Reporte de práctica de laboratorio

Control de Versiones

29/03/2019

Maestría en Sistemas Computacioneales

*Aplicaciones y Servicios en la Nube*

Prof. Mtro. Rodolfo Luthe Ríos

Juan Fernando Melchor Galindo

ms700500@iteso.mx

# Introducción

El propósito de la práctica numero 10 consiste en contar con una gestión de versiones que se realizan a archivos ya sean documentos planos o a código fuente, lo que centralmente se enfocan actualmente estas herramientas.

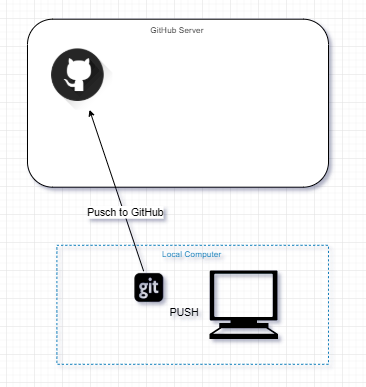
# Marco Teórico

Dentro de un esquema organizado entre las empresas consta el correcto control de versiones de archivos que resultan importantes para el negocio, así mismo como el poder gestionar el versionamiento de código fuente mediante herramientas centralizadas donde de forma colaborativa se pueda compartir y unificar a la vez información, dentro de algunos servicios actuales existen los siguientes:

GIT: Aplicación local la cual permite mediante publicaciones a un repositorio en GitHub, esto para poder realizar versionamiento de objetos de manera colaborativa, tal como se indica en [2].

Github: Es una herramienta para alojar el control de versiones de las publicaciones que realice GIT, es en la actualidad una de las mayores herramientas utilizadas para el control de versiones de los equipos de desarrollo [3].

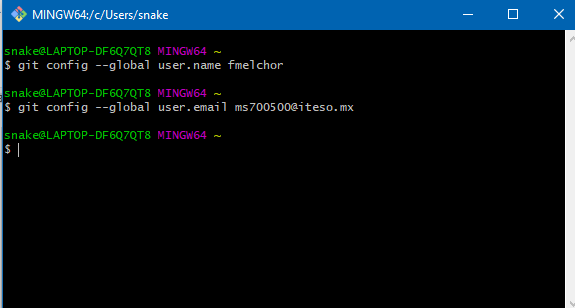
# Diagrama



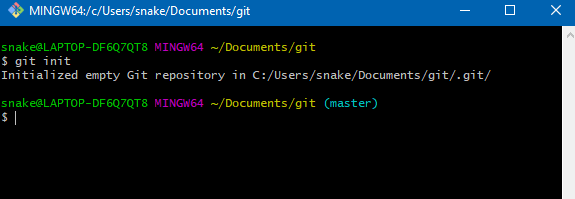
# Desarrollo de la Práctica.

**Configurar Repositorio Local**

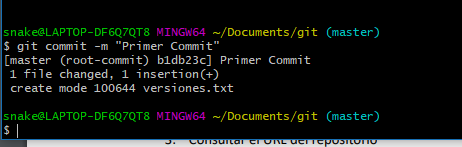
Para la práctica local de git es requerido descargar la herramienta local de la dirección [2], posteriormente ejecutar los siguientes comandos para configurar la cuenta con los datos de mi cuenta de ITESO.



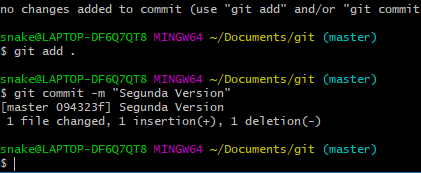
Posteriormente crear el repositorio local que será sincronizado con github



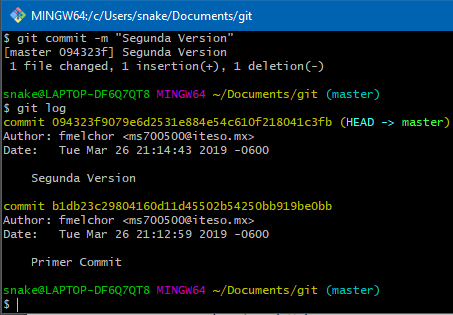
Creamos en el directorio un archivo txt llamado versiones.txt y posteriormente ejecutamos la instrucción para marcar el cambio (commit).



Publico los cambios al repositorio con push y realizo un cambio en el archivo versionest.txt para marcar un segundo commit de cambios.

