

Hay varias cosas para hacer. Están itemizadas, con una explicación después de cada tarea:

- Instalar git.

```
#Para tener las versiones de paquetes más recientes  
sudo apt-get update  
#GIT  
sudo apt-get install git
```

Dicho mal y pronto, git es un programa usado para trabajar en equipo sobre un proyecto (como overleaf/sharelatex, dropbox, etc), pero con algunas diferencias. La más importante es que si uno hace cambios en los archivos, éstos **no se actualizan instantáneamente** en la computadora de la otra persona.

Además, podemos trabajar sobre el mismo archivo (en lugares distintos), y luego combinar los cambios de manera muy fácil.

- Crease una cuenta en <https://github.com/>

El proyecto que hice para el final está alojado (hosteado) en un sitio web - el que más se usa - llamado **github**. Es gratis. Tendrías que hacerte una cuenta ahí así te puedo agregar al proyecto.

(¿Estás de acuerdo con que sea público?)

- Compartirme tu usuario de github por email, así te agrego al proyecto y podés subir cambios.

- Elegir carpeta donde trabajar.

En tercer lugar, tenés que elegir una carpeta en tu máquina donde vas tus proyectos de github. Yo uso **/home/billy/Projects/git/github**, pero no hace falta que sea ni remotamente parecida.

- Bajarse el proyecto de github.

```
#Estate parada en la carpeta donde vas a trabajar  
#La carpeta no debería ser "MESSI"  
#porque este comando te baja todo a una carpeta "MESSI"  
git clone https://github.com/billy-mosse/MESSI.git
```

Deberías ver una carpeta “MESSI” con subcarpetas “code, garbaje, instructions”, etc.

- Instalar la última versión de **python3**:

```
#Creo que viene por default ya, igual.  
sudo apt-get install python3  
#Si ya lo tenías instalado, actualizalo  
sudo apt-get upgrade python3
```

- Instalar **pip**, para bajarse librerías de python, y otros yuyos:

```
sudo apt-get -y install python3-pip
pip3 install --upgrade pip
```

```
#Yo tuve un problema y tuve que hacer esto
pip3 install --upgrade setuptools pip
```

- (Opcional, pero recomendado) Instalar **virtualenv**:

```
#Instalar virtualenv por medio de pip.
python3 -m pip install --user virtualenv
```

Virtualenv sirve para cuando tenés más de un proyecto en python. Tal vez para uno necesitás la versión 3.7 de la librería matplotlib, pero otro es medio antiguo y sólo anda con la versión 2.2. Virtualenv hace que puedas no instalar paquetes **globalmente**.

Cuando uno trabaje en este proyecto, “prende” el entorno virtual de python correspondiente, que va a tener instaladas las versiones específicas de las librerías necesarias. (Es un comando muy simple.)

De nuevo, no hace falta, a mí no me cambia nada si lo usás o no. Es más para uno.

- (Opcional, cont) Crear el virtualenv correspondiente al proyecto, en la misma carpeta del proyecto, o en una carpeta virtualenvs/ en tu home:

```
python3 -m virtualenv env_messi
```

¡Avisame si en vez de env_messi usás otro nombre para el virtualenv, porque en ese caso tengo que hacer algo!

- Prender el entorno virtual:

```
source env_messi/bin/activate
```

- Instalar las librerías que necesita el proyecto:

```
#El doble AND es para correr comandos en serie (uno detras de otro)
#siempre y cuando mp devuelvan error
pip3 install numpy &&\
pip3 install matplotlib &&\
pip3 install pandas &&\
pip3 install networkx
```

- Instalar sage para python:

Bajarse sage de <http://linorg.usp.br/sage/linux/index.html>

Descomprimirlo, y correr:

```
make configure &&\
./configure --with-python=3 &&\
make build
```

<https://packaging.python.org/guides/installing-using-pip-and-virtualenv/> Instrucciones para instalarse el entorno de trabajo:

```
sudo apt-get install python3
```