



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA



PROGRAMACION ORIENTADO A OBJETOS TRABAJO EN CASA

PROCEDIMIENTO DE TRABAJO
CON LOS CODIGOS DE GIT

NOMBRE: PABLO ANDRES TIPAN CHANCUSIG

CURSO: SEGUNDO "A"

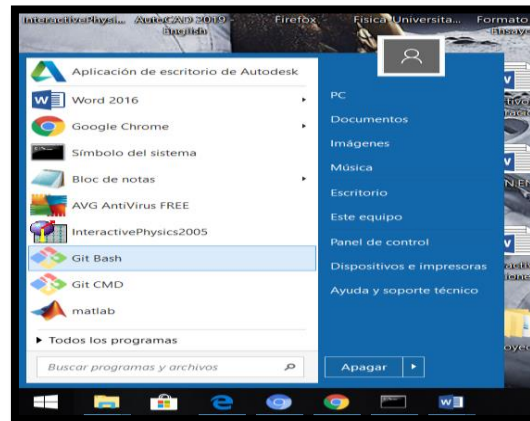
DOCENTE: Ing. NANCY JACHO

NRC:

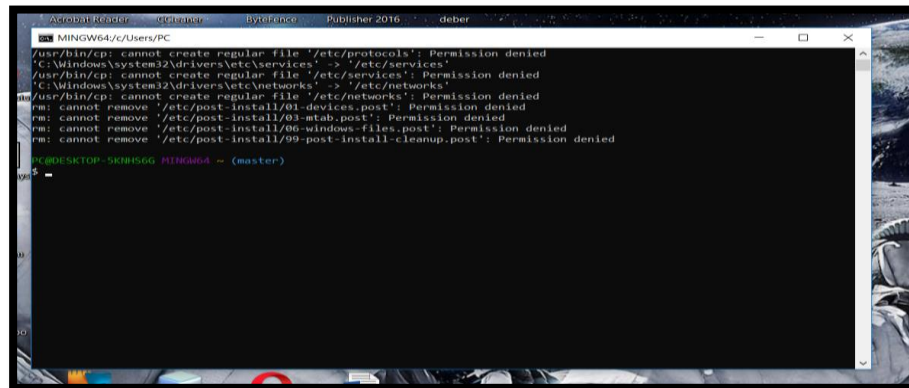
BIBLIOGRAFIA: www.hostinger.es

PERIODO ACADEMICO: OCT.2018-FEB.2019

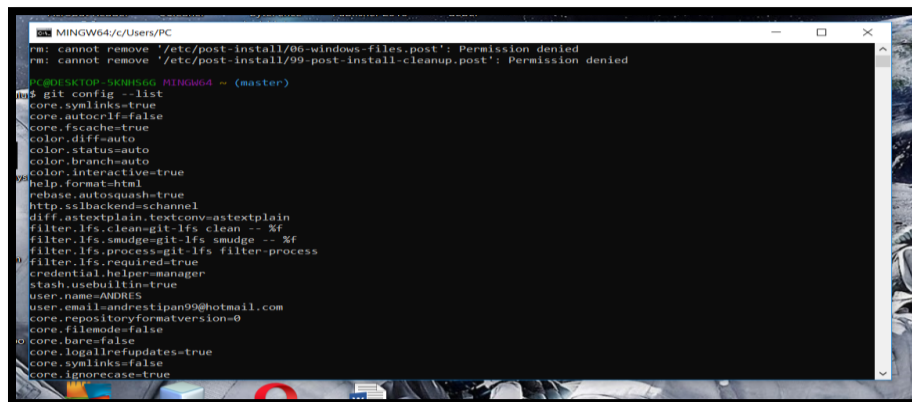
1. Abrir el GIT BASH que se encuentra en la ventana de Windows o también podemos acceder mediante la opción de búsqueda.



2. Inmediatamente aparecerá una ventana como la que se muestra a continuación en la cual empezaremos a efectuar los códigos y el proceso de guardado de proyectos.



