



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

# Linux Sistemas Embebidos

# GitHub

P R E S E N T A N

González Pérez Cesar Israel  
Ochoa Ortiz Abel  
Reyes González David Antonio  
García Silva Adán Alberto

PROFESOR: Iván Chavero

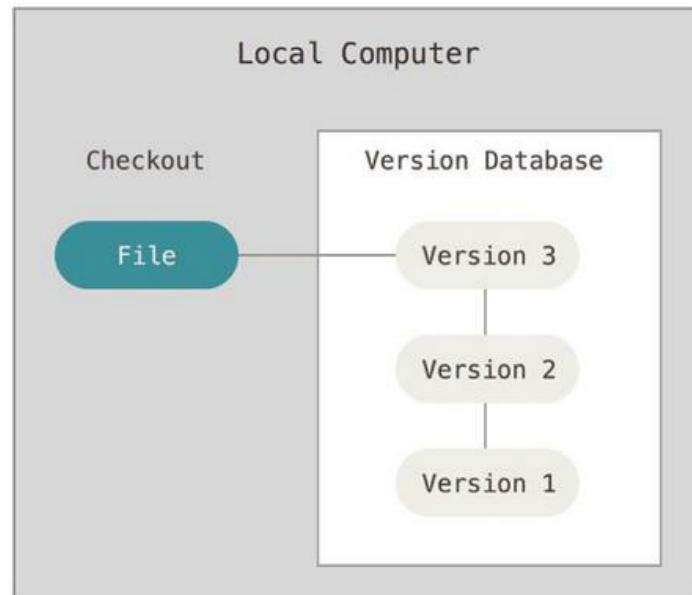
MÉXICO, D.F.

ABRIL, 2016

# Introducción

¿Qué es GitHub?

GitHub es una plataforma de desarrollo colaborativo de software para alojar proyectos utilizando el sistema de control de versiones Git.



GitHub aloja tu repositorio de código y te brinda herramientas muy útiles para el trabajo en equipo, dentro de un proyecto. Algo muy importante, con esto podemos contribuir a mejorar el desarrollo de los demás.

## Control de versiones

El control de versiones hoy en día es una de las actividades primordiales en la administración de proyectos para el desarrollo de software de forma general. De este modo surge la necesidad de mantener y llevar control del código que vamos programando, manteniendo los cambios hechos al mismo. Esto agiliza el desarrollo en equipo y los cambios impactados a un desarrollo.

Es necesario gestionar los cambios sin importar el número de involucrados en un desarrollo logrando tener un control en las versiones y en el avance del mismo.

En cuanto a equipos de trabajo se refiere, todavía se hace más necesario disponer de un control de versiones.

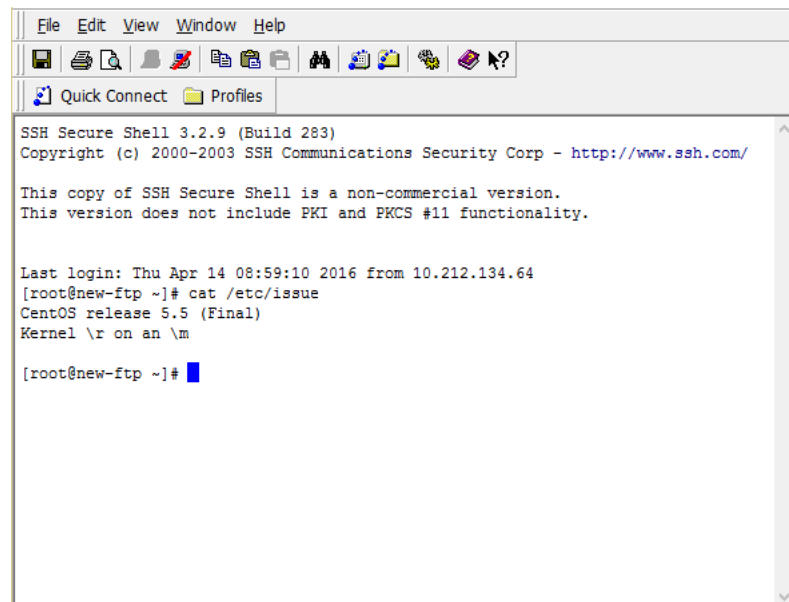
## Desarrollo

A continuación, se explicará a detalle lo realizado con esta herramienta demostrando las grandes capacidades de la misma, así como los problemas comunes y como resolverlos.

Partimos de colocar un repositorio maestro en el que todos los involucrados en el equipo realicen sus updates a las partes correspondientes de sus actividades. Para este repositorio decidimos realizarlo en un servidor en producción con una IP publica para el acceso de cada uno de los integrantes.

