

## Actividad Evaluable

Módulo	Servicios de red
Nombre y Apellidos:	Álvaro Santa Cruz Montero
Nombre y Apellidos:	Érika Concha Duque
Nombre y Apellidos:	Ángel Cañas Gavilanes
Estas soluciones deben ser entregadas en PDF.	

**FECHA DE ENTREGA: --/--/21**

### Objetivos

---

El objetivo de esta actividad individual es la de repasar, asentar y adquirir un mayor conocimiento de lo impartido en clase.

El archivo debe nombrarse como: **nombre\_apellido1\_apellido2.PDF**

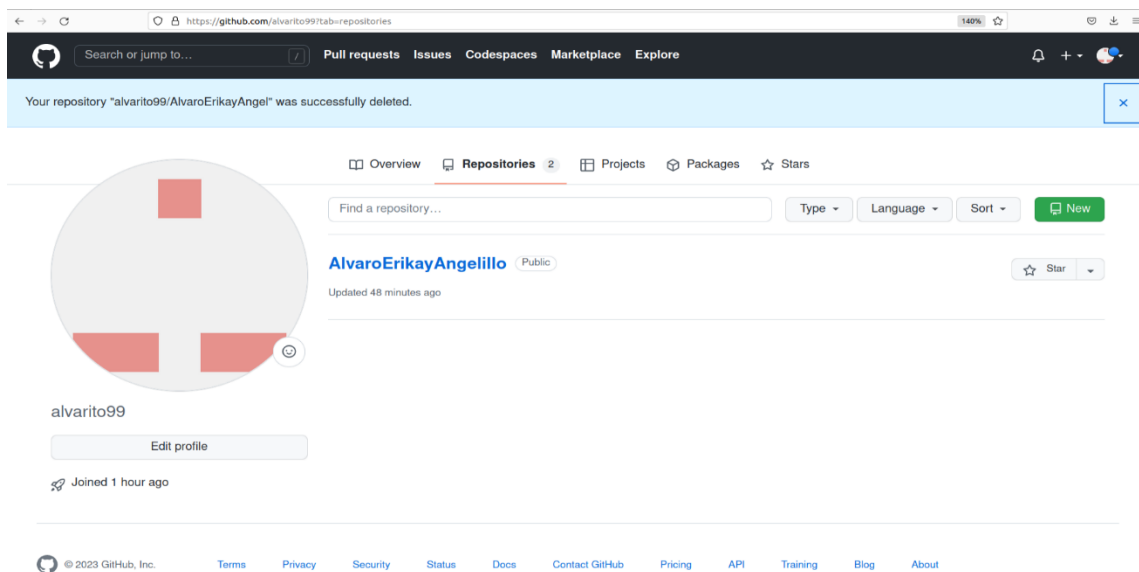
Ej: Belen\_Vargas\_Sabater.pdf

## Cuestiones

EJERCICIO 1: **Práctica:** Realizar una guía práctica a modo de tutorial, sobre el uso de GitHub, algunos de los elementos que debe cubrir son:

- Creación de repositorio y conexión con repositorio personal de gitHub mediante terminal con ejemplos

Lo primero es crearnos una cuenta en GitHub, una vez hecho, creamos un nuevo repositorio.



Desde la terminal de **Ubuntu** (Linux), creamos una carpeta llamada "trabajo".

Iniciamos un git desde esta carpeta (**git init**), luego añadimos en este caso el README.md que es como una descripción de nuestro repositorio (add README.md).

Creamos el commit (**git commit -m**) y también lo vinculamos de forma remota con nuestro repositorio de gihhub a través del enlace.

En el siguiente ejercicio veremos en que consiste **el push y el pull**, y en este caso hacemos un push para subir a Github el commit de origen.

A screenshot of a terminal window titled 'Actividades' and 'Terminal'. The terminal shows a series of commands being executed in a virtual machine named 'alvaroubuntu@alvaroubuntu-VirtualBox'. The commands are: `cd Escritorio/trabajo/`, `echo "# AlvaroErikayAngelillo" >> README.md`, `git init`, `git add README.md`, `git commit -m "first commit"`, `git branch -M main`, `git remote add origin https://github.com/alvarito99/AlvaroErikayAngelillo.git`, and `git push -u origin main`. The terminal output shows the successful execution of these commands.

Una vez hecho esto nos pedirá el nombre de nuestro usuario de **GitHub** y una contraseña. Para saber lo que poner en la contraseña tendremos que ir a los ajustes de GitHub y generar un **token**.