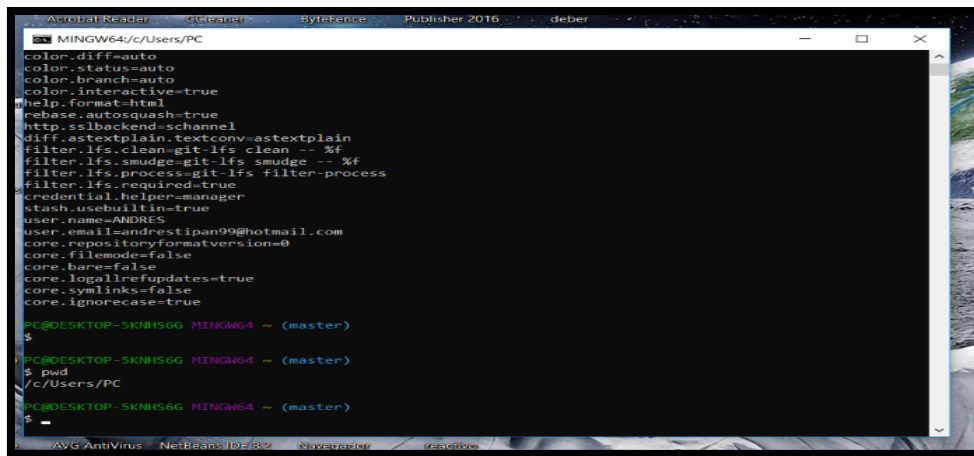
3. Insertamos el código (git config --list) para que nos muestre toda la información acerca de nuestra computadora que nos muestra ciertas características importantes.



En caso de que no nos muestre el nombre de usuario y el correo electrónico tenemos que crear ambas instancias utilizando los siguientes códigos:

```
git config --global user.name "Nombre del usuario": Nombre de usuario
git config --global user.email "correo electrónico": Correo electrónico
```

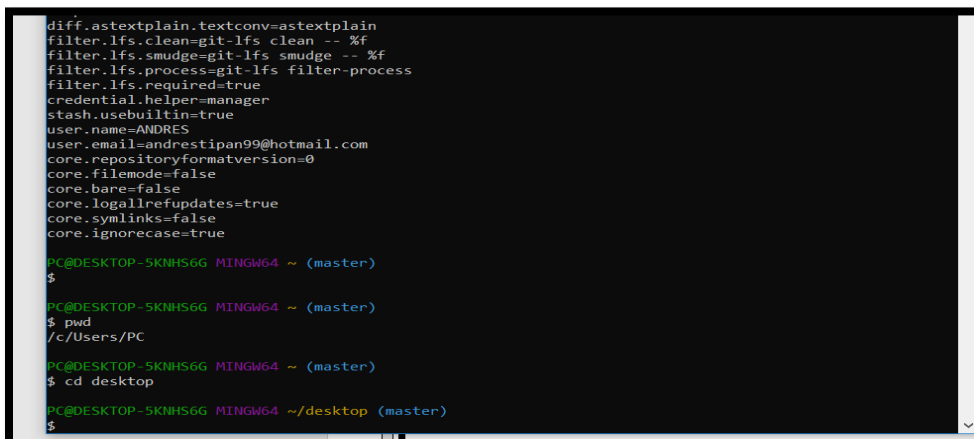
4. Introducimos el código (pwd) que nos visualiza en que carpeta o en qué lugar del computador nos encontramos.



```
MINGW64/c/Users/PC
color.diff=auto
color.status=auto
color.branch=auto
color.interactive=true
help.format=html
rebase.autosquash=true
http.sslbackend=schannel
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
stash.usebuiltin=true
user.name=ANDRES
user.email=andrestipan99@hotmail.com
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.symlinks=false
core.ignorecase=true

PC@DESKTOP-5KNHS6G MINGW64 ~ (master)
$
PC@DESKTOP-5KNHS6G MINGW64 ~ (master)
$ pwd
/c/Users/PC
PC@DESKTOP-5KNHS6G MINGW64 ~ (master)
$
```