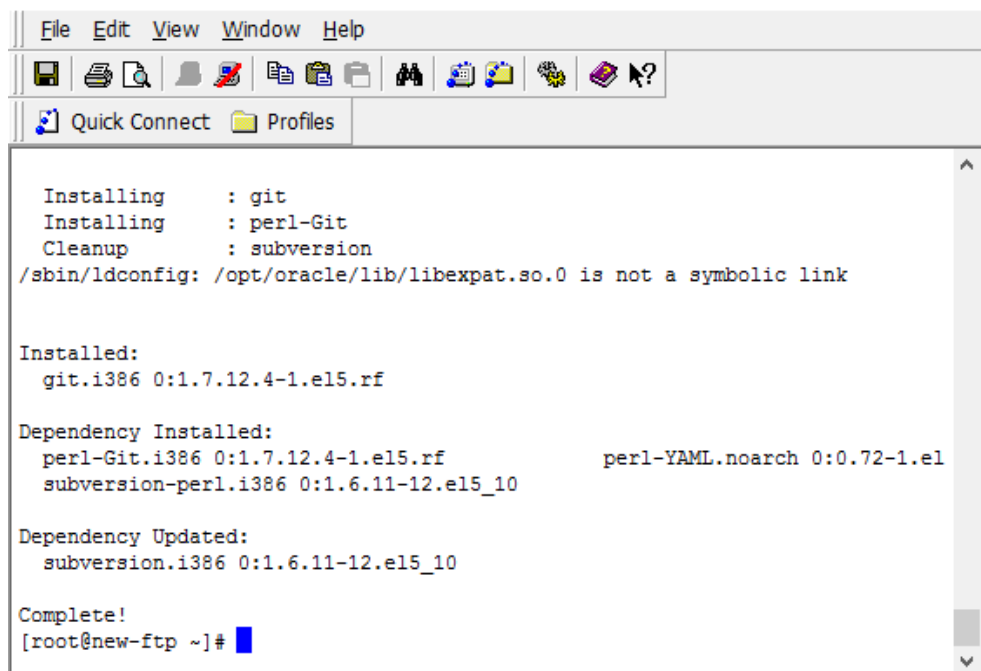
Partiendo de la instalación del servidor, se realizó lo siguiente:

- 1- Se realizará en una instancia Linux, en nuestro caso lo implementamos sobre CentOS.

A screenshot of an SSH terminal window. The window has a menu bar with 'File', 'Edit', 'View', 'Window', and 'Help'. Below the menu bar is a toolbar with various icons. The main area of the window displays the output of an SSH session. It starts with 'SSH Secure Shell 3.2.9 (Build 283)' and 'Copyright (c) 2000-2003 SSH Communications Security Corp - http://www.ssh.com/'. It then shows a disclaimer: 'This copy of SSH Secure Shell is a non-commercial version. This version does not include PKI and PKCS #11 functionality.' The next line is 'Last login: Thu Apr 14 08:59:10 2016 from 10.212.134.64'. The prompt is '[root@new-ftp ~]#'. The user has entered 'cat /etc/issue', and the output is 'CentOS release 5.5 (Final)' and 'Kernel \r on an \m'. The prompt is now '[root@new-ftp ~]#' with a blue cursor.

