



**Tecnológico Nacional de México  
Instituto Tecnológico de Tijuana**

SUBDIRECCIÓN ACADÉMICA  
Departamento de Sistemas y Computación

SEMESTRE  
agosto - diciembre 2021

CARRERA  
Ing. Tecnologías de la Información y Comunicación

MATERIA Y CLAVE:  
Datos Masivos      BDD-1704TI9A

NOMBRE Y MATRICULA DEL ALUMNO:  
Velázquez Farrera César Alejandro      17212937

NOMBRE DEL TRABAJO:  
Práctica 1: Git Flow

UNIDAD POR EVALUAR  
Unidad I

NOMBRE DEL MAESTRO (A):  
M.C. Jose Christian Romero Hernández



## Objetivos de la práctica

- Instalar Git in Windows o Linux
- Hacer tutorial de Git básico
- Ramas en Git

### 1. Crear una cuenta en Github.

Overview Repositories 7 Projects Packages

**Ing. TIC**

Hola a todos 🙋

- 🎓 Estudiante de la carrera de TIC en el Instituto Tecnológico de Tijuana.
- 🌱 Actualmente estoy estudiando para la certificación CCNA de Cisco.
- 🛠 Me gusta mucho manejar equipos de Cisco y todo lo que tiene que ver con equipos de red.
- 🗣 Lenguajes de programación que conozco: C#, PHP, Python 3, Java y VB.
- 📦 Algunas cosas que me interesan: Docker, Raspberry Pi y Arduino.
- 🚁 Uno de mis proyectos más divertidos que he llevado a cabo es la programación de un dron a través de un API utilizando Python. 🇲🇽
- 📶 Fuertes: Redes, Cisco IOS, Edición de vídeo, Packet Tracer y programación en C#.
- ⚡ Dato interesante: Una instrucción que utilizaras mucho en enrutadores Cisco es:  
Router1# show ip interface brief →

Interface	IP-Address	OK?	Method	Status	Protocol
GigabitEthernet0/0	54.59.109.50	YES	DHCP	up	up
GigabitEthernet0/1	192.168.1.254	YES	Manual	up	up
GigabitEthernet0/2	unassigned	YES	unset	up	down

- 😄 Chiste: Hay 10 tipos de personas en este mundo. Aquellos que entienden binario. Y los que no. ¿Qué tipo de personas eres? xD

### 2 Configurar su cuenta de Github con llave SSH, para no realizar la autenticación con nuestras credenciales cada vez que se realice una contribución.

2.1 Para la creación de las llaves SSH se ejecuta el siguiente comando en la terminal de Ubuntu o cualquier de sus sistemas operativos derivados.

```
cesar@cesar-DESKTOP-BJ0DB6:~$ ssh-keygen -t ed25519 -C "correo_institucional"
```

2.2 Especificar el directorio donde se almacenará la llave SSH

```
Generating public/private ed25519 key pair.  
Enter file in which to save the key (/home/cesar/.ssh/id_ed25519): /home/cesar/desktop/key
```



## 2.3 Ingresar una contraseña para mayor seguridad (vacío para llave sin contraseña)

Enter passphrase (empty for no passphrase):  
Enter same passphrase again:

## 2.4 Abrir el archivo generado con terminación .pub con un editor de texto.

```
Open  ▼  [icon]  gitHub_Keys.pub  Save  [icon]  -  [icon]  x
```

```
1 ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQgQDARgsLby0Zj/xPKqaKSKTFiiv8EOVZWzLXCJuBSMD+/QIAM+WDAJu0mUXXGFfV4/-  
BLBGZTqhzFeY0KSB+4JKC6cG0gqt356JXV2PotKW5YgLCtMJdmFW710WHAr2drrxo9+2phAduSquv0Q3mA8+SzMjIJaWslF499WWUoY0pun9jr  
10S9t20g+eCTyNm4oPB7VlepJStv4bPyT0L55T5gGkQBmxLK0DAvYfptgamq6fdjugUkTmNnyUZ+tERRrMpFqkkKiQ9en0M/hW9iiQn3sSVL/-  
XUuHCLurj6wL6hivkP65diB8cYBML6Nc8x07p4uVTedtp8P2jIPtt5pPM0c4Q6venk8nRUoyZ5v94diRPX+mq86inabE0gaK0LUlg8T/-  
QzfPsMcr3bnCwiR/WDV5U6eHJ94Yver/SR0w0= cesar@cesar-HP-Pavilion-Power-Laptop-15-cb0xx
```

