

Git Extra

A.D.G.

<https://github.com/Andrei-DG/GitExtra>

Haz las dos siguiente partes por linea de comandos.

1. Crea un repositorio en local y luego subelo a github.

Abro la consola con **CTRL + Ñ**, y creo algunas carpetas, archivos y texto de prueba para subir con cmd luego:

Creamos la carpeta con nombre (temporal, luego lo cambié) RepositorioExtra donde estará el repositorio local:

```
PS C:\Users\Andrei> mkdir RepositorioExtra
```

Creo dentro otra carpeta:

```
PS C:\Users\Andrei\RepositorioExtra> mkdir "Una carpeta cualquiera"
```

Me sitúo en la carpeta creada antes:

```
PS C:\Users\Andrei\RepositorioExtra> cd "Una carpeta cualquiera"
```

Añado dos archivos de texto con una prueba de texto:

```
PS C:\Users\Andrei\RepositorioExtra\Una carpeta cualquiera> echo "Esto es para probar añadir un texto a un txt" > "Archivonuevo.txt"
PS C:\Users\Andrei\RepositorioExtra\Una carpeta cualquiera> echo "Más y más texto en otro archivo" > "Otroarchivo.txt"
```

Verifico el texto añadido dentro de un archivo texto:

```
PS C:\Users\Andrei\RepositorioExtra\Una carpeta cualquiera> cat Archivonuevo.txt
Esto es para probar añadir un texto a un txt
```

Me sitúo en una próxima ruta anterior:

```
PS C:\Users\Andrei\RepositorioExtra\Una carpeta cualquiera> cd ..
```

Creo otra carpeta:

```
PS C:\Users\Andrei\RepositorioExtra> mkdir Carpeta2
```

Añado texto a otro archivo más, esta vez un README.md:

```
PS C:\Users\Andrei\RepositorioExtra> echo "Este es el README que debería ser más bonito pero como esta primera parte es solo una prueba de subir un repositorio cualquiera con comandos" > "README.md"
```

Leo el archivo:

```
PS C:\Users\Andrei\RepositorioExtra> cat README.md
Este es el README que debería ser más bonito pero como esta primera parte es solo una prueba de subir un repositorio cualquiera con comandos
```

Verifico lo que tengo creado en el Repositorio de momento:

```
PS C:\Users\Andrei\RepositorioExtra> ls

Directorio: C:\Users\Andrei\RepositorioExtra

Mode                LastWriteTime         Length Name
----                -
d-----          28/09/2022   18:30             Carpeta2
d-----          28/09/2022   18:11      Una carpeta cualquiera
-a----          28/09/2022   18:46             286 README.md
```

Subida del repositorio

Se crea un nuevo repositorio de Git en local:

```
PS C:\Users\Andrei\RepositorioExtra> git init
Reinitialized existing Git repository in C:/Users/Andrei/RepositorioExtra/.git/
```

Se añade en seguimiento de cambios de todo lo actual, tanto de carpetas como de archivos existentes en ese momento:

```
PS C:\Users\Andrei\RepositorioExtra> git add .
```

Hacemos un commit para guardar los cambios que tenemos hasta aquí en local:

```
PS C:\Users\Andrei\RepositorioExtra> git commit -m "Primera vez que subo GitExtra"
[master (root-commit) b7add40] Primera vez que subo GitExtra
3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.md
create mode 100644 Una carpeta cualquiera/Archivonuevo.txt
create mode 100644 Una carpeta cualquiera/Otroarchivo.txt
```

Descargamos Github CLI, instalamos, entramos en la consola, y al usar el comando de crear repositorio nos comenta que tenemos que iniciar sesión para empezar a usar GitHub CLI, usamos su comando y seguimos las preguntas que nos va indicando contestando con las flechas y enter para aceptar opción.

```
C:\Users\Andrei\RepositorioExtra>gh repo create
To get started with GitHub CLI, please run:  gh auth login
Alternatively, populate the GH_TOKEN environment variable with a GitHub API authentication token.

C:\Users\Andrei\RepositorioExtra>gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? [Use arrows to move, type to filter]
  Login with a web browser
> Paste an authentication token
```