- 2- Procedemos a instalar una instancia de Git. Para hacerlo en CentOS instalaremos el paquete de Git de este modo, para lo que ejecutamos lo siguiente:

```
yum install git
```



```
File Edit View Window Help
[Icons]
Quick Connect Profiles

Installing      : git
Installing      : perl-Git
Cleanup        : subversion
/sbin/ldconfig: /opt/oracle/lib/libexpat.so.0 is not a symbolic link

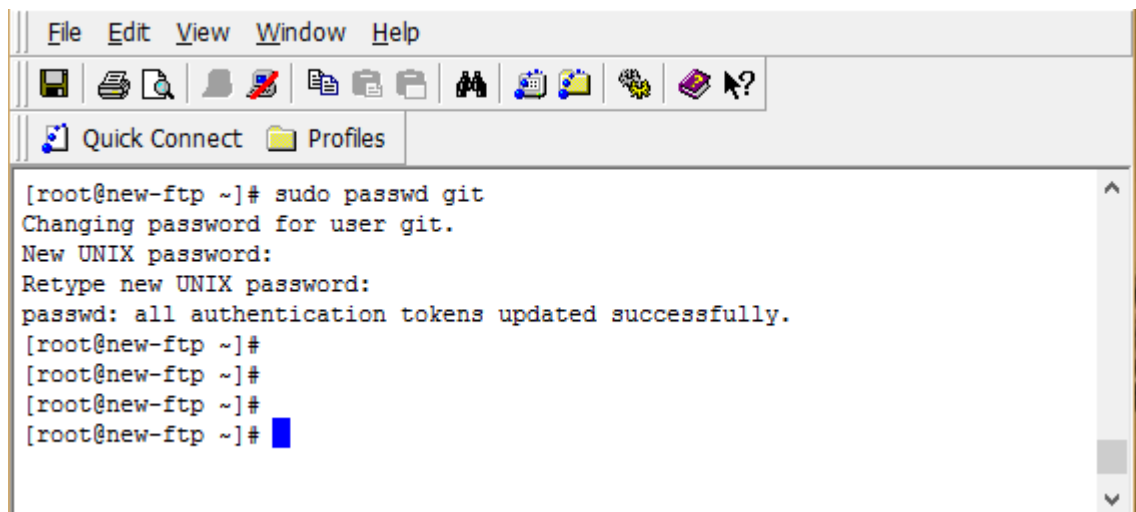
Installed:
git.i386 0:1.7.12.4-1.el5.rf

Dependency Installed:
perl-Git.i386 0:1.7.12.4-1.el5.rf          perl-YAML.noarch 0:0.72-1.el
subversion-perl.i386 0:1.6.11-12.el5_10

Dependency Updated:
subversion.i386 0:1.6.11-12.el5_10

Complete!
[root@new-ftp ~]#
```

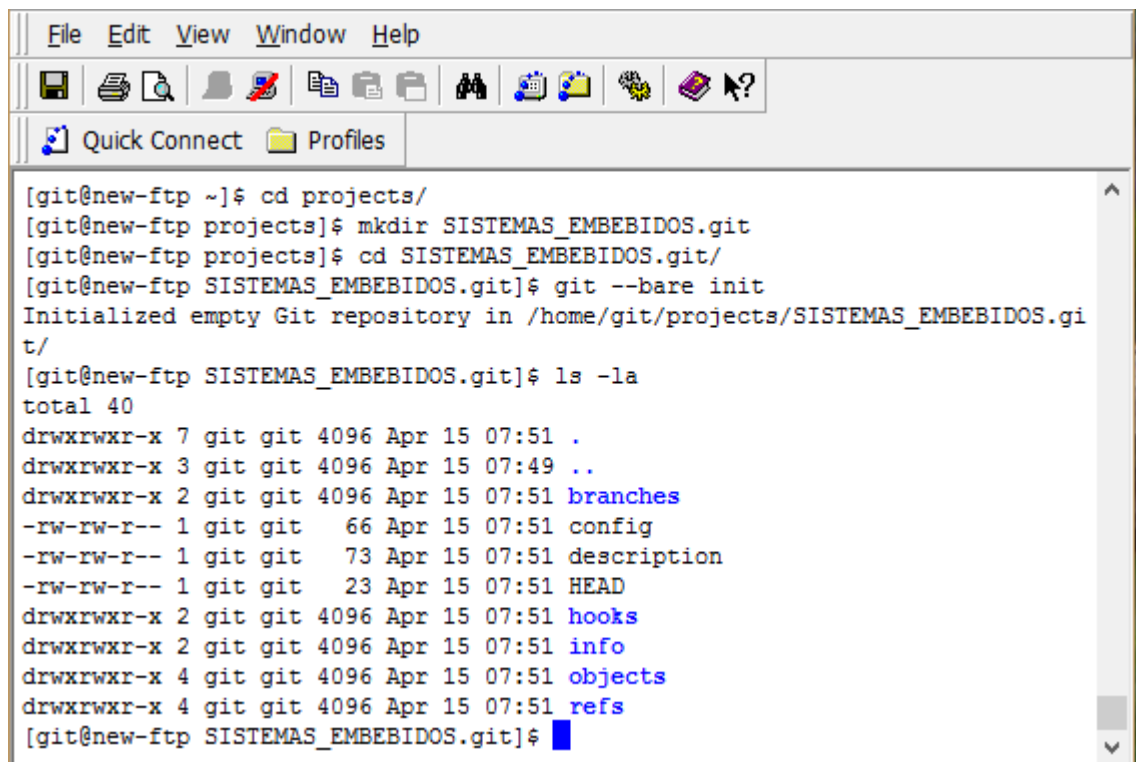
- 3- Para la gestión con Git crearemos un usuario y le asignamos en la siguiente forma de la siguiente forma:



```
File Edit View Window Help
[Icons]
Quick Connect Profiles

[root@new-ftp ~]# sudo passwd git
Changing password for user git.
New UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully.
[root@new-ftp ~]#
[root@new-ftp ~]#
[root@new-ftp ~]#
[root@new-ftp ~]#
```

