Instituto Tecnológico de Tijuana

Ingeniería en Sistemas Computacionales

Práctica:

Práctica Git Flow

Materia: Datos Masivos

Unidad: Unidad I

Facilitador:

José Christian Romero Sánchez

Alumno: Hernández Negrete Juan Carlos - **16212021**

Fecha:

Tijuana Baja California a 17 de Septiembre del 2020.

Practice Git Flow

1. Create an account on github.com

2. Configure your github account with SHH key, so as not to put our username and password every time

- **User creation**

```
$ git config --global user.name "Juan Negrete"
```
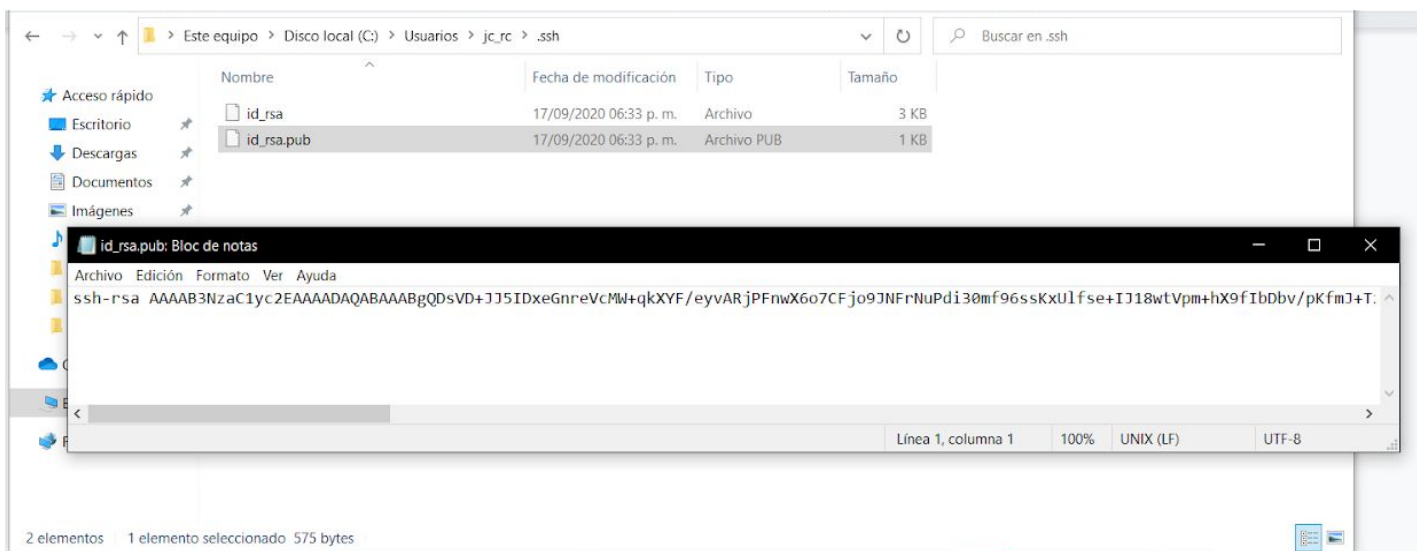
- **Mail in Git**

```
$git config --global user.email "john.carlos.116@gmail.com"
```

- **We generate SSH key**

```
$ ssh -v

$ ssh-keygen
```

- **Access the path of the .ssh folder, open the file "id_rsa.pub" and copy the text**

- **Paste the text in the SSH KEYS section and in New Key**

SSH keys / Add new

Title

Key

Begins with 'ssh-rsa', 'ssh-ed25519', 'ecdsa-sha2-nistp256', 'ecdsa-sha2-nistp384', or 'ecdsa-sha2-nistp521'

Add SSH key

- **The key is created successfully.**

JuanCarlos-Negre...
Personal settings

Profile

Account

Account security

Security log

Security & analysis

Emails

SSH keys   New SSH key

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

SSH

Llave
MD5:50:5e:91:87:ca:cd:11:e3:57:e5:1e:d8:08:90:13:ad
SHA256:DB7jocdZIxxvvnQjYeJ1D1pHQZ0DWkT7gqqYUKEadvo
Added on 17 Sep 2020
Never used — Read/write   Delete

Check out our guide to generating SSH keys or troubleshoot common SSH Problems.

3. Create a new repository called pratice_git_flow

- **We assign a name to the repository**

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Owner *

⬇ JuanCarlos-Negrete ▾ / practica_git_flow ✓

Great repository names are short and memorable. Need inspiration? How about **stunning-pancake**?