Como elegí iniciar sesión por Token (por curiosidad 😊), se me ha generado esta:

```
✓ ghp_qmTEZ2oKDNFmE7IggNby3JG7Ev[REDACTED]Xapgt
```

```
? Paste your authentication token: *****
- gh config set -h github.com git_protocol https
[✓] Configured git protocol
[✓] Logged in as Andrei-DG

C:\Users\Andrei\RepositorioExtra>
```

```
C:\Users\Andrei\RepositorioExtra>gh repo create
? What would you like to do? Push an existing local repository to GitHub
? Path to local repository (.) .

? Path to local repository .
? Repository name (RepositorioExtra) GitExtra

? Repository name GitExtra
? Description (RepositorioExtra)

? Description RepositorioExtra
? Visibility Private
[✓] Created repository Andrei-DG/GitExtra on GitHub
? Add a remote? No

C:\Users\Andrei\RepositorioExtra>
```


Conectamos la carpeta local con Github (el repositorio lo hemos creado ya con GitHub CLI, su comando GH):

```
PS C:\Users\Andrei\RepositorioExtra> git remote add origin https://github.com/Andrei-DG/GitExtra.git
```

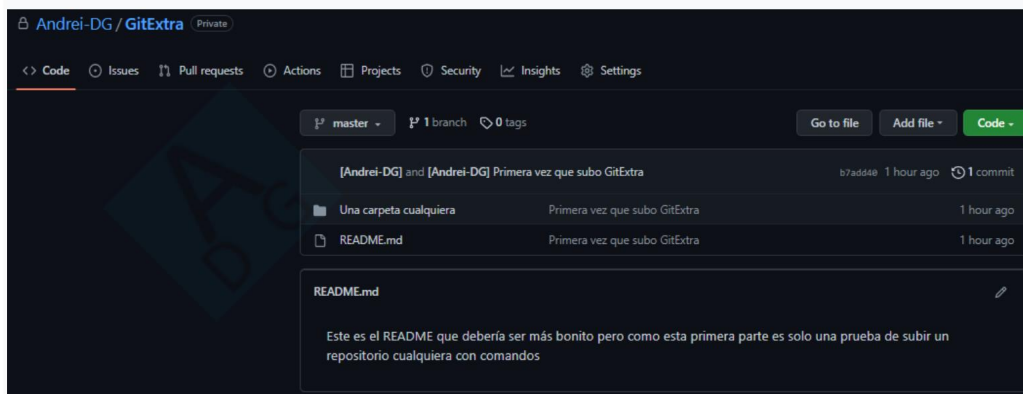
Esto es una confirmación entre remoto y local asegurándose de que el contenido sea el mismo:

```
PS C:\Users\Andrei\RepositorioExtra> git remote -v
origin https://github.com/Andrei-DG/GitExtra.git (fetch)
origin https://github.com/Andrei-DG/GitExtra.git (push)
```

Subimos el contenido al remoto de Github con nombre cambiado a GitExtra:

```
PS C:\Users\Andrei\RepositorioExtra>
PS C:\Users\Andrei\RepositorioExtra> git push -u origin master
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 649 bytes | 649.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Andrei-DG/GitExtra.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
PS C:\Users\Andrei\RepositorioExtra>
```

Y ya lo podemos ver en nuestro Github:



2. Hacer la primera practica de git basico pero con gitlab.

Añadimos carpeta del repositorio GitBasico2ASIR a mi escritorio sin el .git, inicio consola GIT Bash, me sitúo en la carpeta del repositorio, **git init** para inicializar el repositorio creando la carpeta oculta .git, **git add .** para poner en seguimiento todo el contenido existente ahora mismo. Un primer **commit** del estado actual del repositorio. Se añade el origen con **git remote** del GitLab. El **remote -v** para verificar orígenes. **Git push** para subir a GitLab, pedirá login la primera vez.

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop/GitExtra2
$ git init
Initialized empty Git repository in C:/Users/Andrei/Desktop/GitExtra2/.git/

Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop/GitExtra2 (master)
$ git add .

Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop/GitExtra2 (master)
$ git commit -m "Primer commit de GitLab"
[master (root-commit) d649cfe] Primer commit de GitLab
10 files changed, 240 insertions(+)
create mode 100644 A.D.G.-Logo.png
create mode 100644 "Administraci303263n de sistemas gestores de bases de dato
s/README.md"
create mode 100644 "Administraci303263n de sistemas operativos/README.md"
create mode 100644 Empresa e iniciativa emprendedora/README.md
create mode 100644 EncabezadoASIR.png
create mode 100644 "Implantaci303263n de aplicaciones web/README.md"
create mode 100644 "Ingl303251s/README.md"
create mode 100644 README.md
create mode 100644 Seguridad y alta disponibilidad/README.md
create mode 100644 Servicios en red/README.md

Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop/GitExtra2 (master)
$ git remote add origin https://gitlab.com/2asir1/GitExtra2

Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop/GitExtra2 (master)
$ git remote -v
origin https://gitlab.com/2asir1/GitExtra2 (fetch)
origin https://gitlab.com/2asir1/GitExtra2 (push)

Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop/GitExtra2 (master)
$ git push -u origin --all
```

Connect to GitLab



Sign in

[Browser](#) [Token](#) [Password](#)

[Sign in with your browser](#)

Don't have an account? [Sign Up](#)

```

Andrei@DESKTOP-VV322F MINGW64 ~/Desktop/GitExtra2 (master)
$ git push -u origin --all
warning: redirecting to https://gitlab.com/2asir1/GitExtra2.git/
Enumerating objects: 19, done.
Counting objects: 100% (19/19), done.
Delta compression using up to 8 threads
Compressing objects: 100% (12/12), done.
Writing objects: 100% (19/19), 32.03 KiB | 10.68 MiB/s, done.
Total 19 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: To create a merge request for master, visit:
remote: https://gitlab.com/2asir1/GitExtra2/-/merge_requests/new?merge_request%5Bsource_branch%5D=master
remote:
To https://gitlab.com/2asir1/GitExtra2
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.

Andrei@DESKTOP-VV322F MINGW64 ~/Desktop/GitExtra2 (master)
$ |

```

Ya tendría listo mi repositorio subido a GitLab 😊:

2ASIR > GitExtra2 > Repository

You pushed to **master** 2 minutes ago
 ×

Create merge request

master
 GitExtra2 /
 +
 History
 Find file
 Web IDE
 Clone

Primer commit de GitLab
 [Andrei-DG] authored 7 minutes ago
 d649cfed

Name	Last commit	Last update
Administración de sistemas gest...	Primer commit de GitLab	7 minutes ago
Administración de sistemas oper...	Primer commit de GitLab	7 minutes ago
Empresa e iniciativa emprended...	Primer commit de GitLab	7 minutes ago
Implantación de aplicaciones web	Primer commit de GitLab	7 minutes ago
Inglés	Primer commit de GitLab	7 minutes ago
Seguridad y alta disponibilidad	Primer commit de GitLab	7 minutes ago
Servicios en red	Primer commit de GitLab	7 minutes ago
A.D.G.-Logo.png	Primer commit de GitLab	7 minutes ago
EncabezadoASIR.png	Primer commit de GitLab	7 minutes ago
README.md	Primer commit de GitLab	7 minutes ago

README.md

GRADO SUPERIOR IFCS01

Técnico Superior en Administración de Sistemas Informáticos en Red

Este profesional será capaz de:

Configurar, administrar y mantener sistemas informáticos, garantizando la funcionalidad, la integridad de los recursos y servicios del

3. Investiga que es git flow y haz uso de él. Cada feature nueva simulala con un fichero md nuevo.

Crear la carpeta de GitFlow

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop
$ mkdir GitFlow
```

Iniciar repositorio:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop
$ git init
Initialized empty Git repository in C:/Users/Andrei/Desktop/.git/
```

Crear las branches para diferenciar ramas, de los nombres de las ramas por defecto:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (master)
$ git flow init
No branches exist yet. Base branches must be created now.
Branch name for production releases: [master]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [C:/Users/Andrei/Desktop/.git/hooks]
```

Creemos una nueva feature y nos sitúa en esa rama en esa Revision1:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (develop)
$ git flow feature start 1Revision
Switched to a new branch 'feature/1Revision'

Summary of actions:
- A new branch 'feature/1Revision' was created, based on 'develop'
- You are now on branch 'feature/1Revision'

Now, start committing on your feature. When done, use:

    git flow feature finish 1Revision
```

Podemos verificar que se nos haya creado y en cuál estamos ahora:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/1Revision)
$ git branch
  develop
* feature/1Revision
  master
```

Vemos las revisiones que tenemos creadas en la rama con:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/1Revision)
$ git flow feature list
* 1Revision
```

Simulamos la feature con un archivo md:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/1Revision)
$ touch fichero.md
```


Podemos ver los archivos que están en seguimiento, fichero.md no está tracked (hay muchas cosas en escritorio):

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/1Revision)
$ git status
```

Lo añadimos con git add:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/1Revision)
$ git add fichero.md
```

Y ya aparecerá arriba en git status nuestro cambio al que debemos hacer commit.

```
$ git status
On branch feature/1Revision
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   fichero.md
```

Hacemos el commit:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/1Revision)
$ git commit -m "añadido fichero.md"
[feature/1Revision ae736d5] añadido fichero.md
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 fichero.md
```

Aquí puedo comprobar mis commits:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/1Revision)
$ git log --oneline
ae736d5 (HEAD -> feature/1Revision) añadido fichero.md
fc72b1e (master, develop) Initial commit
```

Creo nueva rama para la segunda revisión:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/1Revision)
$ git flow feature start 2Revision
Switched to a new branch 'feature/2Revision'

Summary of actions:
- A new branch 'feature/2Revision' was created, based on 'develop'
- You are now on branch 'feature/2Revision'

Now, start committing on your feature. When done, use:

    git flow feature finish 2Revision

Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/2Revision)
```

Simulo la feature con un archivo md otravez:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/2Revision)
$ touch fichero2.md
```

Añado el fichero2.md al seguimiento:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/2Revision)
$ git add fichero2.md
```

Hago commit:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/2Revision)
$ git commit -m "añadido fichero2.md"
[feature/2Revision 5840303] añadido fichero2.md
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 fichero2.md
```

Revisamos que esté el commit:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/2Revision)
$ git log --oneline
5840303 (HEAD -> feature/2Revision) añadido fichero2.md
```

Me cambio a la rama de la 1Revision:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/2Revision)
$ git checkout feature/1Revision
Switched to branch 'feature/1Revision'
```

Y damos por terminada la 1Revision, que se mandará de la rama de feature a develop:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/1Revision)
$ git flow feature finish
Switched to branch 'develop'
Updating fc72b1e..ae736d5
Fast-forward
 fichero.md | 0
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 fichero.md
Deleted branch feature/1Revision (was ae736d5).

Summary of actions:
- The feature branch 'feature/1Revision' was merged into 'develop'
- Feature branch 'feature/1Revision' has been locally deleted
- You are now on branch 'develop'
```

Podemos ver que ya no sigue 1Revision:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/2Revision)
$ git branch
develop
* feature/2Revision
master
```

Hago lo mismo con la 2Revision para terminar ya con las dos:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (develop)
$ git checkout feature/2Revision
Switched to branch 'feature/2Revision'
```

La que nos falta es mergear la 2Revision de la branch dentro de develop también:

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (feature/2Revision)
$ git flow feature finish
Switched to branch 'develop'
Merge made by the 'ort' strategy.
 fichero2.md | 0
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 fichero2.md
Deleted branch feature/2Revision (was 5840303).

Summary of actions:
- The feature branch 'feature/2Revision' was merged into 'develop'
- Feature branch 'feature/2Revision' has been locally deleted
- You are now on branch 'develop'
```

