

LABORATORIO DE SISTEMAS TELEMÁTICOS II



Universidad
del Cauca

MICROPROYECTO GITHUB

Estudiante:

Viviana Burbano

Ingeniero:

Fulvio Yesid Vivas Cantero

Facultad de Ingeniería Electrónica y Telecomunicaciones

Programa Tecnología en Telemática

Universidad del Cauca

Popayán

2023

DIARIO DE LABORATORIO

Diario de Campo - Laboratorio: Microproyecto de Github

Fecha: 09 de septiembre de 2023

Objetivos del Laboratorio:

1. Manejo y creación de repositorios Git
2. Realizar las modificaciones requeridas en el código del juego TicTacToe.
3. Realizar pruebas de funcionamiento tanto de los repositorios git como del juego.

Descripción del problema:

Se solicita crear repositorios locales y remotos en git para poder realizar transferencia de datos entre cliente y servidor. Estos datos requeridos van a ser información sobre el juego TicTacToe.

Materiales y Equipos Utilizados:

- Computador
- Conexión a internet.
- GitHub
- VirtualBox: Es un software de virtualización de código abierto desarrollado por Oracle Corporation. Permite la creación y ejecución de máquinas virtuales en un sistema host, lo que significa que los usuarios pueden instalar y ejecutar múltiples sistemas operativos (como Windows, Linux, macOS, etc.) de forma simultánea en una única computadora física.
- Sistema operativo para la máquina virtual: ubuntu-22.04.3-live-server-amd64.iso.
- Sistema operativo local Windows 10 Pro

Procedimiento:

Descripción paso a paso del desarrollo de la práctica:

Instalación Git Server en Ubuntu 22.04

- Instalar Git usando APT:

1. Actualizar el servidor:

```
vburbano@vburbano:~$ sudo apt update -y
```

2. Instalación y versión de Git:

```
vburbano@vburbano:~$ sudo apt install git -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git ya está en su versión más reciente (1:2.34.1-1ubuntu1.10).
```

CONFIGURACIÓN GIT

1. Crear usuario:

```
vburbano@vburbano:~$ sudo adduser gitvb
[sudo] password for vburbano:
Adding user `gitvb' ...
Adding new group `gitvb' (1006) ...
Adding new user `gitvb' (1004) with group `gitvb' ...
Creating home directory `/home/gitvb' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for gitvb
Enter the new value, or press ENTER for the default
  Full Name []: vburbanob
  Room Number []: 123
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] y
vburbano@vburbano:~$ _
```

2. Inicio de sesión como usuario de git

```
vburbano@vburbano:~$ sudo su - gitvb
gitvb@vburbano:~$ _
```

3. Se realiza la creación de un nuevo repositorio (Tic Tac toe) vacío, usando el siguiente comando:

```
gitvb@vburbano:~$ git init --bare ~/TicTacToe.git
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/gitvb/TicTacToe.git/
```

```
gitvb@vburbano:~$ ls
TicTacToe.git
```

4. Habilitar el enlace posterior a la actualización copiando el archivo de muestra de la siguiente manera:

```
gitvb@vburano:~$ cd TicTacToe.git/hooks/
gitvb@vburano:~/TicTacToe.git/hooks$ cp post-update.sample post-update
gitvb@vburano:~/TicTacToe.git/hooks$ ls
applypatch-msg.sample      pre-applypatch.sample      pre-rebase.sample
commit-msg.sample          pre-commit.sample          pre-receive.sample
fsmonitor-watchman.sample  pre-merge-commit.sample    push-to-checkout.sample
post-update                 prepare-commit-msg.sample  update.sample
post-update.sample          pre-push.sample
```

INSTALACIÓN Y CONFIGURACIÓN DE GIT COMO CLIENTE

Como cliente se utilizará Windows 10 Pro.

1. Instalación git: Git se instaló desde google.
2. Versión de Git

```
C:\Users\TELEMÁTICA>git --version
git version 2.42.0.windows.2
C:\Users\TELEMÁTICA>
```

- Configuración de git:

Nota: La ip del servidor es: 192.168.128.3. En este caso la ip del servidor es de una máquina virtual con el sistema operativo de ubuntu server 22.04.

```
gitvb@vburano:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.128.3 netmask 255.255.255.0 broadcast 192.168.128.255
```

1. Asignamos nombre de usuario y email (ip del servidor):

```
TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~
$ git config --global user.name "gitvb"
TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~
$ git config --global user.email "gitvb@192.168.128.3"
TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~
$ cat .gitconfig
[user]
    name = gitvb
    email = gitvb@192.168.128.3
```

2. Crear un directorio donde se guardarán todos tus proyectos:

```
TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~
$ mkdir ~/gitproyectos
TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~
$ cd ~/gitproyectos/
TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos
```