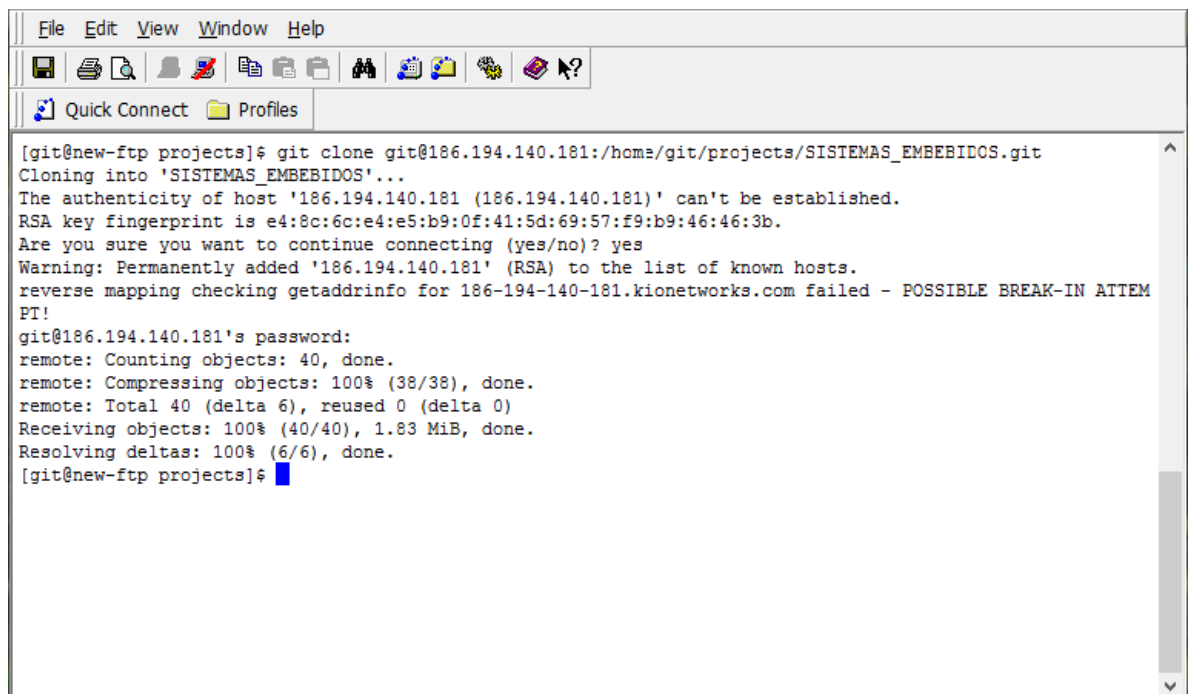
- 4- Ingresamos con el usuario creado y posteriormente creamos una carpeta donde se localizarán los proyectos que manejemos en Git, nombraremos nuestro proyecto y accederemos a esa carpeta, una vez dentro de este directorio vamos a crear el proyecto de Git con el comando:



The screenshot shows a terminal window with a menu bar (File, Edit, View, Window, Help) and a toolbar with various icons. Below the toolbar are tabs for 'Quick Connect' and 'Profiles'. The terminal text shows the following commands and output:

```
[git@new-ftp ~]$ cd projects/
[git@new-ftp projects]$ mkdir SISTEMAS_EMBEBIDOS.git
[git@new-ftp projects]$ cd SISTEMAS_EMBEBIDOS.git/
[git@new-ftp SISTEMAS_EMBEBIDOS.git]$ git --bare init
Initialized empty Git repository in /home/git/projects/SISTEMAS_EMBEBIDOS.git/
[git@new-ftp SISTEMAS_EMBEBIDOS.git]$ ls -la
total 40
drwxrwxr-x 7 git git 4096 Apr 15 07:51 .
drwxrwxr-x 3 git git 4096 Apr 15 07:49 ..
drwxrwxr-x 2 git git 4096 Apr 15 07:51 branches
-rw-rw-r-- 1 git git  66 Apr 15 07:51 config
-rw-rw-r-- 1 git git  73 Apr 15 07:51 description
-rw-rw-r-- 1 git git  23 Apr 15 07:51 HEAD
drwxrwxr-x 2 git git 4096 Apr 15 07:51 hooks
drwxrwxr-x 2 git git 4096 Apr 15 07:51 info
drwxrwxr-x 4 git git 4096 Apr 15 07:51 objects
drwxrwxr-x 4 git git 4096 Apr 15 07:51 refs
[git@new-ftp SISTEMAS_EMBEBIDOS.git]$
```