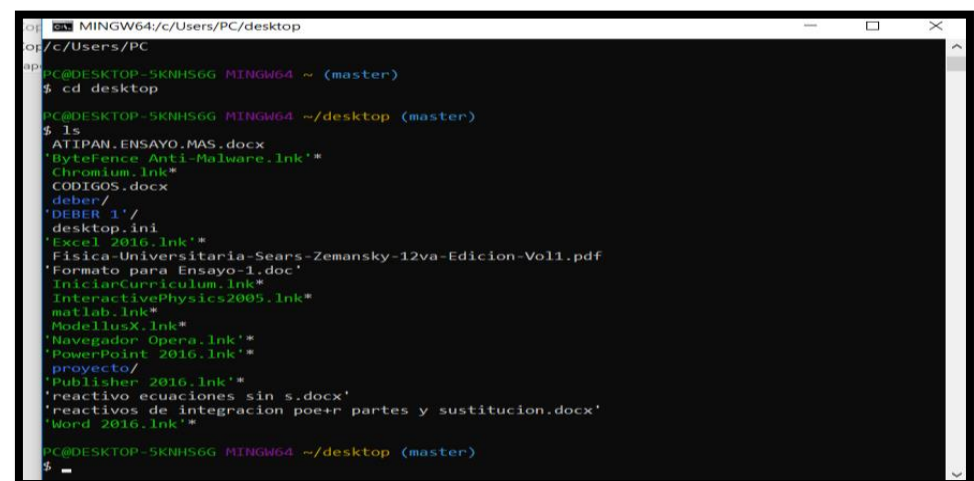
5. Nos dirigimos al escritorio de nuestro computador y para eso insertamos el código (cd desktop). Nos aparecerá lo siguiente:



```
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
stash.usebuiltin=true
user.name=ANDRES
user.email=andrestipan99@hotmail.com
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.symlinks=false
core.ignorecase=true

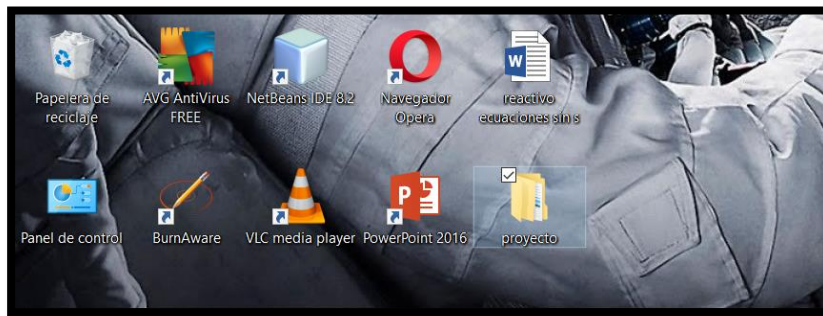
PC@DESKTOP-5KNHS6G MINGW64 ~ (master)
$
PC@DESKTOP-5KNHS6G MINGW64 ~ (master)
$ pwd
/c/Users/PC
PC@DESKTOP-5KNHS6G MINGW64 ~ (master)
$ cd desktop
PC@DESKTOP-5KNHS6G MINGW64 ~/desktop (master)
$
```

6. Insertamos el código (ls) para que nos indique las carpetas y archivos que se encuentran en el escritorio, mostrándonos ciertas características de cada una de estas.



```
MINGW64/c/Users/PC/desktop
op/c/Users/PC
PC@DESKTOP-5KNHS6G MINGW64 ~ (master)
$ cd desktop
PC@DESKTOP-5KNHS6G MINGW64 ~/desktop (master)
$ ls
ATIPAN. ENSAYO.MAS.docx
'ByteFence Anti-Malware.lnk'*
'Chromium.lnk'*
CODIGOS.docx
deber/
'DEBER 1'/'
desktop.ini
'Excel 2016.lnk'*
Fisica-Universitaria-Sears-Zemansky-12va-Edicion-Vol1.pdf
'Formato para Ensayo-1.doc'
'IniciarCurriculum.lnk'*
'InteractivePhysics2009.lnk'*
matlab.lnk*
ModellusX.lnk*
'Navegador Opera.lnk'*
'PowerPoint 2016.lnk'*
proyecto/
'Publisher 2016.lnk'*
'reactivo ecuaciones sin s.docx'
'reactivos de integracion poe+r partes y sustitucion.docx'
'Word 2016.lnk'*
PC@DESKTOP-5KNHS6G MINGW64 ~/desktop (master)
$
```

