

UNIVERSIDAD DE PANAMÁ
CENTRO REGIONAL UNIVERSITARIO DE VERAGUAS
FACULTAD DE INFORMÁTICA, ELECTRÓNICA Y
COMUNICACIÓN

JORNADA DE ACTUALIZACIÓN ACADÉMICA CRUV-FIEC-
2021

Tema

- ***Configuración de un entorno de desarrollo.***

Recopilado por
Carlos J. Concepción P.

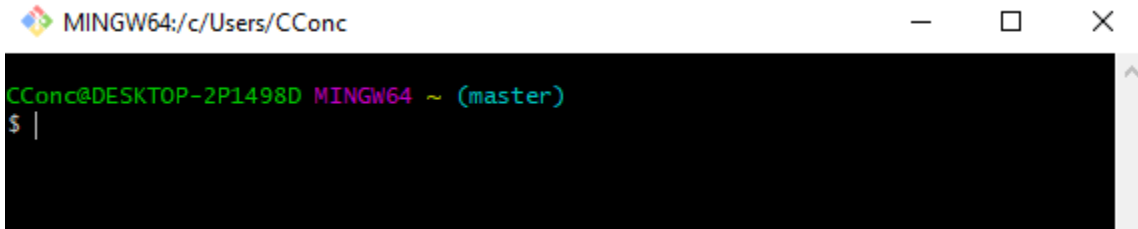
Jornada realizada desde
El 8 al 12 de febrero de 2021

Carlos Concepción

- **Evidencia de la realización de las prácticas acerca de Entornos de Desarrollo.**

CLONANDO REPOSITORIOS CON GIT

Antes de empezar a clonar repositorios, debemos descargar e instalar Git. Luego ejecutamos el Git Bash:



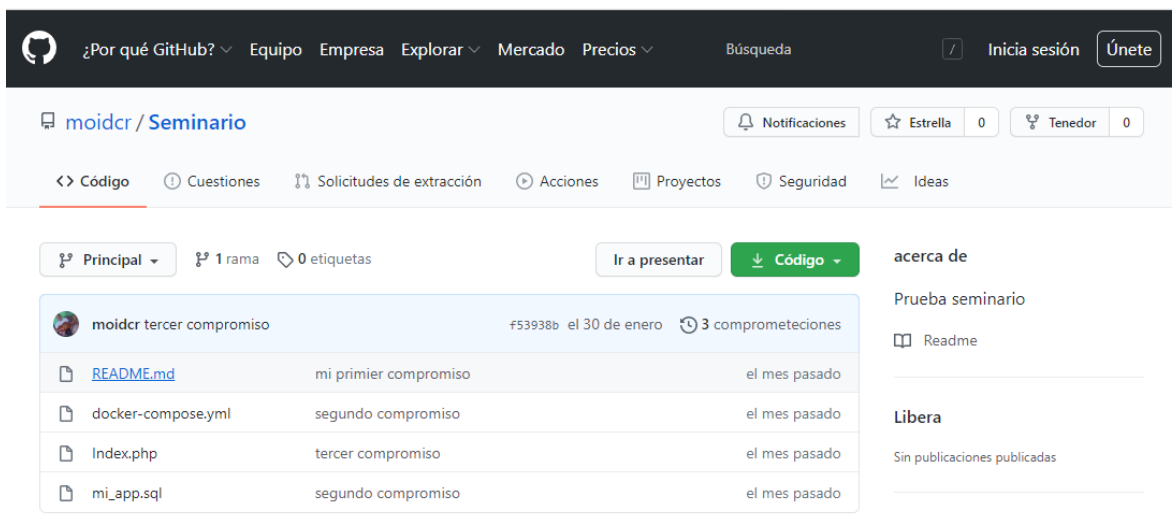
```

MINGW64: c:/Users/CConc

CConc@DESKTOP-2P1498D MINGW64 ~ (master)
$ |
  
```

Seguidamente, para clonar un repositorio o archivos con git necesitamos el url o ruta donde se encuentra alojado, en este caso utilizaremos Github, la ruta la obtendremos en el repositorio deseado.