Entramos en ajustes, luego a la opción **developer setting**. Le damos a crear nuevo token clásico. Y seleccionamos las siguientes opciones:

<input checked="" type="checkbox"/> <b>repo</b>	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo_deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input type="checkbox"/> <b>admin:org</b>	Full control of orgs and teams
<input type="checkbox"/> write:org	Read and write org and team membership
<input type="checkbox"/> read:org	Read org and team membership
<input type="checkbox"/> <b>admin:public_key</b>	Full control of user public keys
<input type="checkbox"/> write:public_key	Write user public keys
<input type="checkbox"/> read:public_key	Read user public keys
<input type="checkbox"/> <b>admin:repo_hook</b>	Full control of repository hooks
<input type="checkbox"/> write:repo_hook	Write repository hooks
<input checked="" type="checkbox"/> read:repo_hook	Read repository hooks
<input type="checkbox"/> <b>admin:org_hook</b>	Full control of organization hooks
<input type="checkbox"/> <b>gist</b>	Create gists
<input type="checkbox"/> <b>notifications</b>	Access notifications
<input type="checkbox"/> <b>user</b>	Update all user data
<input type="checkbox"/> user:email	Access user email addresses (read-only)
<input type="checkbox"/> user:follow	Follow and unfollow users
<input checked="" type="checkbox"/> <b>delete_repo</b>	Delete repositories

Esto generará otro un **link** que tendrás que copiar para utilizar en la contraseña que nos pedían previamente.

```
Username for 'https://github.com': alvarito99
Password for 'https://alvarito99@github.com':
Enumerando objetos: 3, listo.
Contando objetos: 100% (3/3), listo.
Escribiendo objetos: 100% (3/3), 237 bytes | 237.00 KiB/s, listo.
Total 3 (delta 0), reusados 0 (delta 0), pack-reusados 0
To https://github.com/alvarito99/AlvaroErikayAngelillo.git
* [new branch]      main -> main
Rama 'main' configurada para hacer seguimiento a la rama remota 'main' de 'origin'.
alvaroubuntu@alvaroubuntu-VirtualBox:~/Escritorio/trabajo$
```

- Push Y pulls a los repositorios desde el terminal con ejemplos.

El comando **git push** se usa para cargar contenido del repositorio local a un repositorio remoto. El envío es la forma de transferir confirmaciones desde tu repositorio local a un repositorio remoto.

En este caso creamos un archivo de texto, escribimos lo que queramos y hacemos un `git add *`, después creamos el commit como vemos en la captura de pantalla.

Ya solo nos faltaría utilizar el comando `git push` para mandar los cambios y el commit al repositorio remoto de GitHub.

```
alvaroubuntu@alvaroubuntu-VirtualBox:~/Escritorio/trabajo$ nano prueba.txt
alvaroubuntu@alvaroubuntu-VirtualBox:~/Escritorio/trabajo$ git add *
alvaroubuntu@alvaroubuntu-VirtualBox:~/Escritorio/trabajo$ git commit -m "hemos creado un txt"
[main e230c73] hemos creado un txt
1 file changed, 1 insertion(+)
create mode 100644 prueba.txt
alvaroubuntu@alvaroubuntu-VirtualBox:~/Escritorio/trabajo$ git push
Username for 'https://github.com': alvarito99
Password for 'https://alvarito99@github.com':
Enumerando objetos: 4, listo.
Contando objetos: 100% (4/4), listo.
Comprimiendo objetos: 100% (2/2), listo.
Escribiendo objetos: 100% (3/3), 316 bytes | 316.00 KiB/s, listo.
Total 3 (delta 0), reusados 0 (delta 0), pack-reusados 0
To https://github.com/alvarito99/AlvaroErikayAngelillo.git
1744fe8..e230c73  main -> main
```



GNU nano 6.2 prueba.txt  
angelito fue a pescar y se cayo

**Git pull** es un comando de Git utilizado para actualizar la versión local (desde Linux) de un repositorio desde otro remoto (GitHub).

Nos vamos a nuestro repositorio de GitHub entramos en uno de los commit creados donde teníamos un documento de texto llamado “prueba.txt”

```
1 angelito fue a pescar y se cayó.
2 Erika fue a comprar con sus padres y se perdió.
```

Añadimos en la línea de texto número 2 una frase cualquiera para ver si al aplicar el `git pull` en el repositorio local La modificación se baja y se descarga en este mismo.