## 2.5 En Github entrar a los ajustes de cuenta y buscar el apartado de SSH y llaves GPG

**vcesar454**  
Your personal account

[Go to your personal profile](#)

Account settings

- Profile
- Account
- Appearance
- Account security
- Billing & plans
- Security log
- Security & analysis
- Sponsorship log
- Emails
- Notifications
- Scheduled reminders
- SSH and GPG keys**
- Repositories
- Packages
- Organizations
- Saved replies
- Applications

### SSH keys

[New SSH key](#)

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

**vcesar454**  
SHA256:u0L0Yryx4/8U101uc12Y5qnsFsZV1RhlLw1anycE8  
Added on Aug 31, 2021  
Last used within the last week — Read/write

[Delete](#)

Check out our guide to [generating SSH keys](#) or [troubleshoot common SSH problems](#).

### GPG keys

[New GPG key](#)

There are no GPG keys associated with your account.  
[Learn how to generate a GPG key and add it to your account.](#)

### Vigilant mode Beta

☐ **Flag unsigned commits as unverified**  
This will include any commit attributed to your account but not signed with your GPG or S/MIME key.  
Note that this will include your existing unsigned commits.  
[Learn about vigilant mode.](#)

## 2.6 Presionar el botón con la leyenda “Agregar llave SSH”, copiar el contenido de la llave dentro de la caja de texto grande y dar clic en Agregar llave SSH”. Es posible que se requiera una autenticación.



## SSH keys / Add new

Title

test

Key

```
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAILAC2rRyJfgS+NiT1mV6T2h873tOFsl0Yt4a29UoNeM6  
cesar.velazquez17@tectijuana.edu.mx
```

Add SSH key

### 3. Crear un nuevo repositorio llamado *practica\_gitflow* con las siguientes características:

#### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

#### Repository template

Start your repository with a template repository's contents.

No template ▾

Owner \*

vcesar454 ▾

Repository name \*

/ practica\_gitflow ✓

Great repository names are short and memorable. Need inspiration? How about [musical-doodle?](#)

Description (optional)



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

#### Initialize this repository with:

Skip this step if you're importing an existing repository.

☒ Add a README file

This is where you can write a long description for your project. [Learn more](#).

☐ Add .gitignore

Choose which files not to track from a list of templates. [Learn more](#).

☐ Choose a license

A license tells others what they can and can't do with your code. [Learn more](#).

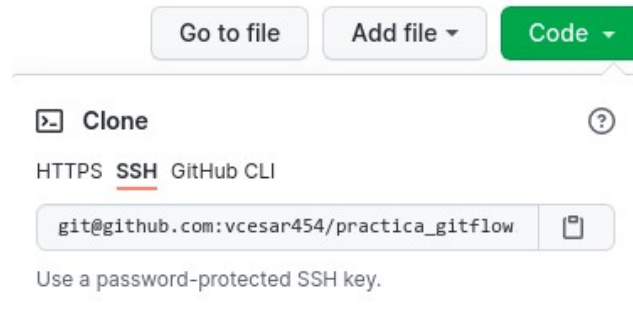
This will set main as the default branch. Change the default name in your [settings](#).

Create repository

- Sin plantilla
- **Nombre de repositorio:**  
*practica\_gitflow*
- **Descripción:** Opcional
- **Publico**
- **Agregar** README file por defecto.