Description (optional)

Practica #1

- **Was created successfully**

🖥 JuanCarlos-Negrete / **practica_git_flow**

<> Code ⓘ Issues ⏎ Pull requests ⊙ Actions ▦ Projects 📖 Wiki ⛉ Security ⠘ Insights ⚙ Settings

⌥ master ▾   ⌥ 1 branch ⬡ 0 tags   Go to file   Add file ▾   ⬇ Code ▾

⬇ JuanCarlos-Negrete Initial commit   8e0e5d3 15 minutes ago   ⦵ 1 commits

📄 README.md   Initial commit   15 minutes ago

README.md   ✎

practica_git_flow

Practica #1

4. Create a default README.md

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**
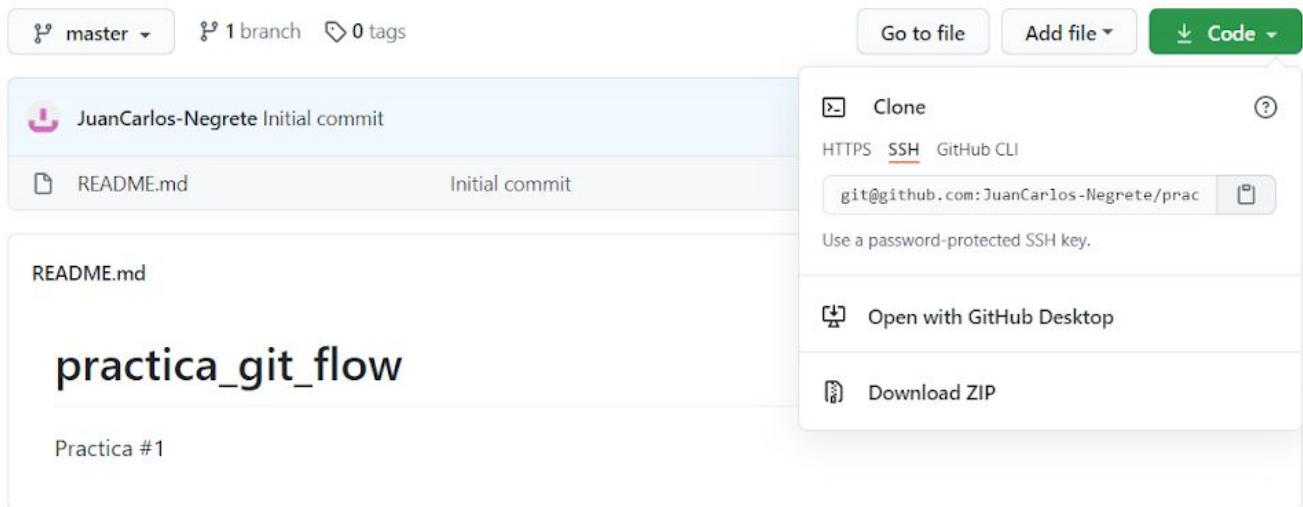Choose which files not to track from a list of templates. Learn more.

☐ **Choose a license**
A license tells others what they can and can't do with your code. Learn more.

This will set ⌥ master as the default branch. Change the default name in your settings.
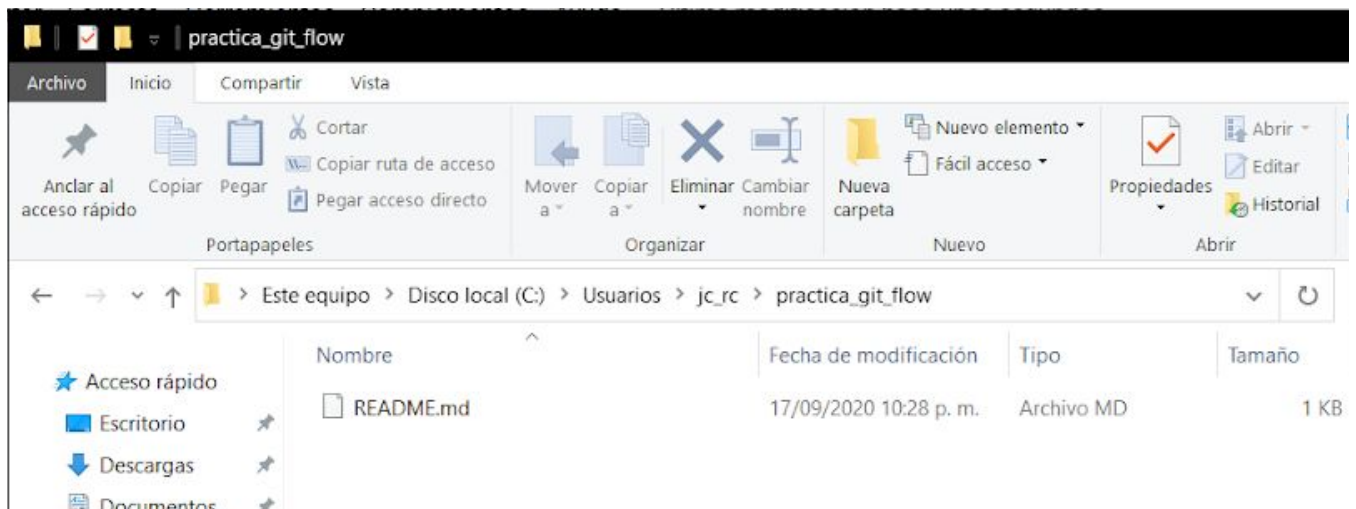
Create repository

