

Percepción

Git y Github para poetas, parte 4

2 de septiembre de 2018

Víctor Medrano Zarazúa

`victor.medranozr@uanl.edu.mx`

Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

Contenido

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

Terminal

Git

Tarea

Información de
contacto

Percepción

Víctor Medrano
Zarazúa

2 Repaso

Introducción

Terminal

Git

Tarea

Información de
contacto

Recapítulemos...

- ▶ ¿Qué es un issue?
- ▶ ¿Para qué sirve un issue?
- ▶ ¿De qué plataforma forma parte el concepto de issue?
- ▶ ¿Qué es Markdown?

Percepción

Víctor Medrano
Zarazúa

Repaso

3 Introducción

Terminal

Git

Tarea

Información de
contacto

Git y la terminal

- ▶ Veremos cómo mantener un control de versiones con archivos almacenados localmente en nuestra computadora.
- ▶ Usaremos Git de forma offline y sin necesidad de acceder al sitio de Github.
- ▶ Veremos los comandos principales a ejecutar en una terminal y haremos uso de comandos de Git para administrar repositorios.

```

Sep 15:51 *
0. Sep 2015  bin -> usr/bin
01. Sep 09:31 boot
01. Sep 15:50 dev
09. Sep 09:32 etc
21. Sep 15:52 home
30. Sep 2015  lib -> usr/lib
30. Sep 2015  lib64 -> usr/lib
04 23. Jul 10:01 lost+found
06 1. Aug 22:45 mnt
06 30. Sep 2015 opt
06 21. Sep 15:52 private -> /home/encrypted
06 12. Aug 08:15 proc
06 21. Sep 15:52 root
06 7 30. Sep 2015 run
06 30. Sep 2015
```

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

4 Terminal

Git

Tarea

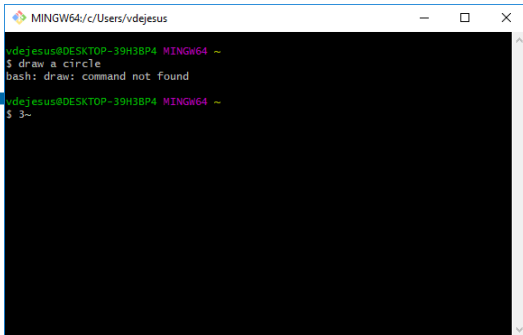
Información de
contacto

Según tu sistema operativo...

- ▶ Si eres usuario de Mac o Linux, puedes hacer uso de la terminal que ya está incluida en el mismo sistema operativo.
- ▶ Si eres usuario de Windows se recomienda bajar el Git bash haciendo clic en el [enlace](#), ya que usaremos comandos de Unix/Linux.

¿Qué es el bash/terminal?

Es un intérprete de comandos que ejecuta tareas específicas dentro del sistema operativo.



```
MINGW64: c:/Users/vdejesus

vdejesus@DESKTOP-39H3BP4 MINGW64 ~
$ draw a circle
bash: draw: command not found

vdejesus@DESKTOP-39H3BP4 MINGW64 ~
$ 3~
$ ~
```

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

5 Terminal

Git

Tarea

Información de
contacto

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

6 Terminal

Git

Tarea

Información de
contacto

- ▶ cd (cambia de directorio)
- ▶ pwd (print working directory, muestra el directorio actual)
- ▶ ls (hace una lista de los archivos dentro del directorio actual)
- ▶ clear (limpia la terminal)
- ▶ help (muestra ayuda para diferentes comandos)

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

7 Terminal

Git

Tarea

Información de
contacto

- ▶ Para regresar un directorio hacia arriba hacemos uso del comando `cd ..`
- ▶ La tecla *TAB* sirve para evitar teclear el nombre completo de un archivo o directorio. Nos ahorra tiempo al navegar por el sistema de archivos.
- ▶ Las teclas *y* *n* nos permiten navegar a través de los comandos previos que hemos introducido en la terminal.
- ▶ Podemos arrastrar directorios hacia la terminal para obtener de forma automática la ruta que poseen en el sistema de archivos.
- ▶ Los comandos de Unix/Linux frecuentemente son modificados con argumentos (e.g: `ls -all`)

Git ready...