## 4. Clonar repositorio a nuestra PC

4.1 Copiar el enlace de conexión tipo SSH bajo el botón de “Codigo”.



4.2 Utilizar la siguiente instrucción en la terminal y dentro del directorio donde deseamos clonar el repositorio. Inicializar git.

```
cesar@cesar-DESKTOP-BJ0DB6:~/Desktop$: git clone  
git@github.com:vcesar454/practica_gitflow.git  
cesar@cesar-DESKTOP-BJ0DB6:~/Desktop$: git init
```

```
Cloning into 'practica_gitflow'...  
Warning: Permanently added the RSA host key for IP address '140.82.112.4' to the list of known hosts.  
remote: Enumerating objects: 3, done.  
remote: Counting objects: 100% (3/3), done.  
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0  
Receiving objects: 100% (3/3), done.
```

## 5. Crear una rama (branch) llamado *development* y hacer esta la rama principal por defecto.

5.1 Utilizar la siguiente instrucción en la terminal

```
cesar@cesar-DESKTOP-BJ0DB6:~/Desktop$: git checkout -b development  
Switched to a new branch 'development'
```

5.2 Actualizar el repositorio con el siguiente comando. También se establece el branch a actualizar.

```
cesar@cesar-DESKTOP-BJ0DB6:~/Desktop$: git push --set-upstream origin  
development
```

```
Total 0 (delta 0), reused 0 (delta 0)  
remote:  
remote: Create a pull request for 'development' on GitHub by visiting:  
remote: https://github.com/vcesar454/practica_gitflow/pull/new/development  
remote:  
To github.com:vcesar454/practica_gitflow.git  
* [new branch]      development -> development  
Branch 'development' set up to track remote branch 'development' from 'origin'.
```



5.3 Desde el panel en Github asignar al branch *development* como la rama principal

Switch default branch to another branch ×

Choose another branch to use as the default branch of vcesar454/practica\_gitflow instead of `main`:

development ▾

Update

6. Realizar un cambio al archivo *README.md* con un editor de código. Después ejecutar el comando `git status` para verificar el cambio.

```
On branch development
Your branch is up to date with 'origin/development'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md

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

7. Agregar el cambio con el siguiente comando:

```
cesar@cesar-DESKTOP-BJ0DB6:~/Desktop$: git add README.md
```

```
On branch development
Your branch is up to date with 'origin/development'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   README.md
```

8. Realizar un *commit* en la rama *development* con el siguiente comando:

```
cesar@cesar-DESKTOP-BJ0DB6:~/Desktop$: git commit -m "primercommit"
```

```
[development f4846c8] primer commit
1 file changed, 3 insertions(+), 1 deletion(-)
```





**9. Realizar un *push* a la rama *development* con el siguiente comando:**

```
cesar@cesar-DESKTOP-BJ0DB6:~/Desktop$:git push
```

```
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 291 bytes | 291.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:vcesar454/practica_gitflow.git
d1a2813..f4846c8 development -> development
```

**10. Crear una rama llamada *features* con el siguiente comando:**

```
cesar@cesar-DESKTOP-BJ0DB6:~/Desktop$:git checkout -b features
```

**11. Realizar cambios al archivo *README.md***

```
On branch features
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md

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

**12. Realizar las operaciones *add*, *commit* y *push* a la rama *features***

```
cesar@cesar-DESKTOP-BJ0DB6:~/Desktop$:git add README.md
```

```
cesar@cesar-DESKTOP-BJ0DB6:~/Desktop$:git commit -m "commit a la rama
features"
```

```
cesar@cesar-DESKTOP-BJ0DB6:~/Desktop$:git push -set-upstream origin features
```

```
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 328 bytes | 328.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'features' on GitHub by visiting:
remote:   https://github.com/vcesar454/practica_gitflow/pull/new/features
remote:
To github.com:vcesar454/practica_gitflow.git
 * [new branch]      features -> features
Branch 'features' set up to track remote branch 'features' from 'origin'.
```



🔗 features had recent pushes 7 minutes ago

Compare & pull request

🔗 features ▾

🔗 3 branches

🔗 0 tags

Go to file

Add file ▾

Code ▾

This branch is 1 commit ahead of development.

🔗 Contribute ▾



vcesar454 commit a la rama features

d8dbf08 9 minutes ago ⌚ 3 commits

📄 README.md

commit a la rama features

9 minutes ago

README.md



## practica\_gitflow

ESTO ES UNA PRUEBA

CREACION DE LA RAMA FEATURES