- 5- Para probar el repositorio, desde la misma máquina o desde otra máquina cliente, podemos hacer un clone del proyecto con el comando:

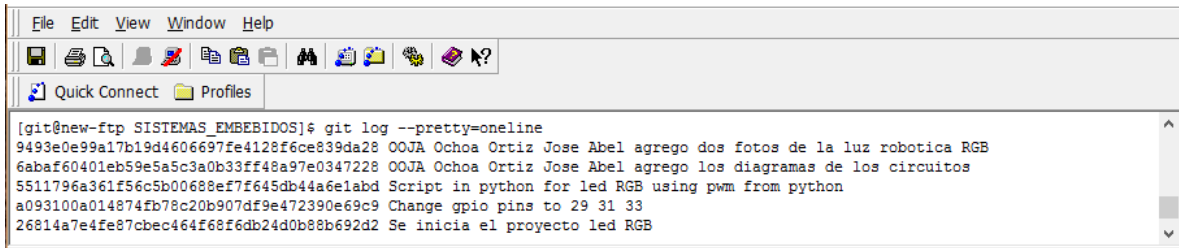


The screenshot shows a terminal window with the same interface as the previous one. The terminal text shows the following commands and output:

```
[git@new-ftp projects]$ git clone git@186.194.140.181:/home/git/projects/SISTEMAS_EMBEBIDOS.git
Cloning into 'SISTEMAS_EMBEBIDOS'...
The authenticity of host '186.194.140.181 (186.194.140.181)' can't be established.
RSA key fingerprint is e4:8c:6c:e4:e5:b9:0f:41:5d:69:57:f9:b9:46:46:3b.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '186.194.140.181' (RSA) to the list of known hosts.
reverse mapping checking getaddrinfo for 186-194-140-181.kionetworks.com failed - POSSIBLE BREAK-IN ATTEMPT!
git@186.194.140.181's password:
remote: Counting objects: 40, done.
remote: Compressing objects: 100% (38/38), done.
remote: Total 40 (delta 6), reused 0 (delta 0)
Receiving objects: 100% (40/40), 1.83 MiB, done.
Resolving deltas: 100% (6/6), done.
[git@new-ftp projects]$
```

Podemos analizar los cambios realizados en el repositorio mediante el comando:

```
git log --pretty=oneline
```



```
[git@new-ftp SISTEMAS_EMBEBIDOS]$ git log --pretty=oneline
9493e0e99a17b19d4606697fe4128f6ce839da28 OOJA Ochoa Ortiz Jose Abel agrego dos fotos de la luz robotica RGB
6abaf60401eb59e5a5c3a0b33ff48a97e0347228 OOJA Ochoa Ortiz Jose Abel agrego los diagramas de los circuitos
5511796a361f56c5b00688ef7f645db44a6e1abd Script in python for led RGB using pwm from python
a093100a014874fb78c20b907df9e472390e69c9 Change gpio pins to 29 31 33
26814a7e4fe87cbec464f68f6db24d0b88b692d2 Se inicia el proyecto led RGB
```

## Aplicación *RoboticLedLight*

Comenzamos a desarrollar una aplicación en la que se involucró el desarrollo de software, la implementación de hardware, así como el manejo de control de versiones mediante Git en un repositorio propio que se encuentra en línea.

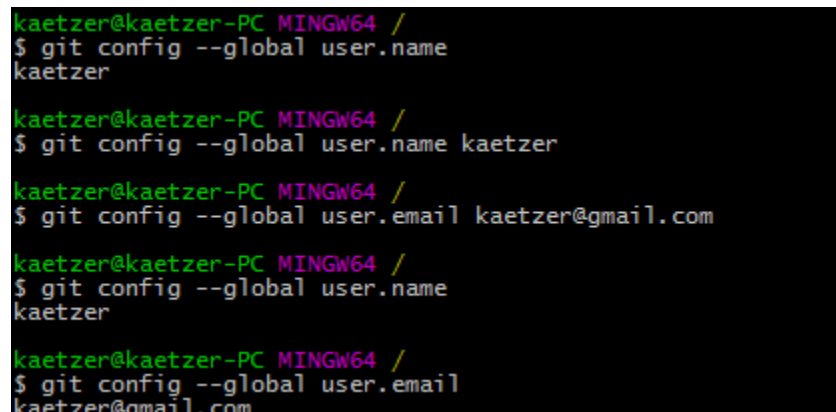
Se realizaron las siguientes ramificaciones en las que cada uno deposito lo que se aportó en el desarrollo de este embebido.



```
alberto@SEMHUNT-DEBIAN: ~/projects/SISTEMAS_EMBEBIDOS
File Edit View Search Terminal Help
alberto@SEMHUNT-DEBIAN:~/projects/SISTEMAS_EMBEBIDOS$ git che
checkout      cherry      cherry-pick
alberto@SEMHUNT-DEBIAN:~/projects/SISTEMAS_EMBEBIDOS$ git checkout
development   dv0.0.1      HEAD          ORIG_HEAD      origin/diagramasAbel  origin/master
diagramasAbel FETCH_HEAD  master        origin/development  origin/HEAD
alberto@SEMHUNT-DEBIAN:~/projects/SISTEMAS_EMBEBIDOS$ git checkout
```

A continuación, se detalla el desarrollo de la aplicación realizando lo siguiente:

Se configuran las variables globales del repositorio para estar identificados en el repositorio.



```
kaetzer@kaetzer-PC MINGW64 /
$ git config --global user.name
kaetzer

kaetzer@kaetzer-PC MINGW64 /
$ git config --global user.name kaetzer

kaetzer@kaetzer-PC MINGW64 /
$ git config --global user.email kaetzer@gmail.com

kaetzer@kaetzer-PC MINGW64 /
$ git config --global user.name
kaetzer

kaetzer@kaetzer-PC MINGW64 /
$ git config --global user.email
kaetzer@gmail.com
```

Se clona el repositorio con el siguiente comando

```
MINGW64:/  
kaetzer@kaetzer-PC MINGW64 /  
$ git clone git@186.194.140.181:/home/git/projects/SISTEMAS_EMBEBIDOS.git
```

Se trabajaron los primeros prototipos para el proyecto módulo de espacio de usuario

```
MINGW64:/c/Users/kaetzer/SISTEMAS_EMBEBIDOS  
Se inicia el proyecto led RGB  
# Please enter the commit message for your changes. Lines starting  
# with '#' will be ignored, and an empty message aborts the commit.  
# On branch master  
#  
# Initial commit  
#  
# Changes to be committed:  
#   new file:   bash/GPIO.sh  
#   new file:   bash/LEEME.TXT  
#   new file:   bash/runRGB.sh  
#   new file:   c/LEEME.TXT  
#   new file:   c/runRGB.c  
#  
~  
~
```

Se buscan los cambios que existen en el repositorio remoto

```
MINGW64:/c/Users/kaetzer/SISTEMAS_EMBEBIDOS  
kaetzer@kaetzer-PC MINGW64 ~/SISTEMAS_EMBEBIDOS (master)  
$ ls  
bash/ c/  
kaetzer@kaetzer-PC MINGW64 ~/SISTEMAS_EMBEBIDOS (master)  
$ git fetch origin  
git@186.194.140.181's password:  
remote: Counting objects: 7, done.  
remote: Compressing objects: 100% (3/3), done.  
remote: Total 4 (delta 1), reused 0 (delta 0)  
Unpacking objects: 100% (4/4), done.  
From 186.194.140.181:/home/git/projects/SISTEMAS_EMBEBIDOS  
* [new branch]      development -> origin/development  
* [new tag]         dv0.0.1    -> dv0.0.1  
kaetzer@kaetzer-PC MINGW64 ~/SISTEMAS_EMBEBIDOS (master)  
$ |
```

Damos pull para actualizar nuestro repositorio local

```
kaetzer@kaetzer-PC MINGW64 ~/SISTEMAS_EMBEBIDOS (master)  
$ git pull  
git@186.194.140.181's password:  
Updating 26814a7..5511796  
Fast-forward  
 bash/runRGB.sh | 6 +++--  
python/leds.py | 31 ++++++  
2 files changed, 34 insertions(+), 3 deletions(-)  
create mode 100644 python/leds.py
```

Creamos una rama de desarrollo

```
kaetzer@kaetzer-PC MINGW64 ~/SE/SISTEMAS_EMBEBIDOS/python (master)
$ git branch development
```

Cambiamos de rama a desarrollo

```
MINGW64:/c/Users/kaetzer/SE/SISTEMAS_EMBEBIDOS/python

kaetzer@kaetzer-PC MINGW64 ~/SE/SISTEMAS_EMBEBIDOS/python (master)
$ git branch
  development
* master
```

Buscamos en log para hacer un cherry-pick a la rama de desarrollo para continuar los cambios de la rama master

```
MINGW64:/c/Users/kaetzer/SE/SISTEMAS_EMBEBIDOS/python

kaetzer@kaetzer-PC MINGW64 ~/SE/SISTEMAS_EMBEBIDOS/python (master)
$ git branch
  development
* master

kaetzer@kaetzer-PC MINGW64 ~/SE/SISTEMAS_EMBEBIDOS/python (master)
$ git log
commit 5511796a361f56c5b00688ef7f645db44a6e1abd
Author: David Reyes <david.proteco@gmail.com>
Date:   Sun Apr 10 21:52:48 2016 -0500

    Script in python for led RGB using pwm from python

commit a093100a014874fb78c20b907df9e472390e69c9
Author: David Reyes <david.proteco@gmail.com>
Date:   Sat Apr 9 18:54:29 2016 -0500

    Change gpio pins to 29 31 33

commit 26814a7e4fe87cbec464f68f6db24d0b88b692d2
Author: kaetzer <kaetzer@gmail.com>
Date:   Sat Apr 9 00:25:45 2016 -0500

    Se inicia el proyecto led RGB

kaetzer@kaetzer-PC MINGW64 ~/SE/SISTEMAS_EMBEBIDOS/python (master)
$ git checkout development
Switched to branch 'development'

kaetzer@kaetzer-PC MINGW64 ~/SE/SISTEMAS_EMBEBIDOS/python (development)
$ git branch
* development
  master

kaetzer@kaetzer-PC MINGW64 ~/SE/SISTEMAS_EMBEBIDOS/python (development)
$ git cherry-pick 5511796a361f56c5b00688ef7f645db44a6e1abd
On branch development
You are currently cherry-picking commit 5511796.

nothing to commit, working directory clean
The previous cherry-pick is now empty, possibly due to conflict resolution.
If you wish to commit it anyway, use:
```



Hacemos el commit de la rama de desarrollo

```
MINGW64:/c/Users/kaetzer/SE/SISTEMAS_EMBEBIDOS/python
Script in python for led RGB using pwm from python

#
# It looks like you may be committing a cherry-pick.
# If this is not correct, please remove the file
#   .git/CHERRY_PICK_HEAD
# and try again.

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit
#
# Author:      David Reyes <david.proteco@gmail.com>
# Date:       Sun Apr 10 21:52:48 2016 -0500
#
# On branch development
# You are currently cherry-picking commit 5511796.
#
# Changes to be committed:
#   modified:   leds.py
#
```

Mandamos nuestros cambios al repositorio remoto

```
MINGW64:/c/Users/kaetzer/SE/SISTEMAS_EMBEBIDOS
kaetzer@kaetzer-PC MINGW64 ~/SE/SISTEMAS_EMBEBIDOS (master)
$ git push origin development
git@186.194.140.181's password:
Counting objects: 4, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 1.04 KiB | 0 bytes/s, done.
Total 4 (delta 1), reused 0 (delta 0)
To git@186.194.140.181:/home/git/projects/SISTEMAS_EMBEBIDOS.git
 * [new branch]      development -> development
kaetzer@kaetzer-PC MINGW64 ~/SE/SISTEMAS_EMBEBIDOS (master)
```

Creamos el primer tag de la primera versión de nuestro código de desarrollo

```
kaetzer@kaetzer-PC MINGW64 ~/SE/SISTEMAS_EMBEBIDOS (development)
$ git tag dv0.0.1

kaetzer@kaetzer-PC MINGW64 ~/SE/SISTEMAS_EMBEBIDOS (development)
$ git estatus
git: 'estatus' is not a git command. See 'git --help'.

Did you mean this?
    status

kaetzer@kaetzer-PC MINGW64 ~/SE/SISTEMAS_EMBEBIDOS (development)
$ git status
On branch development
nothing to commit, working directory clean

kaetzer@kaetzer-PC MINGW64 ~/SE/SISTEMAS_EMBEBIDOS (development)
$ git push origin --tags
git@186.194.140.181's password:
Total 0 (delta 0), reused 0 (delta 0)
To git@186.194.140.181:/home/git/projects/SISTEMAS_EMBEBIDOS.git
 * [new tag]         dv0.0.1 -> dv0.0.1

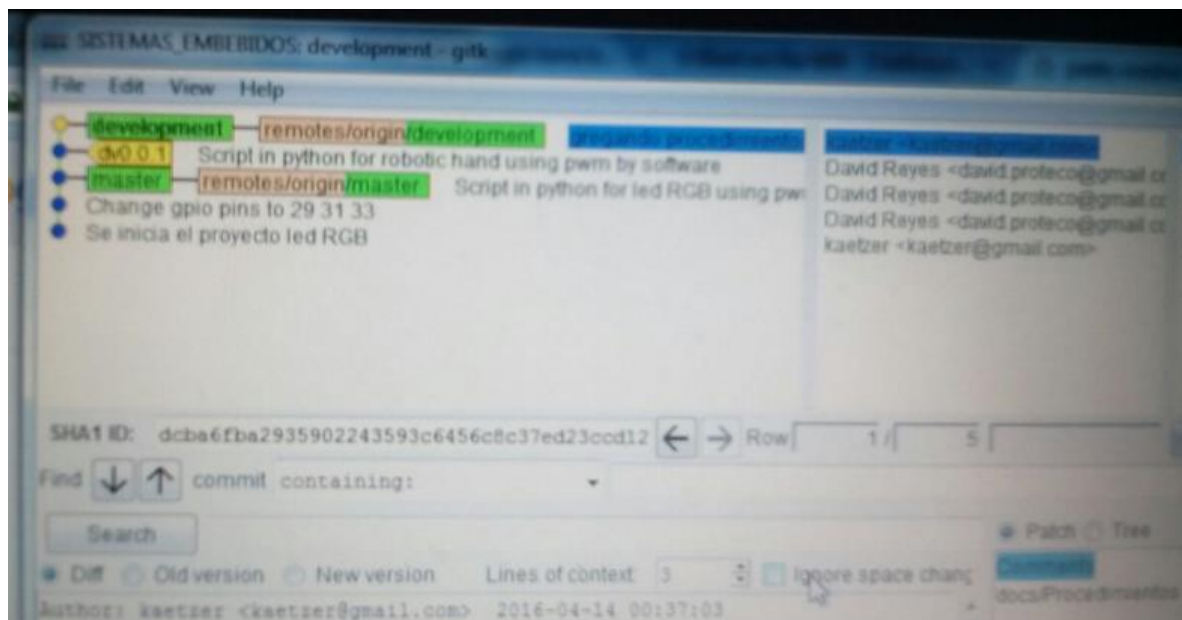
kaetzer@kaetzer-PC MINGW64 ~/SE/SISTEMAS_EMBEBIDOS (development)
$
```

## Problemas

Durante el desarrollo de esta tarea, nos encontramos con una situación en la que al tratar de hacer un pull, este marcaba el siguiente error, indicando que se tenía que hacer un commit antes de hacer el pull, esto fue ocasionado debido a que en el repositorio (servidor) se estaban realizando cambios.

```
alberto@SEMHUNT-DEBIAN:~/Projects/SISTEMAS_EMBEBIDOS$ sudo chmod -R 777 .git/
alberto@SEMHUNT-DEBIAN:~/Projects/SISTEMAS_EMBEBIDOS$ ls -la
total 28
drwxr-xr-x 7 root root 4096 Apr 15 00:22 .
drwxr-xr-x 3 root root 4096 Apr 15 00:22 ..
drwxrwxrwx 2 root root 4096 Apr 15 00:22 [REDACTED]
drwxrwxrwx 2 root root 4096 Apr 15 00:22 [REDACTED]
drwxrwxrwx 2 root root 4096 Apr 15 00:22 [REDACTED]
drwxrwxrwx 8 root root 4096 Apr 15 00:22 [REDACTED]
drwxrwxrwx 2 root root 4096 Apr 15 00:22 [REDACTED]
alberto@SEMHUNT-DEBIAN:~/Projects/SISTEMAS_EMBEBIDOS$ git pull origin development
git@186.194.140.181's password:
From 186.194.140.181:/home/git/projects/SISTEMAS_EMBEBIDOS
 * branch                development -> FETCH_HEAD
error: Your local changes to the following files would be overwritten by merge:
       python/leds.py
Please, commit your changes or stash them before you can merge.
Aborting
alberto@SEMHUNT-DEBIAN:~/Projects/SISTEMAS_EMBEBIDOS$ git merge origin/development
error: Your local changes to the following files would be overwritten by merge:
       python/leds.py
Please, commit your changes or stash them before you can merge.
Aborting
alberto@SEMHUNT-DEBIAN:~/Projects/SISTEMAS_EMBEBIDOS$
```

Para solucionar este incidente se debió acceder al repositorio y concluir los cambios para que en los equipos (usuarios) se pudiera continuar con el desarrollo.



## Conclusión

Hemos visto la importancia de mantener un versionado ya que podemos tener un control total y absoluto de los avances, cambios, mejoras, updates, etc. Git tiene una serie de otros beneficios que listaremos a continuación:

- Los repositorios y ramas Git son mucho más pequeños.
- Las ramas Git llevan toda su historia completa.
- Los formatos de archivo de los repositorios son simples, la reparación es fácil y la corrupción es rara.
- El clonado de repositorios actúa como repositorio de copias de seguridad completas.
- Proporciona una mejor auditoría de eventos de ramificación (branch) y fusión (merge).