Asegurando que la instalación de Git se haya hecho correctamente, teclearemos el comando **git**.

```
MINGW64/c/Users/vdejesu
vdejesu@DESKTOP-99H3BP4 MINGW64 ~
$ git
usage: git [--version] [--help] [-C <path>] [-c name=value]
          [--exec-path=<path>] [--html-path] [--man-path] [--info-path]
          [-p | --paginate | --no-pager] [--no-replace-objects] [--bare]
          [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
          <command> [<args>]

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
  clone      Clone a repository into a new directory
  init       Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
  add        Add file contents to the index
  mv         Move or rename a file, a directory, or a symlink
  reset      Reset current HEAD to the specified state
  rm         Remove files from the working tree and from the index

examine the history and state (see also: git help revisions)
  bisect     Use binary search to find the commit that introduced a bug
  grep       Print lines matching a pattern
  log        Show commit logs
  show       Show various types of objects
  status     Show the working tree status

grow, mark and tweak your common history
  branch     List, create, or delete branches
  checkout   Switch branches or restore working tree files
  commit     Record changes to the repository
  diff       Show changes between commits, commit and working tree, etc
  merge      Join two or more development histories together
  rebase     Reapply commits on top of another base tip
  tag        Create, list, delete or verify a tag object signed with GPG

collaborate (see also: git help workflows)
  fetch      Download objects and refs from another repository
  pull       Fetch from and integrate with another repository or a local branch
  push       Update remote refs along with associated objects
```

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

Terminal

8 Git

Tarea

Información de
contacto

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

Terminal

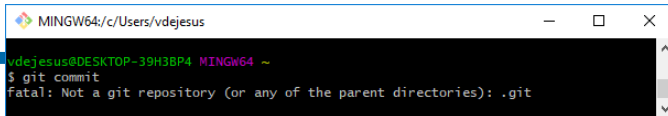
9 Git

Tarea

Información de
contacto

Not a git repository...

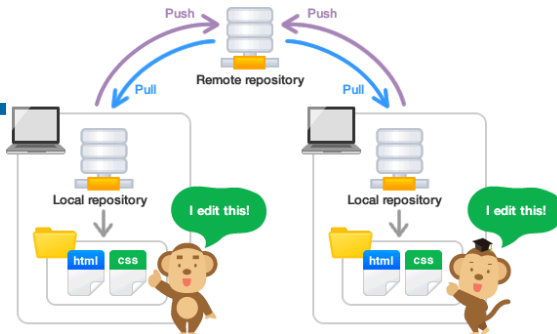
Intentando hacer un commit. Pero el esfuerzo es infructuoso :(



```
MINGW64:/c/Users/vdejesus
vdejesus@DESKTOP-39H3BP4 MINGW64 ~
$ git commit
fatal: Not a git repository (or any of the parent directories): .git
```

Repositorios locales

Anteriormente estuvimos trabajando con repositorios alojados en el sitio web de Github. Ahora veremos como crear y administrar estos repositorios de forma local en nuestra PC.



Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

Terminal

100 Git

Tarea

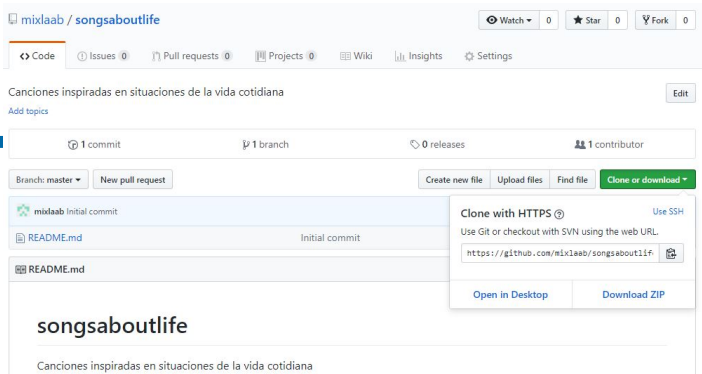
Información de
contacto

It's a new dawn, it's a new day, it's a new life...