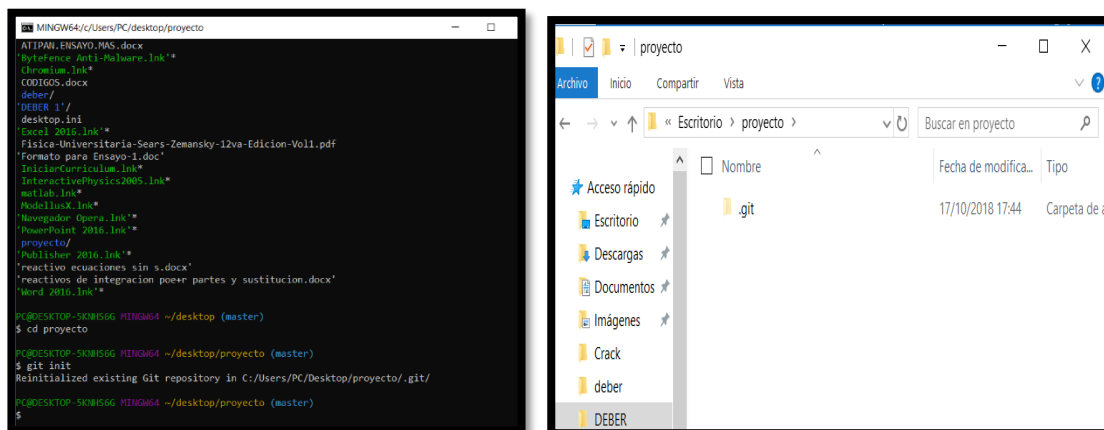
7. En nuestro escritorio creamos una carpeta con el nombre proyecto.



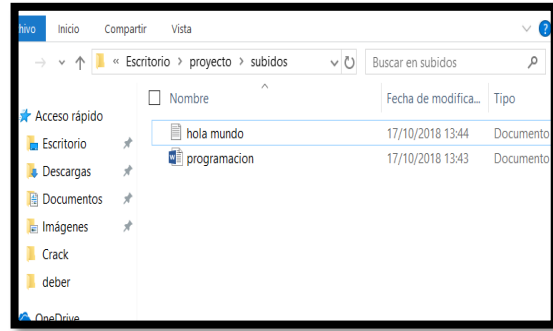
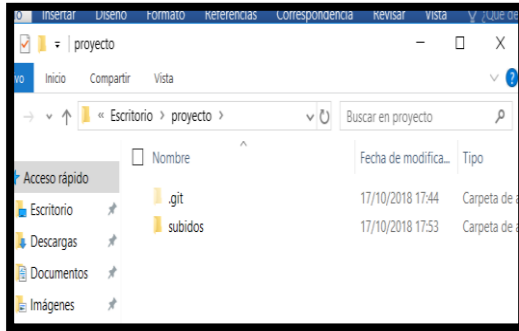
8. Ahora nos dirigimos nuevamente a la ventana de códigos git y vamos a introducir el código (cd proyecto) para que nos dirija hasta esta carpeta.

```
MINGW64/c/Users/PC/desktop/proyecto
$ cd desktop
PC@DESKTOP-SKNH56G MINGW64 ~/desktop (master)
$ ls
ATIPAN.ENSAYO.MAS.docx
ByteFence Anti-Malware.lnk*
Chromium.lnk*
CODIGOS.docx
deber/
DEBER 1' /
desktop.ini
Excel 2016.lnk*
Fisica-Universitaria-Sears-Zemansky-12va-Edicion-Vol1.pdf
Formato para Ensayo-1.docx
IniciarCurriculum.lnk*
InteractivePhysics2005.lnk*
matlab.lnk*
ModellusX.lnk*
Navegador Opera.lnk*
PowerPoint 2016.lnk*
proyecto/
Publisher 2016.lnk*
reactivo ecuaciones sin s.docx
reactivos de integracion poe+r partes y sustitucion.docx
Word 2016.lnk*
PC@DESKTOP-SKNH56G MINGW64 ~/desktop (master)
$ cd proyecto
PC@DESKTOP-SKNH56G MINGW64 ~/desktop/proyecto (master)
$
```

9. Para inicializar un nuevo git vamos a introducir el comando (git init). Si revisamos el archivo proyecto que creamos en nuestro escritorio vemos que se creo un archivo oculto llamado .git, como se muestra a continuación:



10. En nuestra carpeta proyecto creamos una nueva carpeta con el nombre subidos y en esta creamos dos archivos en diferentes formatos como se muestra a continuación:



11. Introducimos el comando (git status) que nos muestra el estado de los archivos dentro de la carpeta proyecto. Como podemos observar las carpetas que se marcan de color rojo son aquellas que no están agregadas.

```

MINGW64/c/Users/PC/desktop/proyecto
reactivos de integracion poe+r partes y sustitucion.docx
Word 2016.lnk**

PC@DESKTOP-5KNHS6G MINGW64 ~/desktop (master)
$ cd proyecto

PC@DESKTOP-5KNHS6G MINGW64 ~/desktop/proyecto (master)
$ git init
Reinitialized existing Git repository in C:/Users/PC/Desktop/proyecto/.git/

PC@DESKTOP-5KNHS6G MINGW64 ~/desktop/proyecto (master)
$ git status
On branch master
Your branch is up to date with 'subidos/master'.

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        deleted:    andres.txt
        deleted:    programacion.docx

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        subidos/

no changes added to commit (use "git add" and/or "git commit -a")
PC@DESKTOP-5KNHS6G MINGW64 ~/desktop/proyecto (master)
$

```

12. Para agregar los archivos vamos a insertar el código (git add "nombre del archivo"), dependiendo de los archivos que queramos agregar. Cuando hayamos terminado los archivos nuevamente insertamos el código (git status) y los archivos cambiaran a un color verde que significa que ya están agregados.

```

MINGW64/c/Users/PC/desktop/proyecto
$ git init
Reinitialized existing Git repository in C:/Users/PC/Desktop/proyecto/.git/

PC@DESKTOP-5KNHS6G MINGW64 ~/desktop/proyecto (master)
$ git status
On branch master
Your branch is up to date with 'subidos/master'.

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        deleted:    andres.txt
        deleted:    programacion.docx

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        subidos/

no changes added to commit (use "git add" and/or "git commit -a")
PC@DESKTOP-5KNHS6G MINGW64 ~/desktop/proyecto (master)
$ git add andres.txt
PC@DESKTOP-5KNHS6G MINGW64 ~/desktop/proyecto (master)
$ git add programacion.docx
PC@DESKTOP-5KNHS6G MINGW64 ~/desktop/proyecto (master)
$ git status
On branch master
Your branch is up to date with 'subidos/master'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        deleted:    andres.txt
        deleted:    programacion.docx

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        subidos/

PC@DESKTOP-5KNHS6G MINGW64 ~/desktop/proyecto (master)
$

```

13. Para empezar el proceso de guardado insertamos el siguiente código (git commit -m "Mi primera actualización") para dar un nombre general a nuestro archivo.

```
MINGW64/c/Users/PC/desktop/proyecto
$ git add andres.txt
PC@DESKTOP-5KNH56G MINGW64 ~/desktop/proyecto (master)
$ git add programacion.docx
PC@DESKTOP-5KNH56G MINGW64 ~/desktop/proyecto (master)
$ git status
On branch master
Your branch is up to date with 'subidos/master'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

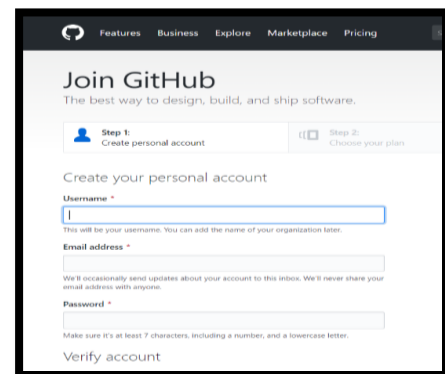
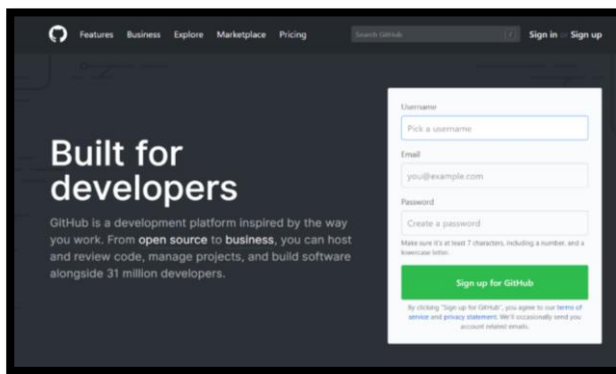
        deleted:    andres.txt
        deleted:    programacion.docx

Untracked files:
  (use "git add <file>..." to include in what will be committed)

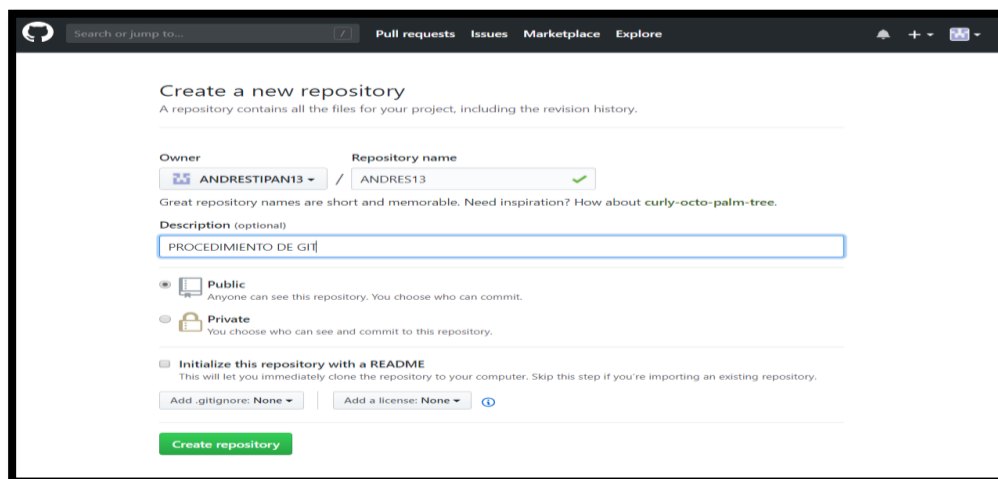
        subidos/

PC@DESKTOP-5KNH56G MINGW64 ~/desktop/proyecto (master)
$ git commit -m "Mi primera actualizacion"
[master 0e4f7c5] Mi primera actualizacion
 2 files changed, 1 deletion(-)
 delete mode 100644 andres.txt
 delete mode 100644 programacion.docx
PC@DESKTOP-5KNH56G MINGW64 ~/desktop/proyecto (master)
$
```