La ruta en este caso sería: <https://github.com/moidcr/seminario.git>



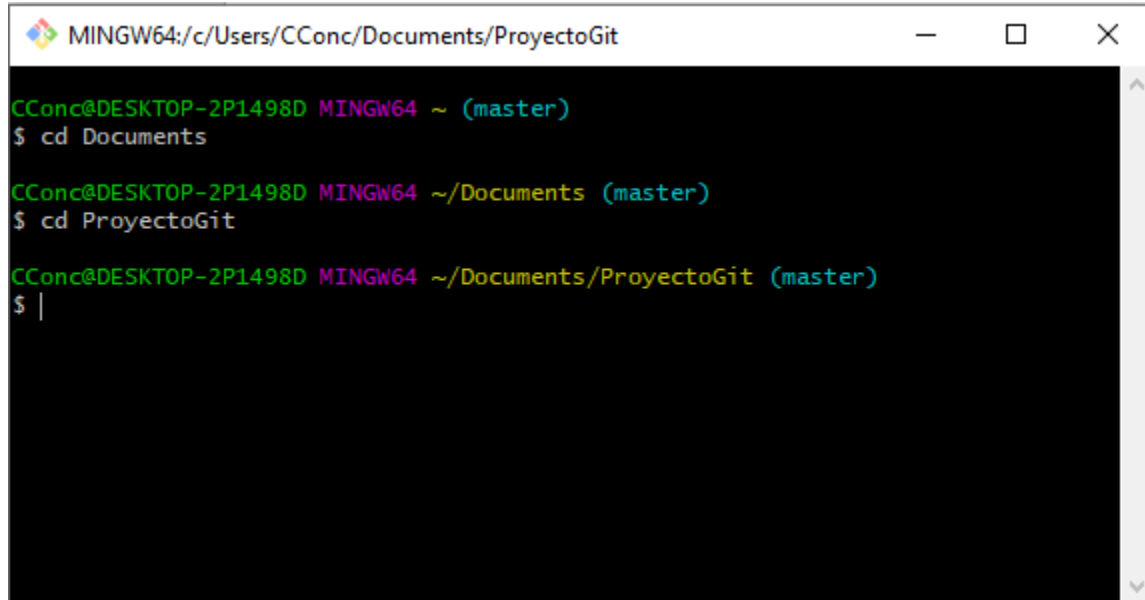
Ahora utilizando el Git Bash, nos dirigiremos a una ubicación que nos resulte sencilla para clonar los archivos. En mi caso, lo haré en la siguiente ubicación:

C:\Users\CConc\Documents\ProyectoGit



Carlos Concepción

Lo cual en el Git Bash, sería así:



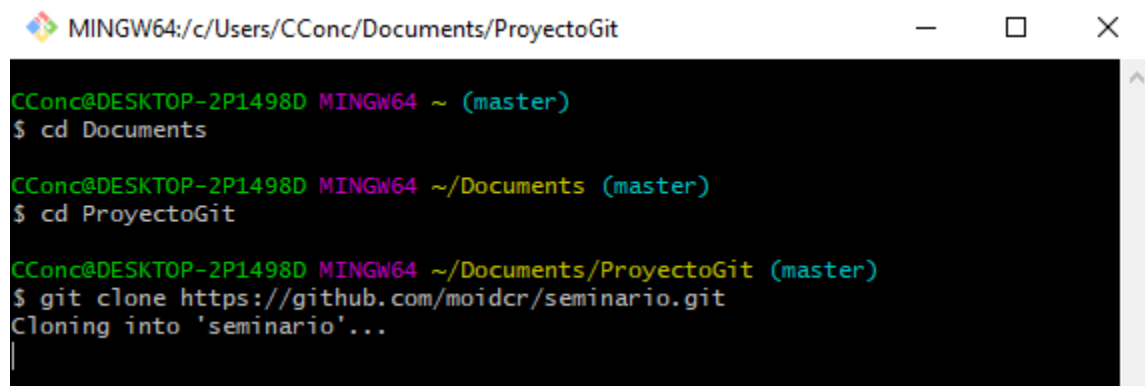
```
MINGW64:/c/Users/CConc/Documents/ProyectoGit
CConc@DESKTOP-2P1498D MINGW64 ~ (master)
$ cd Documents

CConc@DESKTOP-2P1498D MINGW64 ~/Documents (master)
$ cd ProyectoGit

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit (master)
$ |
```