Con git log volvemos a ver los commits hechos hasta ahora desde la rama develop ya que en feature no nos queda ninguna revisión pendiente que confirmar por terminada:

```
$ git log
commit 5dd2e4d8b8d5b83d73fe8eac3f86fb49116545c3 (HEAD -> develop)
Merge: ae736d5 5840303
Author: [Andrei-DG] <[AndreiSanfer@gmail.com]>
Date:   Fri Sep 30 18:18:16 2022 +0200

    Merge branch 'feature/2Revision' into develop

commit 584030397fd3e7696eee4377de8049df95fdcec8
Author: [Andrei-DG] <[AndreiSanfer@gmail.com]>
Date:   Fri Sep 30 17:49:22 2022 +0200

    añadido fichero2.md

commit ae736d5b90980ae2e7ec7796c0364d147c9045f7
Author: [Andrei-DG] <[AndreiSanfer@gmail.com]>
Date:   Fri Sep 30 17:33:28 2022 +0200

    añadido fichero.md

commit fc72b1ec108d3a6d0e353a2f758f7f1af30d79d3 (master)
Author: [Andrei-DG] <[AndreiSanfer@gmail.com]>
Date:   Fri Sep 30 17:20:58 2022 +0200

    Initial commit
```

```
Andrei@DESKTOP-VVV322F MINGW64 ~/Desktop (develop)
$ git log --oneline
5dd2e4d (HEAD -> develop) Merge branch 'feature/2Revision' into develop
5840303 añadido fichero2.md
ae736d5 añadido fichero.md
fc72b1e (master) Initial commit
```

4. EXTRA del EXTRA: borrar fichero de todos los commits de un repositorio.

Todos los commits que tenía en un repositorio de la tarea GitBasico2ASIR:

```
a013dee (HEAD -> main, origin/main, origin/HEAD) En el conflicto ni del MD de Inglés, elijo v.Github
88b176a Se modificó de Inglés el apartado Objetivos (MWSC)
e9707fd Modificado Objetivos
483e988 Modificado ruta al Encabezado ASIR y nombre de png
ef5ca95 Añadido README de Repositorio y Asignaturas y 2png
```

Resteamos y comprobamos:

```
PS C:\Users\Andrei\Desktop\GitExtra3Prueba> git reset --hard
HEAD is now at ef5ca95 Añadido README de Repositorio y Asignaturas y 2png
PS C:\Users\Andrei\Desktop\GitExtra3Prueba> git log --oneline
ef5ca95 (HEAD -> main) Añadido README de Repositorio y Asignaturas y 2png
PS C:\Users\Andrei\Desktop\GitExtra3Prueba>
```

Ahora, si lo teníamos ya subido el repositorio a remoto lo mejor sería hacer un git revert:

```
Revert "Añadido README de Repositorio y Asignaturas y 2png"
This reverts commit ef5ca9586a9c2e8f0e0fab5bcd985f81fea8265.

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch main
# Your branch is behind 'origin/main' by 4 commits, and can be fast-forwarded.
#   (use "git pull" to update your local branch)
#
# Changes to be committed:
#   deleted:    A.D.G.-Logo.png
#   deleted:    "Administraci\303\263n de sistemas gestores de bases de datos/README.md"
#   deleted:    "Administraci\303\263n de sistemas operativos/README.md"
#   deleted:    Empresa e iniciativa emprendedora/README.md
#   deleted:    Encabezado ASIR.PNG
#   deleted:    "Implantaci\303\263n de aplicaciones web/README.md"
#   deleted:    "Ingl\303\251s/README.md"
#   deleted:    README.md
#   deleted:    Seguridad y alta disponibilidad/README.md
#   deleted:    Servicios en red/README.md
#
```

Escribimos :wq y enter.