3. Se crea un clon del repositorio TicTacToe.git que hemos creado

antes en el servidor:

```
TELEMATICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos
$ git clone gitvb@192.168.128.3:~/TicTacToe.git TicTacToe.git
Cloning into 'TicTacToe.git'...
The authenticity of host '192.168.128.3 (192.168.128.3)' can't be established.
ED25519 key fingerprint is SHA256:YuaF7KvXy12XRMNK/BD5V+e/FRPkD+hLGU1wb35EagQ.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '192.168.128.3' (ED25519) to the list of known hosts.
gitvb@192.168.128.3's password:
Permission denied, please try again.
gitvb@192.168.128.3's password:
warning: You appear to have cloned an empty repository.
```

```
TELEMATICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos
$ ls
TicTacToe.git/
```

4. Crear un archivo txt:

```
TELEMATICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos
$ echo "hola git" > hola.txt
```

5. Agregue estos archivos a nuestro repositorio git:

```
TELEMATICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos/TicTacToe.git (master)
$ git add hola.txt
```

6. Crear un commit inicial, con el mensaje sud

```
TELEMATICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos/TicTacToe.git (master)
$ git commit -am "Initial Commit"
[master (root-commit) c5b6f44] Initial Commit
1 file changed, 1 insertion(+)
create mode 100644 hola.txt
```

7. Realizar un push del repositorio local al repositorio remoto.

```
TELEMATICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos/TicTacToe.git (master)
$ git push origin master
gitvb@192.168.128.3's password:
Permission denied, please try again.
gitvb@192.168.128.3's password: .
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 216 bytes | 216.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To 192.168.128.3:~/TicTacToe.git
* [new branch]      master -> master
```

8. Verifique los cambios, acceda al servidor git y ejecute el siguiente comando para

revisa los registros:

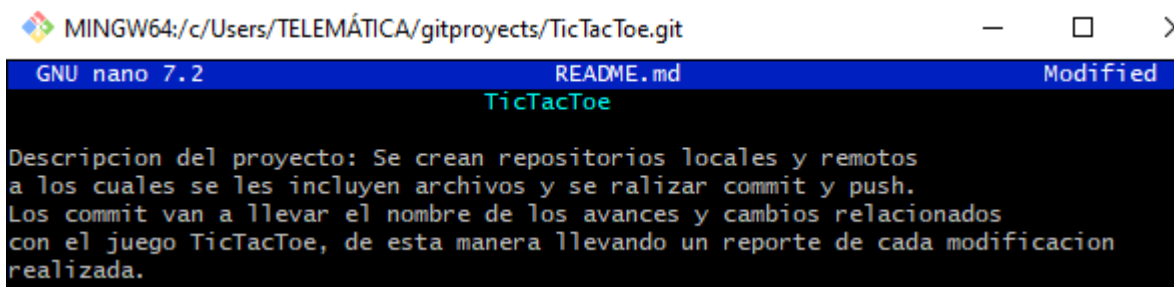
```
TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos/TicTacToe.git (master)
$ git log
commit c5b6f44550a9940972a921f671454c0d3fa680c2 (HEAD -> master, origin/master)
Author: gitvb <gitvb@192.168.128.3>
Date: Thu Oct 5 17:42:07 2023 -0500

Initial Commit
```

9. Se agrega un archivo README.md que explique el proyecto y muestre una imagen de ejemplo.
- Se crea el archivo con el comando "nano":

```
TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos/TicTacToe.git (master)
$ nano README.md
```

- Descripción del proyecto:



The screenshot shows the nano text editor interface. The title bar indicates the file path: MINGW64:/c/Users/TELEMÁTICA/gitproyectos/TicTacToe.git. The editor window shows the README.md file with the following content:

```
GNU nano 7.2 README.md Modified
TicTacToe

Descripcion del proyecto: Se crean repositorios locales y remotos
a los cuales se les incluyen archivos y se realizar commit y push.
Los commit van a llevar el nombre de los avances y cambios relacionados
con el juego TicTacToe, de esta manera llevando un reporte de cada modificacion
realizada.
```

- Pasos para agregar la imagen a README.md:

- Se copia la ruta donde se encuentra la imagen y se coloca el nombre del directorio al que se desea agregar.

```
TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos/TicTacToe.git (master)
$ cp /c/Users/TELEMÁTICA/Pictures/TTTJUEGO.png ~/gitproyectos/TicTacToe.git
```

- Una vez que la imagen se encuentra en el directorio adecuado en tu repositorio local, utiliza el siguiente comando de Git para agregarla al área de preparación:
- Git add