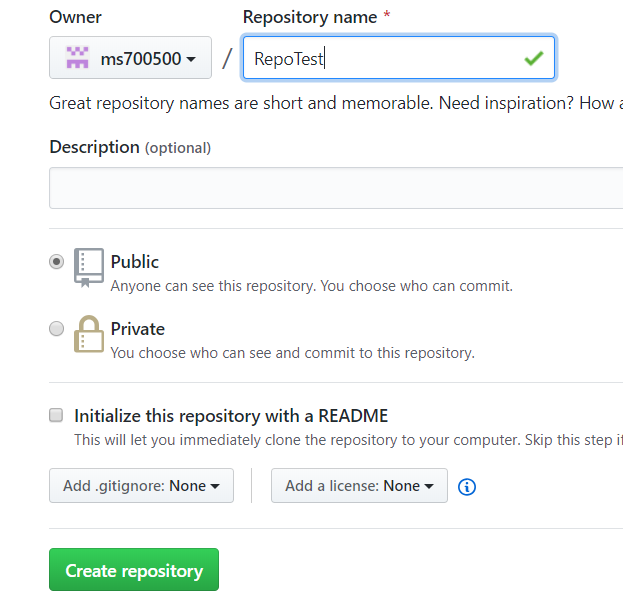


Posteriormente para revisar el log de cambios se puede ejecutar la siguiente sentencia git log.



**Configurar Repositorio en GitHub**

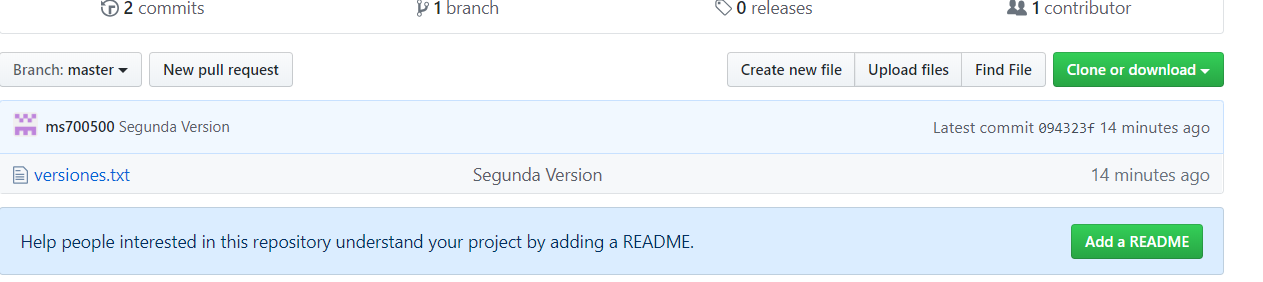
Posteriormente me dirijo a la página [3] para crear una cuenta Github con mis credenciales de ITESO, posteriormente creo un repositorio de prueba para la sincronización.



Donde la URL resultante es la siguiente:

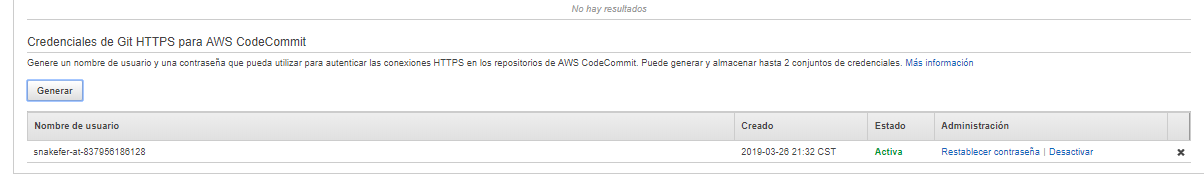
<https://github.com/ms700500/RepoTest.git>

Posteriormente valido que al hacer push de mis cambios, los puedo revisar directamente en github:



