

commit 2e781c06698fee54db2f05db9a9f315621548c8c
Author: Sergio Moreno <moreno20212@iesmarenosttrum.com>
Date: Tue Sep 22 13:41:59 2020 +0200

    Añado Películas y comidas

commit 55a3984c03fa2fa6d4a641790fff7b2f57b74e7d
Author: Sergio Moreno <moreno20212@iesmarenosttrum.com>
Date: Tue Sep 22 13:37:34 2020 +0200

    Libros cinco títulos

commit 84c491380b7e68aa0d9e62e0fb5804c55be12089
Author: Sergio Moreno <moreno20212@iesmarenosttrum.com>
Date: Tue Sep 22 13:36:38 2020 +0200

    Libros tres títulos

moreno20212@SergioMoreno MINGW64 /c/github/repo1/S1R1 (master)
$ |
```

2. Fundamentos GitII

1. Crea una carpeta denominada S2R1. Realiza las siguientes acciones en ella:

a) Crea un repositorio Git.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/s2r1 (master)
$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
```

b) Crea un fichero denominado actores.txt. Añade tres nombres de actores cada uno en una línea distinta.

c) Haz un primer commit.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/s2r1 (master)
$ git commit -m "Añado tres actores"
[master (root-commit) f6230b8] Añado tres actores
1 file changed, 3 insertions(+)
create mode 100644 actores.txt
```

d) Crea una rama denominada test.

e) En la rama test crea un fichero denominado actrices.txt. Añade tres nombres de actrices y realiza un commit en dicha rama.

f) Haz una captura de pantalla del resultado del comando `git log --graph -all`.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/s2r1 (test)
$ git log --graph --all
* commit 4bc8debb4a795d0276c6f5d0e1f2c36011afa5f7 (HEAD -> test)
  Author: Sergio Moreno <moreno20212@iesmarenostrum.com>
  Date: Tue Sep 22 14:52:33 2020 +0200

  Creo Actrices rama test

* commit f6230b879ddf29d0dbd36938bc33d4a7f8bce103 (master)
  Author: Sergio Moreno <moreno20212@iesmarenostrum.com>
  Date: Tue Sep 22 14:43:59 2020 +0200

  Añado tres actores
```

g) Incorpora los cambios de la rama test a la rama master. Haz una captura de pantalla de los comandos que has utilizado y de su resultado.


```

moreno20212@SergioMoreno MINGW64 /c/github/repo1/s2r1 (master)
$ git merge test
Updating dab10d4..88cbc3b
Fast-forward
  actrices.txt | 3 +++
  1 file changed, 3 insertions(+)
  create mode 100644 actrices.txt

```

- h) Posiciónate en la rama master.
- i) Crea una segunda rama denominada test2.
- j) En la rama master, añade una actriz al fichero actrices. txt.
- k) En la rama test2, añade una actriz al fichero actrices. txt.
- l) Haz una captura de pantalla del resultado del comando `git log --graph --all`. Debe haber dos caminos distintos: uno para la rama master y otro para la rama test2.

```

* commit 0873bc9e5fde975d0d487bb13a8f4d437ea5bd00 (HEAD -> test2)
| Author: Sergio Moreno <moreno20212@iesmarenosttrum.com>
| Date: Tue Sep 22 15:39:23 2020 +0200
|
| añadir Julia
|
| * commit 8f8bed9f50a35ac84a3dbf7742af16b9094f03fa (master)
| / Author: Sergio Moreno <moreno20212@iesmarenosttrum.com>
|   Date: Tue Sep 22 15:38:24 2020 +0200
|
|   añadir Lorena
|
| * commit 88cbc3ba17b508e5d649602d29253d11b2a66c1d (test)
| | Author: Sergio Moreno <moreno20212@iesmarenosttrum.com>
| | Date: Tue Sep 22 15:29:09 2020 +0200
| |
| | Añado actrices a test
| |
| * commit dab10d4dc1d365ff8519925241594b892b1ba0fd
| | Author: Sergio Moreno <moreno20212@iesmarenosttrum.com>
| | Date: Tue Sep 22 15:25:34 2020 +0200
| |
| | Añado tres actores

```

- m) Incorpora los cambios de la rama test2 a la rama master. ¿Se produce un conflicto? De ser así realiza una captura del comando `git status`.

```

moreno20212@SergioMoreno MINGW64 /c/github/repo1/s2r1 (master)
$ git merge test2
Auto-merging actrices.txt
CONFLICT (content): Merge conflict in actrices.txt
Automatic merge failed; fix conflicts and then commit the result.

moreno20212@SergioMoreno MINGW64 /c/github/repo1/s2r1 (master|MERGING)
$ git status
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)
    both modified:   actrices.txt

no changes added to commit (use "git add" and/or "git commit -a")

```

n) Resuelve el conflicto incorporando los dos nombres de actrices.

```
actrices.txt: Bloc de notas
Archivo  Edición  Formato  Ver  Ayuda
Penelope.
Paz.
Evelin.
<<<<<<< HEAD
Lorena.
=====
Julia.
>>>>>>> test2
```

ñ) Haz una captura de pantalla del resultado del comando `git log --graph --all`. Observa que se ha creado un nuevo commit que integra los dos caminos anteriores.

```
* commit 2720d736501c4d751efffa678501e5f884922fcc (HEAD -> master)
Merge: 8f8bed9 0873bc9
Author: Sergio Moreno <moreno20212@iesmarenosttrum.com>
Date: Tue Sep 22 16:10:06 2020 +0200

    resolucion de conflicto

* commit 0873bc9e5fde975d0d487bb13a8f4d437ea5bd00 (test2)
Author: Sergio Moreno <moreno20212@iesmarenosttrum.com>
Date: Tue Sep 22 15:39:23 2020 +0200

    añadir Julia

* commit 8f8bed9f50a35ac84a3dbf7742af16b9094f03fa
Author: Sergio Moreno <moreno20212@iesmarenosttrum.com>
Date: Tue Sep 22 15:38:24 2020 +0200

    añadir Lorena

* commit 88cbc3ba17b508e5d649602d29253d11b2a66c1d (test)
Author: Sergio Moreno <moreno20212@iesmarenosttrum.com>
Date: Tue Sep 22 15:29:09 2020 +0200

    Añado actrices a test

* commit dab10d4dc1d365ff8519925241594b892b1ba0fd
Author: Sergio Moreno <moreno20212@iesmarenosttrum.com>
Date: Tue Sep 22 15:25:34 2020 +0200

    Añado tres actores
```

2. Crea una carpeta denominada S2R2-remoto. Inicializa un repositorio Git en su interior mediante el comando `git init --bare`. Esta carpeta se utilizará como repositorio remoto.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/S2R2-remoto
$ git init --bare
Initialized empty Git repository in C:/GitHub/repo1/S2R2-remoto/
```

3. Clona el repositorio S2R2-remoto en una carpeta denominada S2R2. Haz una captura de pantalla del resultado del comando de clonado. A continuación, realiza los siguientes pasos en el repositorio S2R2.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1
$ git clone S2R2-remoto S2R2
Cloning into 'S2R2'...
warning: You appear to have cloned an empty repository.
done.
```

- a) Crea un archivo denominado `directores.txt`. Añade el nombre de tres directores de cine.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/S2R2 (master)
$ git commit -m "Creo directores"
[master (root-commit) ed1b27a] Creo directores
1 file changed, 3 insertions(+)
create mode 100644 directores.txt
```

- b) Haz un commit.

- c) Realiza un push al repositorio remoto. Haz una captura de pantalla del resultado.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/S2R2 (master)
$ git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 249 bytes | 6.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To C:/GitHub/repo1/S2R2-remoto
* [new branch]      master -> master
```

- d) Crea una rama denominada `version1`.
- e) En la rama `version1` añade el nombre de dos directores de cine más al archivo `directores.txt`.

- f) Realiza un push de la rama al repositorio remoto de manera que quede asociada a la rama remota del mismo nombre. Haz una captura de pantalla del resultado.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/S2R2 (version1)
$ git push origin version1
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 314 bytes | 78.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To C:/GitHub/repo1/S2R2-remoto
* [new branch]      version1 -> version1
```

4. Clona el repositorio S2R2-remoto en una segunda carpeta denominada S2R3. Realiza las siguientes acciones sobre ella:

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1
$ git clone S2R2-remoto S2R3
Cloning into 'S2R3'...
done.
```

- a) Muestra en la consola el contenido del fichero directores.txt y el resultado del comando git status.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/S2R3 (master)
$ cat directores.txt
Luis.
Juan.
Sergio.
moreno20212@SergioMoreno MINGW64 /c/github/repo1/s2r3 (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
```

- b) Cambia a la rama version1. Muestra el resultado del comando.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/s2r3 (version1)
$ git status
On branch version1
Your branch is up to date with 'origin/version1'.

nothing to commit, working tree clean
```

- c) Muestra el contenido del fichero directores.txt por la pantalla. Comprueba que se muestran los 5 nombres de directores esperados. Haz una captura de pantalla.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/s2r3 (version1)
$ cat directores.txt
Luis.
Juan.
Sergio.
Messi.
Fernando.
```

- d) Incorpora los cambios de la rama version1 a la rama master.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/s2r3 (version1)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

moreno20212@SergioMoreno MINGW64 /c/github/repo1/s2r3 (master)
$ git merge version1
Updating ed1b27a..b31ee8b
Fast-forward
 directores.txt | 4 +++-
 1 file changed, 3 insertions(+), 1 deletion(-)
```

- e) Sube la rama master actualizada al servidor. Haz una captura de pantalla del resultado del comando.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/s2r3 (master)
$ git push
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To C:/GitHub/repo1/S2R2-remoto
 ed1b27a..b31ee8b master -> master
```

5. Vuelve de nuevo a la carpeta S2R2 y realiza las siguientes acciones:

- a) Obtén los cambios que hay en el repositorio remoto. Haz una captura de pantalla del resultado del comando utilizado.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/s2r2 (master)
$ git fetch
From C:/GitHub/repo1/S2R2-remoto
 ed1b27a..b31ee8b master -> origin/master
```

- b) Actualiza la rama master local con el contenido de la rama master del repositorio remoto. Haz una captura de pantalla del resultado del comando utilizado.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/s2r2 (master)
$ git pull
Updating ed1b27a..b31ee8b
Fast-forward
 directores.txt | 4 +++-
 1 file changed, 3 insertions(+), 1 deletion(-)
```


- c) Comprueba que deben aparecer los 5 nombres de directores esperados.

```
moreno20212@SergioMoreno MINGW64 /c/github/repo1/s2r2 (master)
$ cat directores.txt
Luis.
Juan.
Sergio.
Messi.
Fernando.
```

3. GitHub

1. Crea una cuenta en GitHub
2. Añade tu dirección de correo de educación
3. Solicita un descuento para uso educativo en https://education.github.com/discount_requests/new
4. Haz un fork del repositorio localizado en la siguiente url: <https://github.com/interfacesweb-19-20/semana1-github>

```
moreno20212@SergioMoreno MINGW64 /c/github
$ git clone https://github.com/smf1920/semana1-github.git
Cloning into 'semana1-github'...
remote: Enumerating objects: 7, done.
remote: Total 7 (delta 0), reused 0 (delta 0), pack-reused 7
Unpacking objects: 100% (7/7), 1.23 KiB | 5.00 KiB/s, done.
```

- a) Realiza un primer commit para poner tu nombre y apellidos en el fichero README.md

```
moreno20212@SergioMoreno MINGW64 /c/github/semana1-github (master)
$ dir
discos.txt  libros.txt  LICENSE  peliculas.txt  README.md

moreno20212@SergioMoreno MINGW64 /c/github/semana1-github (master)
$ cat README.MD
# Diseño de Interfaces Web
Repositorio para la práctica de uso de GitHub

Incluye a continuación tu nombre y apellidos:

moreno20212@SergioMoreno MINGW64 /c/github/semana1-github (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")

moreno20212@SergioMoreno MINGW64 /c/github/semana1-github (master)
$ git add .

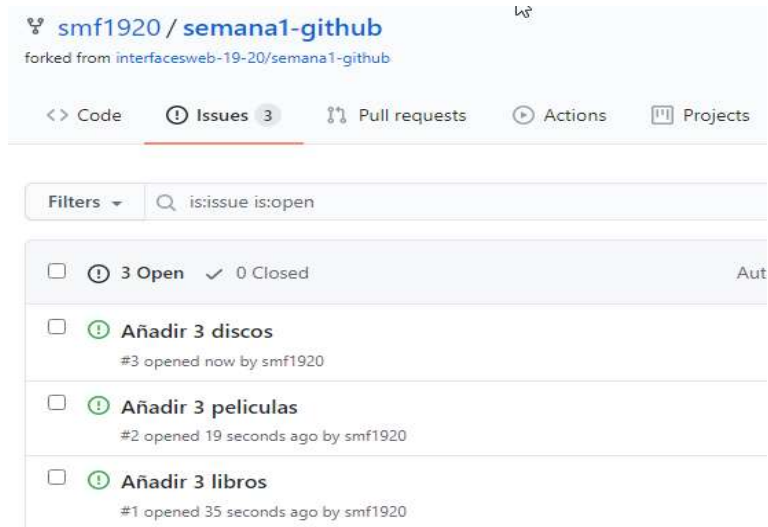
moreno20212@SergioMoreno MINGW64 /c/github/semana1-github (master)
$ git commit -m "mi nombre en readme"
[master e359bc2] mi nombre en readme
1 file changed, 1 insertion(+), 1 deletion(-)
```


b) Crea 3 issues con los siguientes títulos:

Añadir 3 libros

Añadir 3 películas

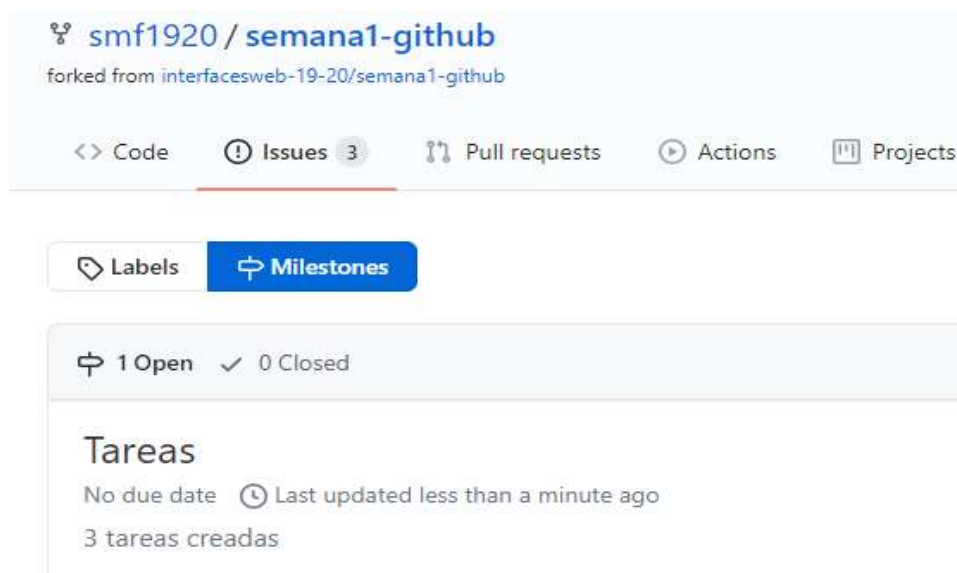
Añadir 3 discos



The screenshot shows the GitHub repository page for 'smf1920 / semana1-github', which is a fork of 'interfacesweb-19-20/semana1-github'. The 'Issues' tab is selected, showing 3 open issues. The issues are listed in descending order of creation time:

- ☐ **Añadir 3 discos** (Issue #3)
opened now by smf1920
- ☐ **Añadir 3 peliculas** (Issue #2)
opened 19 seconds ago by smf1920
- ☐ **Añadir 3 libros** (Issue #1)
opened 35 seconds ago by smf1920

c) Crea una milestone denominada Tareas que contenga los 3 issues creados.



The screenshot shows the same GitHub repository page, but now the 'Milestones' tab is selected. A new milestone named 'Tareas' has been created and is currently the only open milestone. The milestone details are as follows:

- Tareas**
- No due date
- Last updated less than a minute ago
- 3 tareas creadas

- d) Modifica los ficheros correspondientes y realiza 3 commits para realizar cada una de las tareas que se indican en los issues. El mensaje del commit debe hacer que se cierren los issues correspondientes de manera automática.

```
moreno20212@SergioMoreno MINGW64 /c/github/semana1-github (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   discos.txt

no changes added to commit (use "git add" and/or "git commit -a")

moreno20212@SergioMoreno MINGW64 /c/github/semana1-github (master)
$ git add .

moreno20212@SergioMoreno MINGW64 /c/github/semana1-github (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   discos.txt

moreno20212@SergioMoreno MINGW64 /c/github/semana1-github (master)
$ git commit -m "se añaden 3 discos. close #3 "
[master 7e72936] se añaden 3 discos. close #3
1 file changed, 3 insertions(+)

moreno20212@SergioMoreno MINGW64 /c/github/semana1-github (master)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 334 bytes | 334.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/smf1920/semana1-github.git
   e359bc2..7e72936  master -> master
```

Los mismo con libros y películas.

- e) Haz una captura de pantalla de los comandos que has utilizado para hacer los commits y subir los cambios a GitHub.

- f) Incluye todas las capturas de pantalla en el repositorio dentro de la carpeta `capturas`. Añádelas también al repositorio de manera que queden guardadas en tu repositorio en GitHub.
- g) Realiza una `pull request` indicando en el mensaje que has completado la tarea.

4. Entrega de la práctica

La entrega de la tarea se realiza a través de la `pull request` en GitHub. No olvides incluir en la carpeta `capturas` todas las capturas de pantalla realizadas en la práctica (Fundamentos de Git I, Fundamentos de Git II y GitHub).