

Git, GitHub y GitLab

Luis Miguel de la Cruz Salas

GMMC, Recursos Naturales
Proyecto PAPIME PE101019

Octubre 2019



Agradecimientos

Trabajo realizado con el apoyo del
Programa UNAM-DGAPA-PAPIME
PE101019



Git

- Wikipedia: Es un software de control de versiones de aplicaciones, cuando éstas tienen un gran número de archivos de código fuente. Entre otras cosas lleva un registro de los cambios en cada archivo y coordina el trabajo de personas sobre archivos compartidos.
- Se puede usar desde una terminal de comandos.
 - En Windows se recomienda usar **Git Bash** como terminal, la cual se instala junto con git.
- Existen multitud de interfaces gráficas de usuario (gui clients) para uso de git.
 - **Git GUI** es la interfaz gráfica oficial distribuida junto con git.



Seguimiento de un proyecto con Git



Instalación de Anaconda y git

1. Descargar e instalar [anaconda](#) de acuerdo con su S.O.
 - a. Windows (.exe), MacOS (.pkg), Linux (.sh)
 - b. [Instrucciones de instalación](#).
2. Una vez con Anaconda instalado, puede escribir programas en Python con extensión **.py** en cualquier editor de texto.
 - a. Genere el archivo **hola.py** con la siguiente línea: `print('Hola mundo!')`
 - b. Abra una terminal y ejecute el comando: `python hola.py`
 - i. En Windows use **Anaconda Prompt** para abrir una ventana de ejecución.
3. Descargar e instalar git de <https://git-scm.com/> de acuerdo con su S.O.
 - a. Windows (.exe), MacOS (.dmg), Debian/Ubuntu: `apt install git`
 - b. Otras distribuciones de Linux [aquí](#).





Configuración de Git con GitHub o GitLab

Lo que sigue a continuación está basado en el documento [1] de las referencias.

1. Crear una cuenta en [GitHub](#) o [GitLab](#).
2. Configurar el git en su equipo usando la cuenta de GitHub o de GitLab:
 - a. `git config --global user.name "usuario"`
 - b. `git config --global user.email "correo@server.com"`

donde *usuario* y *correo@server.com* son los que se usaron al crear la cuenta en GitHub o GitLab.

3. Verificar que la configuración anterior esté correcta:
 - a. `git config --global --list`





Configuración de Git con GitHub o GitLab

4. Crear un par de llaves pública y privada:

- a. `ssh-keygen -o -t rsa -C "correo@server.com" -b 4096`
- b. Dar <enter> en cada pregunta que haga el comando.

5. Checar el contenido de la llave pública creada:

- a. `cat ~/.ssh/id_rsa.pub`

6. GitHub: entrar a su cuenta:

- a. Ir a: **Settings > SSH and GPG keys**.
- b. Hacer clic en el botón **New SSH Key**.
 - i. Agregar un título, el cual es para identificar a la computadora local.
 - ii. Hacer un *copy-paste* de la llave pública `.ssh/id_rsa.pub` en el espacio **Key**.
 - iii. Hacer clic en el botón **Add SSH key**.
- c. Checar que todo funcione como se describe en la página 5 de [1].

7. GitLab: El proceso completo se describe en [1], páginas 4 y 5.





Proyecto inicial local

1. Abra una terminal de comandos (en windows use: **Anaconda Prompt**).
2. Crear el directorio **PythonBasico**:
 - a. `mkdir PythonBasico`
3. Cambiarse al directorio recién creado:
 - a. `cd PythonBasico`
4. Crear el archivo **hello.py** con el siguiente contenido:

```
print('Hola mundo pythonico!')
```
5. Ejecutar para revisar que funciona:
 - a. `python hello.py`



Sincronizar un proyecto con GitHub o GitLab

- **GitLab:** Estando en el directorio `PythonBasico`, realice todas las instrucciones indicadas desde la página 6 de [1] (iniciando con `git init`), hasta la página 11 (terminando con `git push -u origin master`). Usted usará el archivo `hello.py` , creado anteriormente, como código fuente y el directorio `PythonBasico` (en vez de `SuperProyecto`).
- **GitHub:** El proceso es similar con las siguientes consideraciones:
 - Agregar un `New repository` (en vez de `New Project`) y definir el `Repository name` (en vez `Project name`).
 - Adicionalmente, puede agregar un archivo `README` inicial, un archivo `.gitignore` (seleccione `Python`) y una licencia para el proyecto.
- Usaremos estos repositorios/proyectos durante el curso.



Referencias

- [1] L.M. de la Cruz Salas, *Git y GitLab para gestión de software*, Reportes Internos 2019-01 / septiembre de 2019. [Descarga](#).
- [2] Baarsen, Jeroen Van. *GitLab Cookbook*. Packt Publishing, 2014.
- [3] Chacon, Scott, and Straub, Ben. *Pro Git*. Apress, 2014.