**Configurar Repositorio en AWS CodeCommit**

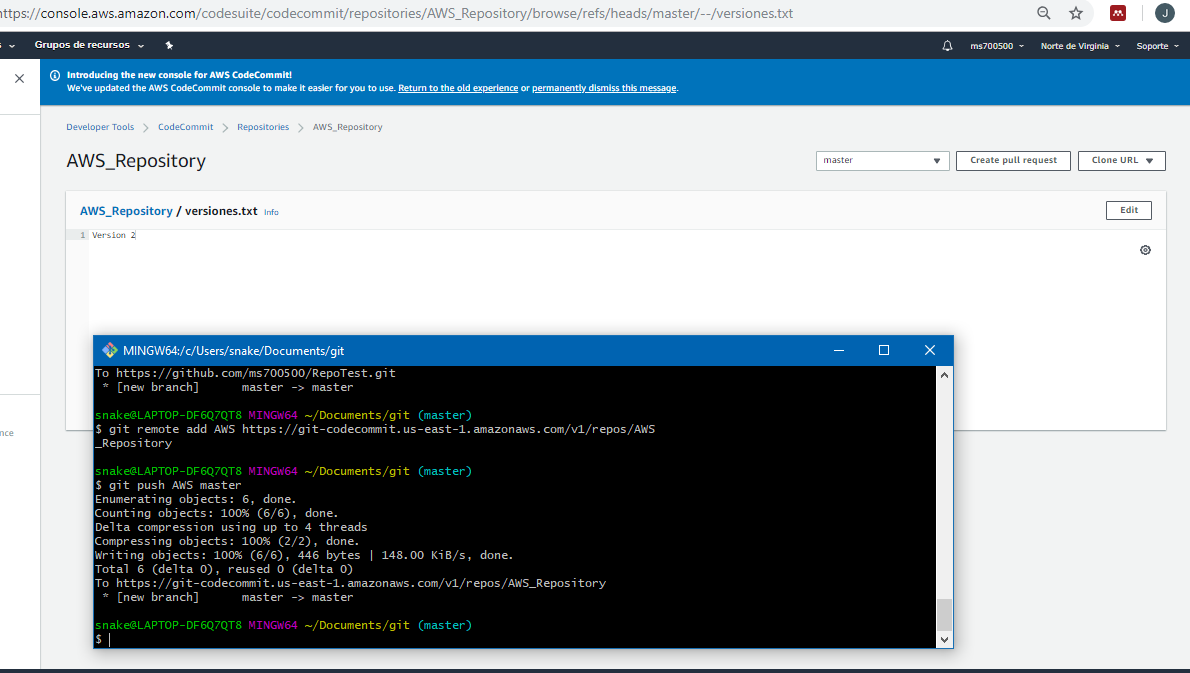
Dentro de AWS entrar a IAM para que en las opciones del final hacer clic en el botón Generar y así crear un Git https:



Obtengo la URL que se genera para este repositorio:

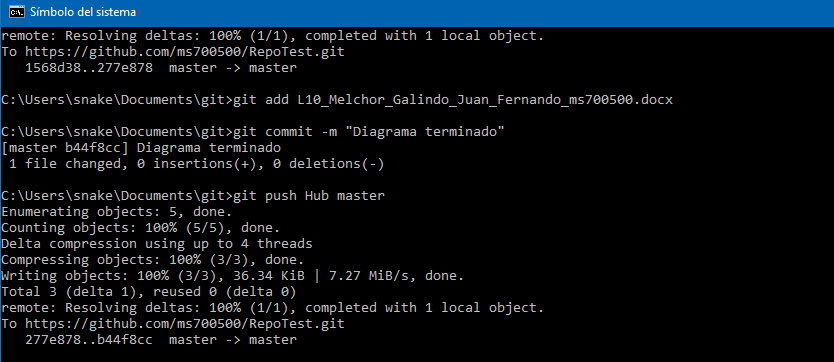
https://git-codecommit.us-east-1.amazonaws.com/v1/repos/AWS\_Repository

Posteriormente con la misma consola local de Git ejecuto la siguiente sintaxis de comandos con la URL en esta ocasión de AWS, ingresando remote y push para subir los cambios:

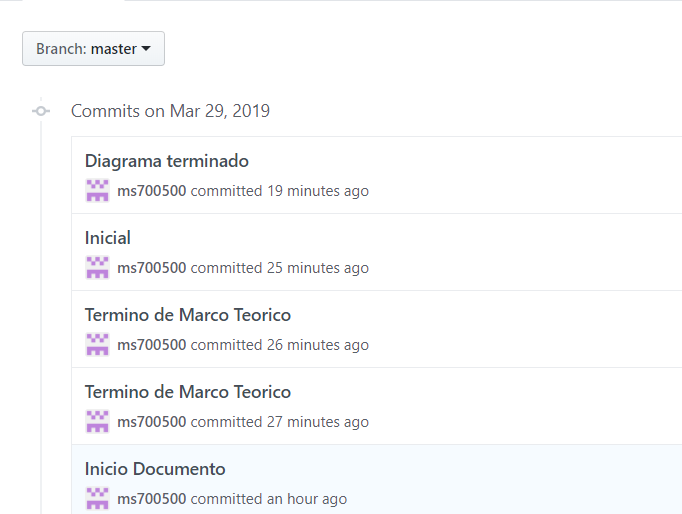


**Configurar Repositorio Local**

Para efectos del último punto en la práctica, realizo el control de versiones con GitHub de igual forma, donde voy dando commits conforme voy avanzando en este documento de tarea,



Iba realizando ciertas pruebas para entender el comportamiento por ello se visualizan varios commits con el mismo texto en ellas, pero al final solo quedo la versión actual en el repositorio con el ultimo commit.



# Conclusiones

.

# Bibliografía

[1] <https://calculator.s3.amazonaws.com/index.html>

[2] <https://git-scm.com/>

[3] <https://github.com/>