Estando ubicado en el archivo donde deseamos clonar el archivo de GitHub, ejecutamos en siguiente comando:

git clone <https://github.com/moidcr/seminario.git>



```
MINGW64:/c/Users/CConc/Documents/ProyectoGit
CConc@DESKTOP-2P1498D MINGW64 ~ (master)
$ cd Documents

CConc@DESKTOP-2P1498D MINGW64 ~/Documents (master)
$ cd ProyectoGit

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit (master)
$ git clone https://github.com/moidcr/seminario.git
Cloning into 'seminario'...
```

Esperamos hasta que culmine el clonado de los archivos.



Carlos Concepción

Al culminar de clonar los archivos, se debe reflejar la siguiente pantalla.

```

MINGW64; c:/Users/CConc/Documents/ProyectoGit
CConc@DESKTOP-2P1498D MINGW64 ~ (master)
$ cd Documents

CConc@DESKTOP-2P1498D MINGW64 ~/Documents (master)
$ cd ProyectoGit





CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit (master)
$ git clone https://github.com/moidcr/seminario.git
Cloning into 'seminario'...
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 11 (delta 2), reused 11 (delta 2), pack-reused 0
Unpacking objects: 100% (11/11), 2.57 KiB | 5.00 KiB/s, done.

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit (master)
$

```

Y así, nos quedaría la carpeta ***C:\Users\CConc\Documents\ProyectoGit\seminario***, luego de clonar los archivos de la dirección ***git clone <https://github.com/moidcr/seminario.git>***.

Este equipo > Disco local (C:) > Usuarios > CConc > Documents > ProyectoGit > seminario

<input type="checkbox"/> Nombre	Fecha de modificación	Tipo	Tamaño
 docker-compose.yml	03/10/2021 7:07 p. m.	Archivo YML	2 KB
 index	03/10/2021 7:07 p. m.	Archivo PHP	3 KB
 mi_app	03/10/2021 7:07 p. m.	SQL Text File	1 KB
 README.md	03/10/2021 7:07 p. m.	Archivo MD	1 KB

Y mostrando desde Git Bash, nos quedaría:

```

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit (master)
$ cd seminario

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ ls
README.md  docker-compose.yml  index.php  mi_app.sql

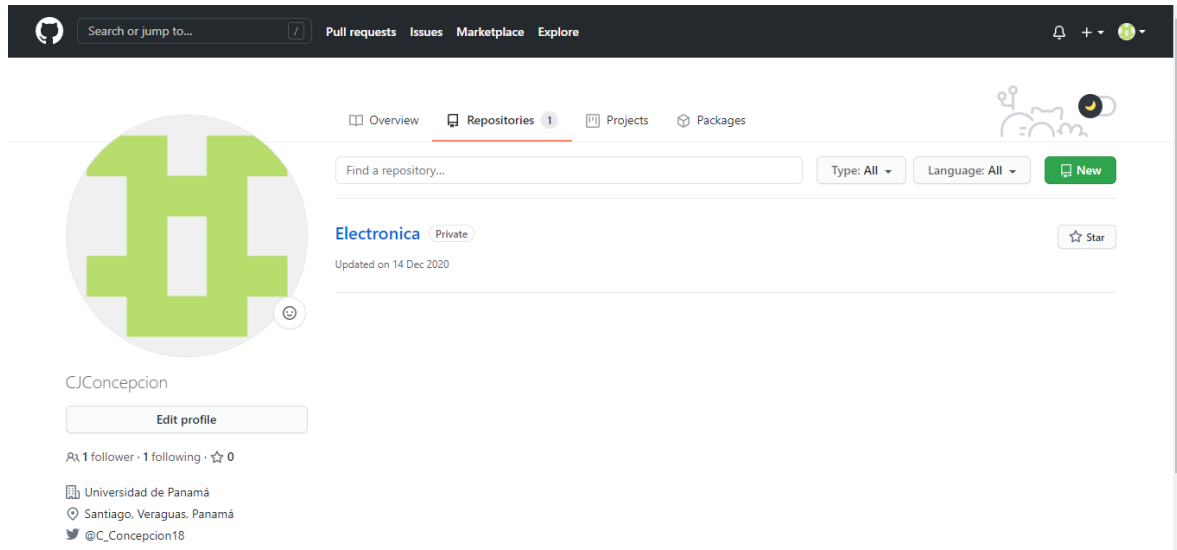
CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$