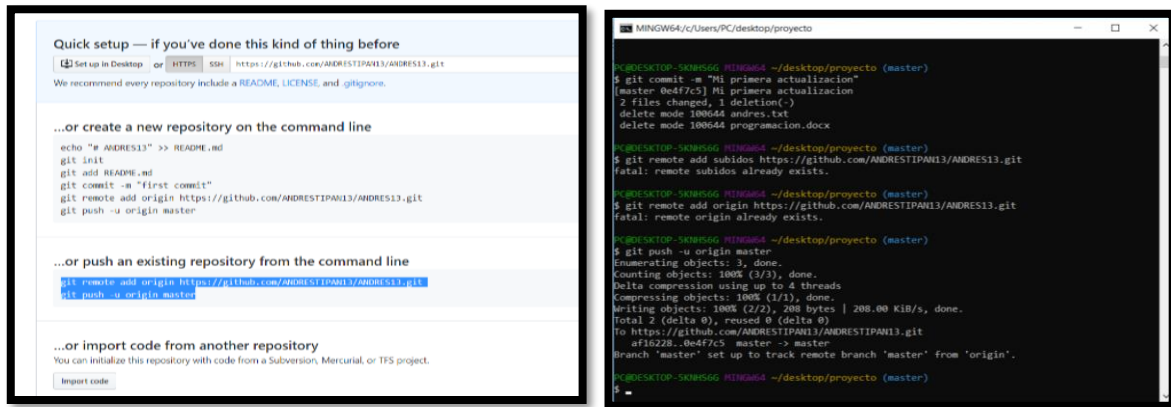
14. Ingresamos via internet a la página oficial de GIT HUB (github.com) y nos creamos una respectiva cuenta en donde conste nuestro nombre de usuario, nuestro correo y nuestra contraseña. Además, nos pedirán otros requisitos para crear la cuenta.



15. Una vez creada nuestra cuenta damos clic en start proyecto y nos aparecerá una ventana donde nos pedirán la información con la que queremos guardar nuestro archivo. Marcamos la opción pública y luego la opción crear repositorio.



16. Nos aparecerá una pantalla con ciertos códigos y copiamos los señalado en nuestra ventana git. Una vez que este copiado vamos a cambiar la palabra origin por la palabra subidos. Cuando realicemos esta operación nos aparecerá un código que nos dice que los archivos se han subido correctamente a nuestra cuenta, como se muestra a continuación



17. Aquí observamos que nuestro archivo ya se ha guardado con el mismo nombre que le dimos (PROCEDIMIENTO DE GIT)