- ▶ La forma más sencilla de empezar es hacer un repositorio en Github y después “descargarlo” en nuestra computadora. A este proceso de “descargar” un repositorio lo llamaremos formalmente como **clone** (primera vez) o **pull** (actualizar repositorio).
- ▶ Se le llama clonar a la acción de tomar un repositorio remoto y almacenarlo como una copia en un repositorio local.
- ▶ Se pueden hacer cambios en el repositorio local y hacer **push** en el repositorio remoto para actualizarlo. O viceversa, puede haber cambios en el repositorio remoto y haremos **pull** para actualizar el repositorio local.

Clonando repositorio

Se crea un repositorio en Github y se copia el enlace del mismo como muestra la imagen de abajo.



The screenshot shows the GitHub interface for the repository 'mixlaab / songsaboutlife'. At the top, there are buttons for 'Watch', 'Star', and 'Fork', each with a count of 0. Below this is a navigation bar with links for 'Code', 'Issues', 'Pull requests', 'Projects', 'Wiki', 'Insights', and 'Settings'. The main content area shows the repository name 'songsaboutlife' and a description 'Canciones inspiradas en situaciones de la vida cotidiana'. There is an 'Edit' button next to the description. Below the description, it shows '1 commit', '1 branch', '0 releases', and '1 contributor'. A 'Clone or download' button is visible, which has opened a modal dialog. The modal dialog has two tabs: 'Clone with HTTPS' (selected) and 'Use SSH'. The 'Clone with HTTPS' tab shows the URL 'https://github.com/mixlaab/songsaboutlife/' and a 'Copy' icon. Below the URL are two buttons: 'Open in Desktop' and 'Download ZIP'. The modal also shows a list of files: 'Initial commit', 'README.md', and 'README.md'.

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

Terminal

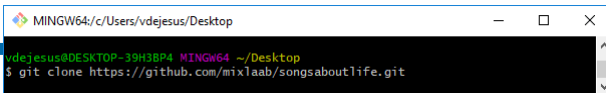
12 Git

Tarea

Información de
contacto

Clonando repositorio

Hacemos uso del comando git clone.



```
MINGW64:/c:/Users/vdejesus/Desktop
vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop
$ git clone https://github.com/mixlaab/songsaboutlife.git
```

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

Terminal

13 Git

Tarea

Información de
contacto

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

Terminal

14 Git

Tarea

Información de
contacto

Clonando repositorio

```
MINGW64:/c:/Users/vdejesus/Desktop
vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop
$ git clone https://github.com/mixlaab/songsaboutlife.git
Cloning into 'songsaboutlife'...
remote: Counting objects: 3, done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
```

Clonando repositorio

Usando el comando `ls`.

```

MINGW64:/c:/Users/vdejesus/Desktop
vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop
$ ls
'~$aneacion_PP.docx'      LaserWorkV5/
'~$Edades.xlsx'          LaserWorkV5.lnk*
'~$RoboticsClubUVM.pptx'  laureate.txt
'~$torial SW.docx'        Materiales_Rbt.xlsx
'~$xamen3.docx'           MATLAB.lnk*
'Addictive Keys.lnk'*     MEGAsync.lnk*
Axtel.txt                 'Microsoft Edge.lnk'*
ball.png                  'mil años.mp3'
bin/                       'MinGW Installer.lnk'*
cel.txt                   ourRobot.jpg
CFDMP_discoColibri3.gcode ourRobot3.png
CH341SER.EXE*             piezaUVM.stl
Colibri3D/                poo_parcial3.pdf
desktop.ini               presupuesto_pt1.pdf
diana.gcode               presupuesto_pt2.pdf
Dropbox.lnk*              'processing - Acceso directo.lnk'*
Gantt_Chart_v2.pdf        processing-3.3.5-windows64/
Gantt_parte1.pdf          SolidWorks/
Gantt_parte1.xlsx         songsaboutlife/
Gantt_parte2.pdf          Spotify.lnk*
Gantt_parte2.xlsx         'uTorrent Web.lnk'*
imageProcessing.py        vdejesus/
javi/                     Win10_1607_SpanishMexico_x64.iso
JuntaUVM2018.txt
  
```

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

Terminal

15 Git

Tarea

Información de
contacto

Status del repositorio

Usando el comando git status. Debemos estar dentro del directorio del repositorio para que el comando funcione.

```
MINGW64:/c:/Users/vdejesus/Desktop/songsaboutlife
vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop
$ git status
fatal: Not a git repository (or any of the parent directories): .git

vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop
$ cd songsaboutlife/

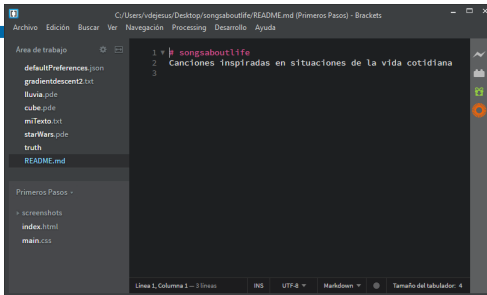
vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
```

Modificando archivos de forma local

Necesitaremos un editor de código. En lo personal recomiendo alguno de los siguientes:

- ▶ Visual Studio Code
- ▶ Brackets
- ▶ Sublime Text



Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

Terminal

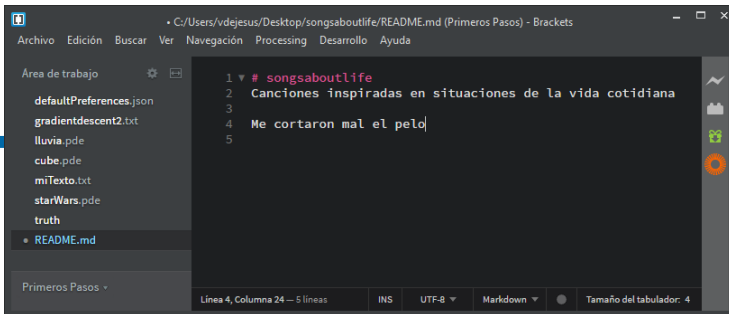
170 Git

Tarea

Información de
contacto

Modificando archivos de forma local

Modificamos el archivo y lo guardamos.



```

1 # songsaboutlife
2 Canciones inspiradas en situaciones de la vida cotidiana
3
4 Me cortaron mal el pelo
5
  
```

Linea 4, Columna 24 — 5 líneas INS UTF-8 Markdown Tamaño del tabulador: 4

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

Terminal

18 Git

Tarea

Información de
contacto

Modificando archivos de forma local

Checamos el status del repositorio y veremos que hay una modificación en el archivo README.md

```
MINGW64:/c:/Users/vdejesus/Desktop/songsaboutlife

vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (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 checkout -- <file>..." to discard changes in working directory)

        modified:   README.md

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

vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$
```

Haciendo push al repositorio remoto de Github

Sin embargo aún no hay cambios en el repositorio remoto. Para lograr traspasar los cambios al repositorio de Github comenzaremos usando el comando `git commit -a`. Es necesario haber accedido previamente los datos de nuestra cuenta, de lo contrario sucederá lo siguiente...

```

MINGW64/c:/Users/vdejesus/Desktop/songsaboutlife
vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git commit -a

*** Please tell me who you are.

Run

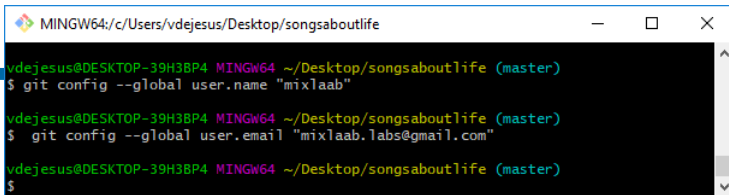
  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'vdejesus@DESKTOP-39H3BP4.(none)')
vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$
    
```

Haciendo push al repositorio remoto de Github

Proporcionamos nuestro nombre de usuario en Github y la cuenta de correo asociada a la cuenta.



```
MINGW64:/c/Users/vdejesus/Desktop/songsaboutlife

vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git config --global user.name "mixlaab"

vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git config --global user.email "mixlaab.labs@gmail.com"

vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$
```

Haciendo push al repositorio remoto de Github

Verificamos que los datos introducidos sean correctos con `git config --list`.

```

MINGW64: c:/Users/vdejesus/Desktop/songsaboutlife
vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git config --list
core.symlinks=false
core.autocrlf=true
core.fscache=true
color.diff=auto
color.status=auto
color.branch=auto
color.interactive=true
help.format=html
rebase.autosquash=true
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
http.sslbackend=openssl
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
credential.helper=manager
user.name=mixlaab
user.email=mixlaab.labs@gmail.com
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.symlinks=false
core.ignorecase=true
remote.origin.url=https://github.com/mixlaab/songsaboutlife.git
remote.origin.fetch+=refs/heads/*:refs/remotes/origin/*
branch.master.remote=origin
branch.master.merge=refs/heads/master
    
```

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

Terminal

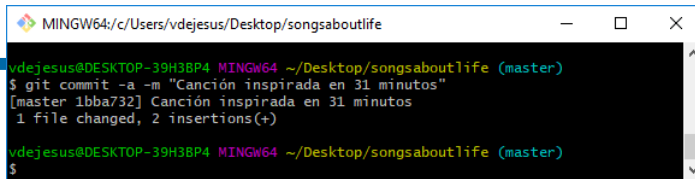
22 Git

Tarea

Información de
contacto

Haciendo push al repositorio remoto de Github

Y ya podremos hacer commit. Además podremos agregar un título para ese commit utilizando el argumento **-m**.



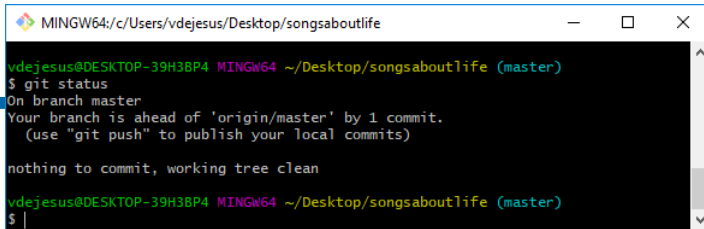
```
MINGW64:/c:/Users/vdejesus/Desktop/songsaboutlife

vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git commit -a -m "Canción inspirada en 31 minutos"
[master 1bba732] Canción inspirada en 31 minutos
1 file changed, 2 insertions(+)

vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$
```


Haciendo push al repositorio remoto de Github

Si vemos el status del repositorio aparecerá lo siguiente.



```
MINGW64:/c/Users/vdejesus/Desktop/songsaboutlife

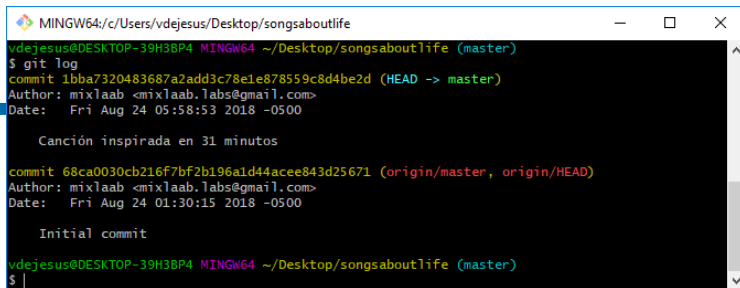
vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)

nothing to commit, working tree clean

vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ |
```

Haciendo push al repositorio remoto de Github

Podemos ver un historial de los commits haciendo uso del comando `git log`.



```
MINGW64:/c:/Users/vdejesus/Desktop/songsaboutlife
vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git log
commit 1bba7320483687a2add3c78e1e878559c8d4be2d (HEAD -> master)
Author: mixlaab <mixlaab.labs@gmail.com>
Date:   Fri Aug 24 05:58:53 2018 -0500

    Canción inspirada en 31 minutos

commit 68ca0030cb216f7bf2b196a1d44acee843d25671 (origin/master, origin/HEAD)
Author: mixlaab <mixlaab.labs@gmail.com>
Date:   Fri Aug 24 01:30:15 2018 -0500

    Initial commit

vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ |
```

Haciendo push al repositorio remoto de Github

También podemos ver una lista de los servidores remotos asociados al repositorio con el comando `git remote` antes de hacer push.

```
MINGW64:/c:/Users/vdejesus/Desktop/songsaboutlife

vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git push
Logon failed, use ctrl+c to cancel basic credential prompt.
Username for 'https://github.com':
vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git remote
origin

vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git remote -v
origin https://github.com/mixlaab/songsaboutlife.git (fetch)
origin https://github.com/mixlaab/songsaboutlife.git (push)

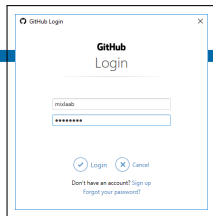
vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$
```

Haciendo push al repositorio remoto de Github

Finalmente hago push al repositorio remoto (**origin**) en la rama pertinente (en este caso, **master**). Es necesario introducir usuario y contraseña del servidor remoto.

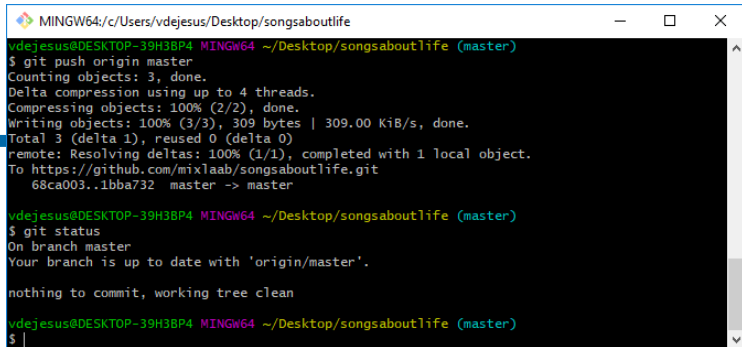
```
MINGW64~/Users/vdejesus/Desktop/songsaboutlife
vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git remote -v
origin https://github.com/mixlaab/songsaboutlife.git (fetch)
origin https://github.com/mixlaab/songsaboutlife.git (push)

vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git push origin master
Logon failed, use ctrl+c to cancel basic credential prompt.
Username for 'https://github.com': mixlaab
error: unable to read aspass response from 'C:/Program Files/Git/mingw64/libexec/git-core/git-gui--askpass'
Password for 'https://mixlaab@github.com':
vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git push origin master
```



Haciendo push al repositorio remoto de Github

Lo hicimos. Hemos actualizado el repositorio remoto.



```
MINGW64:/c:/Users/vdejesus/Desktop/songsaboutlife
vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git push origin master
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 309 bytes | 309.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/mixlaab/songsaboutlife.git
 68ca003..1bba732 master -> master

vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean

vdejesus@DESKTOP-39H3BP4 MINGW64 ~/Desktop/songsaboutlife (master)
$ |
```

Haciendo push al repositorio remoto de Github

Canciones inspiradas en situaciones de la vida cotidiana

[Add topics](#)

Edit

2 commits

1 branch

0 releases

1 contributor

Branch: master


New pull request

Create new file

Upload files

Find file

Clone or download

 mixlaab Canción inspirada en 31 minutos

Latest commit 1bba732 an hour ago

 README.md

Canción inspirada en 31 minutos

an hour ago

 README.md



songsaboutlife

Canciones inspiradas en situaciones de la vida cotidiana

Me cortaron mal el pelo

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

Terminal

29 Git

Tarea

Información de
contacto

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

Terminal

Git

30 Tarea

Información de
contacto

Tarea #4 (Individual)

- ▶ Crear un repositorio en el sitio de Github (debe contener archivo README.md). Recuerda poner una descripción a tu repositorio.
 - ▶ Clonar el repositorio remoto en el repositorio local (tu PC).
 - ▶ Modificar el archivo README.md localmente por medio de un editor de texto.
-
- ▶ Configurar nombre de usuario y correo antes de hacer commit.
 - ▶ Hacer commit junto con un mensaje usando el argumento **-m**.
 - ▶ Hacer push al repositorio remoto (recuerda que te pedirá nombre de usuario y contraseña de Github).

En caso de comentarios, sugerencias, preguntas o errores en las diapositivas no dudes en contactarme.

Víctor Medrano Zarazúa
victor.medranozr@uanl.edu.mx
<https://mixlaab.github.io>
WA: 8119022700

Percepción

Víctor Medrano
Zarazúa

Repaso

Introducción

Terminal

Git

Tarea

31 Información de
contacto

Fin