```



Carlos Concepción

Ahora, para subir archivos a un repositorio en GitHub tenemos que registrar una cuenta de GitHub. En mi caso, ya tenía una cuenta personal registrada, por lo que sólo tuve que subir los archivos.



Ahora utilizando el Git Bash nos ubicamos dentro del directorio seminario, y ejecutamos los siguientes comandos para subir nuestro repositorio directorio a GitHub:

```
MINGW64:/c:/Users/CConc/Documents/ProyectoGit/seminario

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit (master)
$ cd seminario

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ |
```

Luego de ubicarnos dentro del directorio seminario, inicializamos Git en nuestro directorio:



```
MINGW64:/c/Users/CConc/Documents/ProyectoGit/seminario
CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit (master)
$ cd seminario

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ git init
Reinitialized existing Git repository in C:/Users/CConc/Documents/ProyectoGit/seminario/.git/

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ |
```

Seguidamente agreguemos con un punto (.) todos los archivos al **staging** área:

```
CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ git add .

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ |
```

Colocamos nuestro proyecto en el repositorio local con nuestro primer **commit**:

```
CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ git commit -m "Mi primer Commit"
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ |
```

Ahora, definimos la rama de trabajo:

```
CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ git branch -M main

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ |
```

Indiquemos la ruta a nuestro repositorio Git, si obtenemos un error se trata de que sigue apuntando al repositorio de procedencia, cambiémoslo con el **set-url**:




```
CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ git remote add origin https://github.com/CJConcepcion/seminario1.git
fatal: remote origin already exists.

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ git remote set-url origi https://github.com/CJConcepcion/seminario1.git
fatal: No such remote 'origi'


CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ git remote set-url origin https://github.com/CJConcepcion/seminario1.git


CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ |
```

Y por último realizamos el **push** al repositorio indicado, indicando nuestras credenciales de GitHub, obteniendo el siguiente resultado:

 GitHub Login ×

GitHub Login

 Login

 Cancel

Don't have an account? [Sign up](#)

[Forgot your password?](#)



```
CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ git remote add origin https://github.com/CJConcepcion/seminario1.git
error: remote origin already exists.

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ git remote set-url origin https://github.com/CJConcepcion/seminario1.git

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ git push -u origin main
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 4 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (11/11), 2.58 KiB | 43.00 KiB/s, done.
Total 11 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/CJConcepcion/seminario1.git
 * [new branch]      main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.

CConc@DESKTOP-2P1498D MINGW64 ~/Documents/ProyectoGit/seminario (main)
$ |
```

Quedando nuestro repositorio en GitHub de la siguiente manera:

The screenshot shows the GitHub interface for the repository 'CJConcepcion/seminario1'. At the top, there are navigation tabs: Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below these, there's a section for the repository's main branch, 'main', showing 1 branch and 0 tags. A table lists the files in the repository: README.md, docker-compose.yml, index.php, and mi_app.sql, each with its commit history. The README.md file is expanded, showing the title 'seminario'. On the right side, there are sections for 'About', 'Releases', and 'Packages', each with a brief description and a link to create a new one.

File	Commit History
README.md	mi primer commit (last month)
docker-compose.yml	second commit (last month)
index.php	third commit (last month)
mi_app.sql	second commit (last month)

seminario