5. Clone our PC

- **Copy the link of the repository (as SSH) to execute it in GIT BASH of our PC**


```
git clone git@github.com:JuanCarlos-Negrete/practica_git_flow.git
```

It was cloned successfully

6. Create a branch called development and make this branch the main one, this means that it will not be master

Default branch

The default branch is considered the "base" branch in your repository, against which all pull requests and code commits are automatically made, unless you specify a different branch.
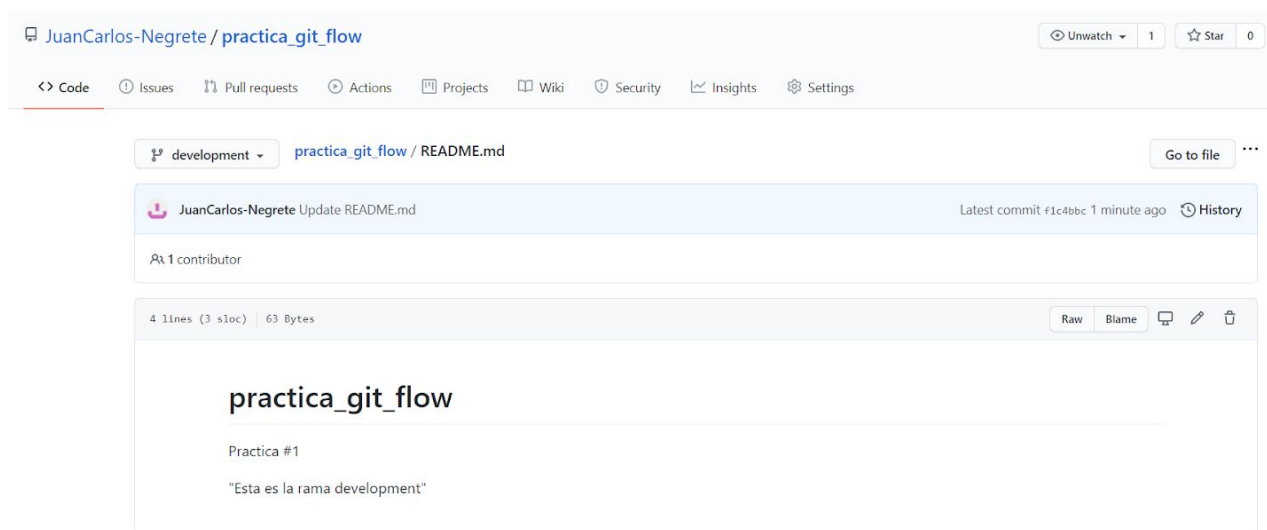
7. Make a change in the development branch in the README.md file could be "This is the development branch"
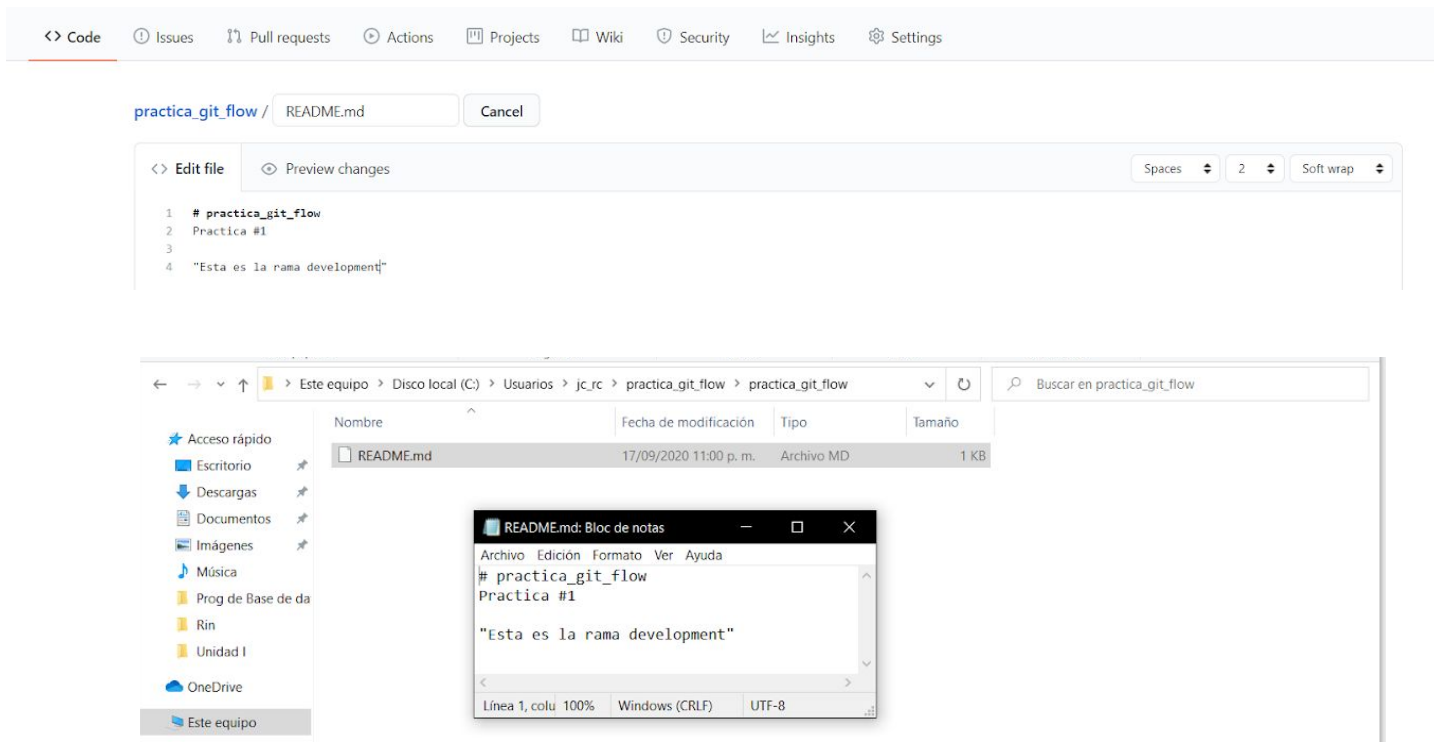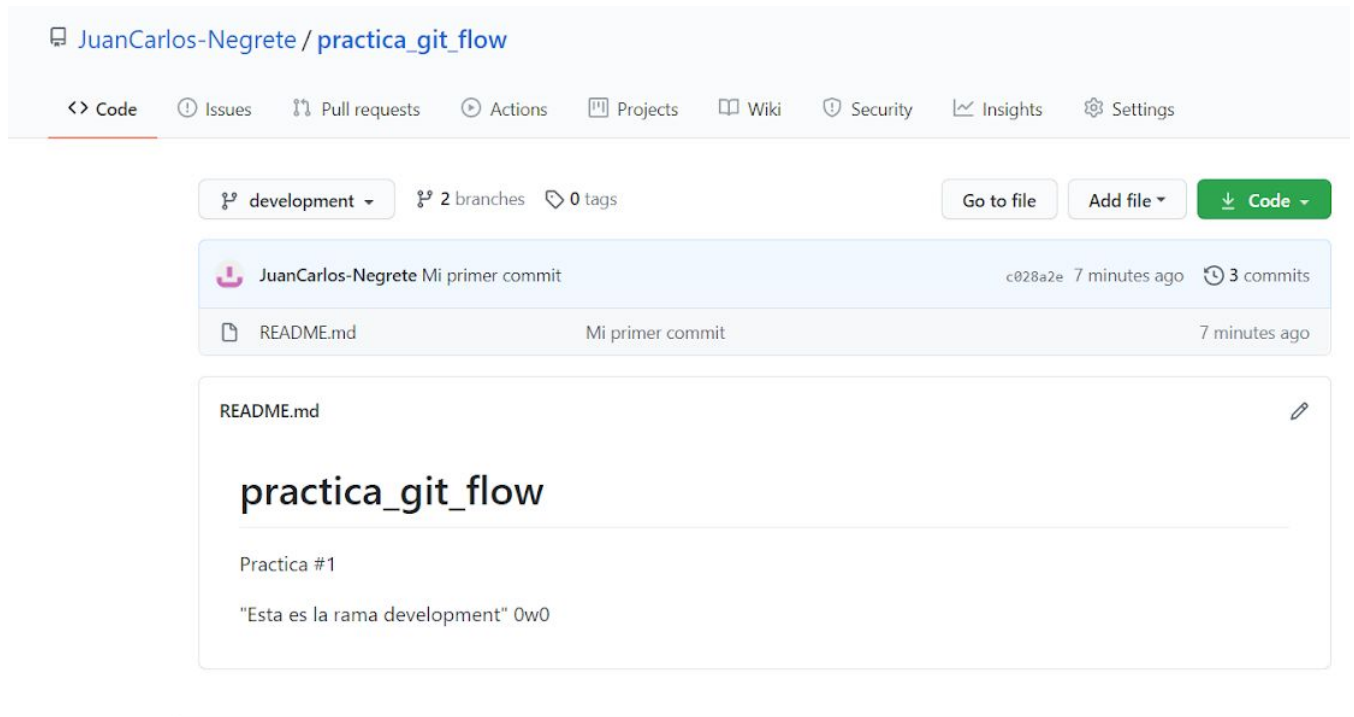