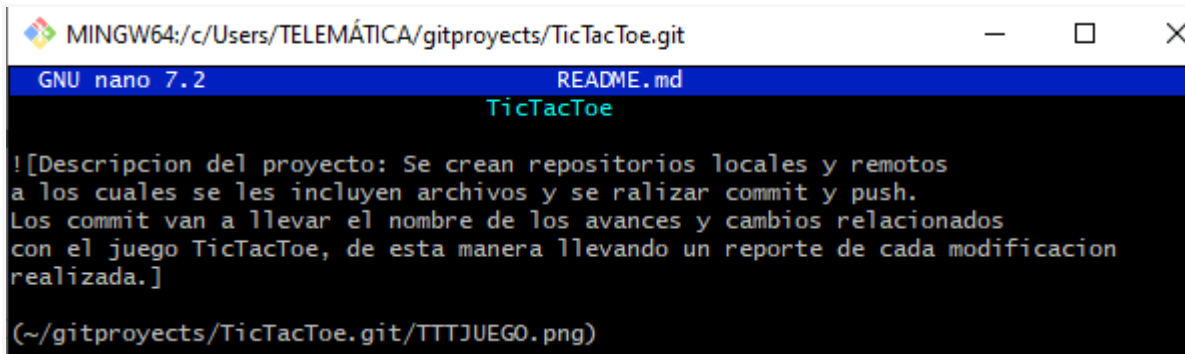
```
TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos/TicTacToe.git (master)
$ git add ~/gitproyectos/TicTacToe.git/TTTJUEGO.png
```

- Hacemos un git status para verificar los pasos realizados con git add:

```
TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos/TicTacToe.git (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)
    new file:   README.md
    new file:   TTTJUEGO.png
```

- Ingresamos al archivo README.md e ingresamos la ruta de la imagen:



```
MINGW64:/c/Users/TELEMÁTICA/gitproyectos/TicTacToe.git
GNU nano 7.2 README.md
TicTacToe

! [Descripción del proyecto: Se crean repositorios locales y remotos
a los cuales se les incluyen archivos y se realiza commit y push.
Los commit van a llevar el nombre de los avances y cambios relacionados
con el juego TicTacToe, de esta manera llevando un reporte de cada modificación
realizada.]

(~/gitproyectos/TicTacToe.git/TTTJUEGO.png)
```

10. Añadir la carpeta del juego TicTacToe a TicTacToe.git para poder subirlo a git.

```
TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos/TicTacToe.git (master)
$ mv ~/eclipse-workspace/TicTacToe ~/gitproyectos/TicTacToe
```

```
TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos/TicTacToe.git (master)
$ mv ~/gitproyectos/TicTacToe ~/gitproyectos/TicTacToe.git

TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos/TicTacToe.git (master)
$ ls
README.md TTTJUEGO.png TicTacToe/ hola.txt

TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos/TicTacToe.git (master)
$ git add TicTacToe/
```

11. Commit:

```
TELEMÁTICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos/TicTacToe.git (master)
$ git log
commit 87d6577b77de3248ca129bc8631cce750ea1f378 (HEAD -> master)
Author: gitvb <gitvb@192.168.128.3>
Date: Mon Oct 9 16:33:40 2023 -0500

    Código del juego sin modificaciones
```

Observaciones: El informe no se pudo completar por un inconveniente con la ip del servidor, debido a que al volver a encender la máquina virtual, está había cambiado su ip, por lo tanto no se pudo seguir haciendo el push entre cliente y servidor, se intentó colocar la ip estática pero seguía sin hacer la conexión.

Siguió presentando el error al push y al final no se logró volver hacer conexión.

Resultados:

Se crea el primer commit, con el mensaje "initial commit"

```
TELEMATICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos/TicTacToe.git (master)
$ git log ←
commit c5b6f44550a9940972a921f671454c0d3fa680c2 (HEAD -> master, origin/master)
Author: gitvb <gitvb@192.168.128.3>
Date: Thu Oct 5 17:42:07 2023 -0500

Initial Commit
```

El primer push se logró correctamente:

- ip del servidor: 192.168.128.3
- Push de cliente a servidor.

```
TELEMATICA@DESKTOP-9BFU78E MINGW64 ~/gitproyectos/TicTacToe.git (master)
$ git push origin master ←
gitvb@192.168.128.3's password:
Permission denied, please try again.
gitvb@192.168.128.3's password: .
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 216 bytes | 216.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To 192.168.128.3:~/TicTacToe.git
* [new branch]      master -> master
```

Análisis y Discusión:

Los resultados obtenidos no fueron los esperados, se presentaron inconvenientes que por cuestiones de materiales y tiempo, no se logró la finalidad del microproyecto.

Lo obtenido se realizó sin problemas, mostrando así la conexión entre cliente y servidor, la creación de usuarios, creación de commits, creación del archivo README.md, agregar una imagen y un push inicial.

Conclusiones:

1. Podemos concluir que la instalación y configuración del cliente y el servidor se pudieron realizar sin inconvenientes siguiendo la guía otorgada.
2. Se configuró y comprobó el push entre cliente y servidor.