Nos vamos al terminal de Ubuntu y aplicamos el git pull, como vemos en la imagen se baja los cambios aplicados en el repositorio remoto. Y ya podremos seguir trabajando con esos datos.

```
alvaroubuntu@alvaroubuntu-VirtualBox:~/Escritorio/trabajo$ git pull
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Desempaquetando objetos: 100% (3/3), 723 bytes | 723.00 KiB/s, listo.
Desde https://github.com/alvarito99/AlvaroErikayAngelillo
e230c73..2f66c17  main      -> origin/main
Actualizando e230c73..2f66c17
Fast-forward
 prueba.txt | 3 ++-
 1 file changed, 2 insertions(+), 1 deletion(-)
alvaroubuntu@alvaroubuntu-VirtualBox:~/Escritorio/trabajo$ cat prueba.txt
angelito fue a pescar y se cayó.
Erika fue a comprar con sus padres y se perdió.
```

- Cada miembro del equipo debe realizar cambios en el repositorio propiedad de uno de los miembros, documenta el proceso.

En este caso el compañero Ángel abre desde su equipo la terminal de Ubuntu.

Desde nuestra cuenta de Github tenemos que añadir como colaborador a nuestro compañero Ángel (añadiendo su cuenta de GitHub).

En la terminal utiliza el comando **git clone** y el enlace de nuestro repositorio remoto. Una vez hecho esto accede a nuestro repositorio desde la terminal.

Decidió continuar el trabajo y crear un nuevo documento de texto (pruebafinal3.txt) añadiendo al staying area y posteriormente creando un nuevo commit.

Por último, realiza un git push para subirlo al repositorio remoto de la cuenta de alvarito99.

Aclarar que angel tendrá que utilizar su propia contraseña de GitHub para poder pushear.

Todo esto se ve en la siguiente imagen:

```
angel@angel-VirtualBox:~$ git clone https://github.com/alvarito99/AlvaroErikayAngelillo
Clonando en 'AlvaroErikayAngelillo'...
remote: Enumerating objects: 12, done.
remote: Counting objects: 100% (12/12), done.
remote: Compressing objects: 100% (8/8), done.
remote: Total 12 (delta 1), reused 9 (delta 0), pack-reused 0
Recibiendo objetos: 100% (12/12), listo.
Resolviendo deltas: 100% (1/1), listo.
angel@angel-VirtualBox:~$ cd Escritorio
angel@angel-VirtualBox:~/Escritorio$ ls -la
.  .. .git git trabajo
angel@angel-VirtualBox:~/Escritorio$ cd ../
angel@angel-VirtualBox:~$ ls -la
.  .. .cache  Imágenes  Público
   .config  .lessht  snap
AlvaroErikayAngelillo Descargas  .local  .sudo_as_admin_successful
.bash_history  Documentos Música  Videos
.bash_logout   Escritorio Plantillas
.bashrc        .gitconfig .profile
angel@angel-VirtualBox:~$ cd AlvaroErikayAngelillo/
angel@angel-VirtualBox:~/AlvaroErikayAngelillo$ touch pruebafinal3.txt
angel@angel-VirtualBox:~/AlvaroErikayAngelillo$ nano pruebafinal3.txt
angel@angel-VirtualBox:~/AlvaroErikayAngelillo$ nano pruebafinal3.txt
angel@angel-VirtualBox:~/AlvaroErikayAngelillo$ git add *
angel@angel-VirtualBox:~/AlvaroErikayAngelillo$ git status -s
A pruebafinal3.txt
angel@angel-VirtualBox:~/AlvaroErikayAngelillo$ git commit -m "nuevo commit"
[main 2a1685a] nuevo commit
1 file changed, 1 insertion(+)
 create mode 100644 pruebafinal3.txt
angel@angel-VirtualBox:~/AlvaroErikayAngelillo$ git push
Username for 'https://github.com': angeeel4
Password for 'https://angeeel4@github.com':
Enumerando objetos: 4, listo.
Contando objetos: 100% (4/4), listo.
Comprimiendo objetos: 100% (2/2), listo.
Escribiendo objetos: 100% (3/3), 309 bytes | 309.00 KiB/s, listo.
Total 3 (delta 1), reusados 0 (delta 0), pack-reusados 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/alvarito99/AlvaroErikayAngelillo
 54d2abc..2a1685a main -> main
```

Finalmente nos queda comprobar que los cambios que realizó Ángel desde su equipo se han guardado en el repositorio remoto llamado AlvaroErikayAngelillo.

Como vemos en la siguiente captura de pantalla Angel creó dos commits llamados “pruebafinal.txt” y “pruebafinal3.txt”.

	README.md	first commit
	prueba.txt	prueba.txt
	pruebafinal.txt	nuevo commit desde otro equipo
	pruebafinal3.txt	nuevo commit