8. Make a commit in the development branch

```
$ git add Readme.md

$ git commit -a
```

9. Push the development branch

```
$ git push
```


10. Create a branch called features

```
$ git branch features
$ git show-branch
$ git checkout features
```

11. Make a change in the features branch in the README.md file it can be "This is the features branch"
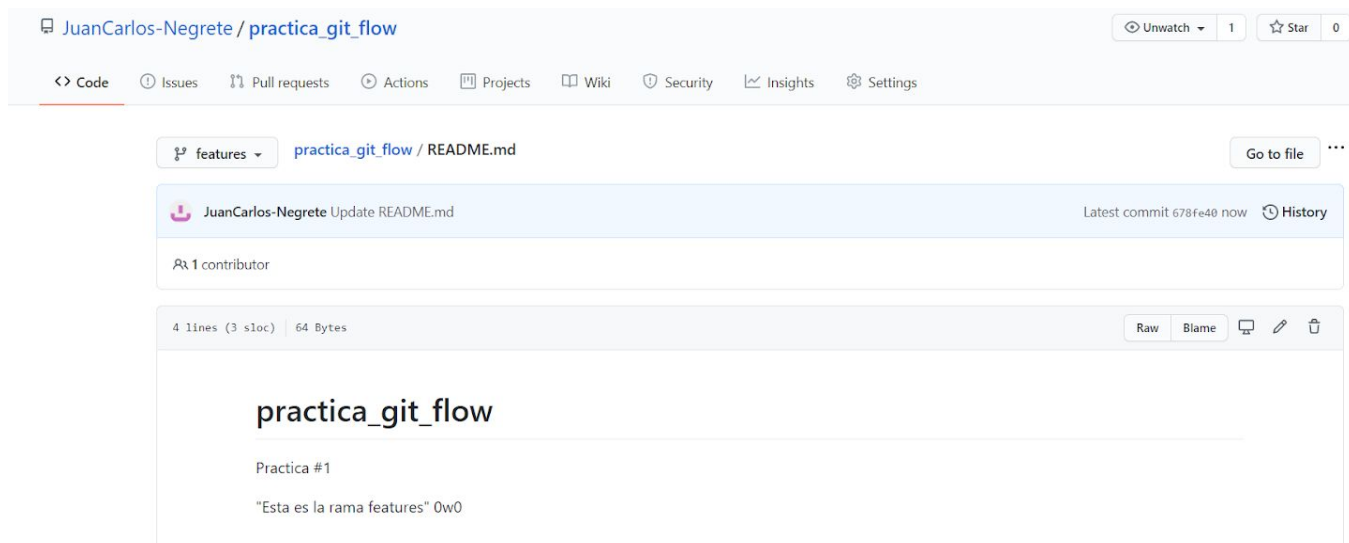
```
$ git add Readme.md
$ git commit -a

 --Se hizo cambio con éxito

$ git push
```

12. Make the commit on this branch.

```
$ git checkout features
$ git commit -a
$ git push
```

13. Do the push and verify that the change is in the repository

Project link: