

UNIVERSIDADE DE SÃO DE PAULO

ENGENHARIA MECATRÔNICA



Primeiros passos com no trabalho com Git

October 14, 2015

1 Como usar o git e o github

1.1 Instalar o Git

1. Primeiramente instalemos o git, ferramenta para o versionamento local, em que você poderá compartilhar de algumas maneiras seus arquivos. Usaremos o github, que permite a criação de repositórios private grátis. Antes de instalar, confirmamos que não esteja previamente instalado digitando `git --version` em um terminal (para abrir um terminal: `ctrl+alt+t`):

```
javier@javier-desktop: /usr/local
javier@javier-desktop:~$ git --version
O programa 'git' não está instalado no momento. Você pode instalá-lo digitando:
sudo apt-get install git
javier@javier-desktop:~$
```

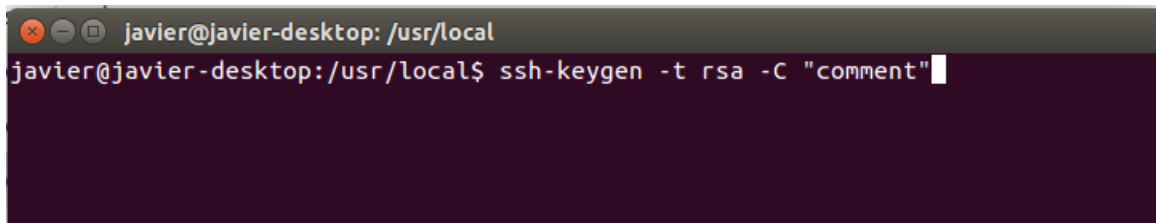
2. Geralmente o Git está instalado em `/etc/local/`. Uma outra forma de saber se está instalado é acessando nesse directório:

```
javier@javier-desktop: /usr/local
javier@javier-desktop:~$ cd /usr/local/
javier@javier-desktop:~$ ls
bin  games      include  man          sbin  src
etc  glassfish-4.1  lib      netbeans-8.0.2  share
javier@javier-desktop:~$
```

3. Para começar a instalação digitar: `sudo apt-get install git-core`:

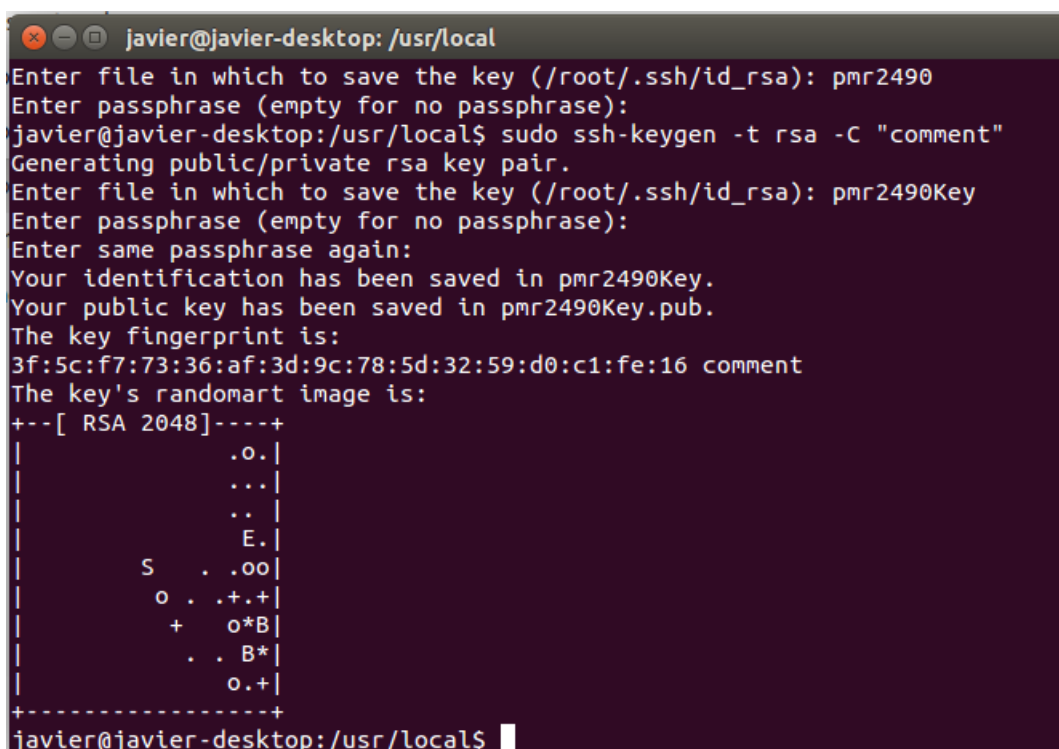
```
javier@javier-desktop: /usr/local
javier@javier-desktop:~$ sudo apt-get install git-core
```

4. É necessário gerar uma chave *ssh* e fazer um cadastro em algum repositório git. Esta etapa é relacionada com a segurança dos repositórios. Para conferir se você já tem alguma chave digitar `ls /.ssh/`, se já existe uma você pode utilizá-la ou simplesmente gerar uma nova digitando `ssh-keygen -t rsa -C "comment"` no directório `/.ssh/` em que "comment" é só um lembrete para saber do que se trata a chave.



```
javier@javier-desktop: /usr/local
javier@javier-desktop: /usr/local$ ssh-keygen -t rsa -C "comment"
```

5. Normalmente para a chave usa-se o seu nome de usuário, ou do serviço que vai usar, por exemplo o github. Em nosso caso usaremos `pmr2490Key` como nome. Além disso deverá especificar uma frase secreta ou password:

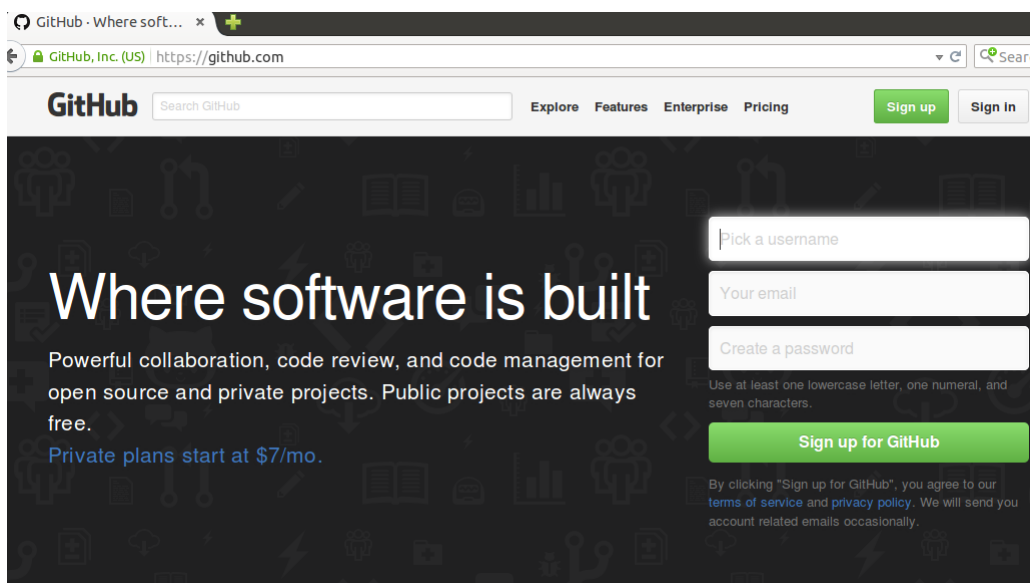


```
javier@javier-desktop: /usr/local
Enter file in which to save the key (/root/.ssh/id_rsa): pmr2490
Enter passphrase (empty for no passphrase):
javier@javier-desktop: /usr/local$ sudo ssh-keygen -t rsa -C "comment"
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): pmr2490Key
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in pmr2490Key.
Your public key has been saved in pmr2490Key.pub.
The key fingerprint is:
3f:5c:f7:73:36:af:3d:9c:78:5d:32:59:d0:c1:fe:16 comment
The key's randomart image is:
+---[ RSA 2048]-----+
|          .o. |
|          ... |
|          ..  |
|           E. |
|      S   .oo |
|     o  .+.+ |
|    +   o*B |
|         . B* |
|          o.+ |
+-----+
javier@javier-desktop: /usr/local$
```

6. Podemos conferir a existencia de nossa nova chave de acesso digitando `ls ~/.ssh/` e a seguir `ll`:

```
javier@javier-desktop:/usr/local$ ls ~/.ssh/
known_hosts
javier@javier-desktop:/usr/local$ ll
total 56
drwxr-xr-x 12 root root 4096 Out 14 12:16 ./
drwxr-xr-x 10 root root 4096 Ago  5 02:12 ../
drwxr-xr-x  2 root root 4096 Ago  5 02:12 bin/
drwxr-xr-x  2 root root 4096 Ago  5 02:12 etc/
drwxr-xr-x  2 root root 4096 Ago  5 02:12 games/
drwxr-xr-x  8 root root 4096 Set  4 16:21 glassfish-4.1/
drwxr-xr-x  2 root root 4096 Ago  5 02:12 include/
drwxr-xr-x  4 root root 4096 Ago  5 02:17 lib/
lrwxrwxrwx  1 root root    9 Set  4 15:34 man -> share/man/
drwxr-xr-x 22 root root 4096 Set  4 16:22 netbeans-8.0.2/
-rw-----  1 root root 1766 Out 14 12:16 pmr2490Key
-rw-r--r--  1 root root  389 Out 14 12:16 pmr2490Key.pub
drwxr-xr-x  2 root root 4096 Ago  5 02:12 sbin/
drwxr-xr-x  7 root root 4096 Ago  5 02:18 share/
drwxr-xr-x  2 root root 4096 Ago  5 02:12 src/
javier@javier-desktop:/usr/local$
```

7. Neste ponto devemos criar nossa conta github.

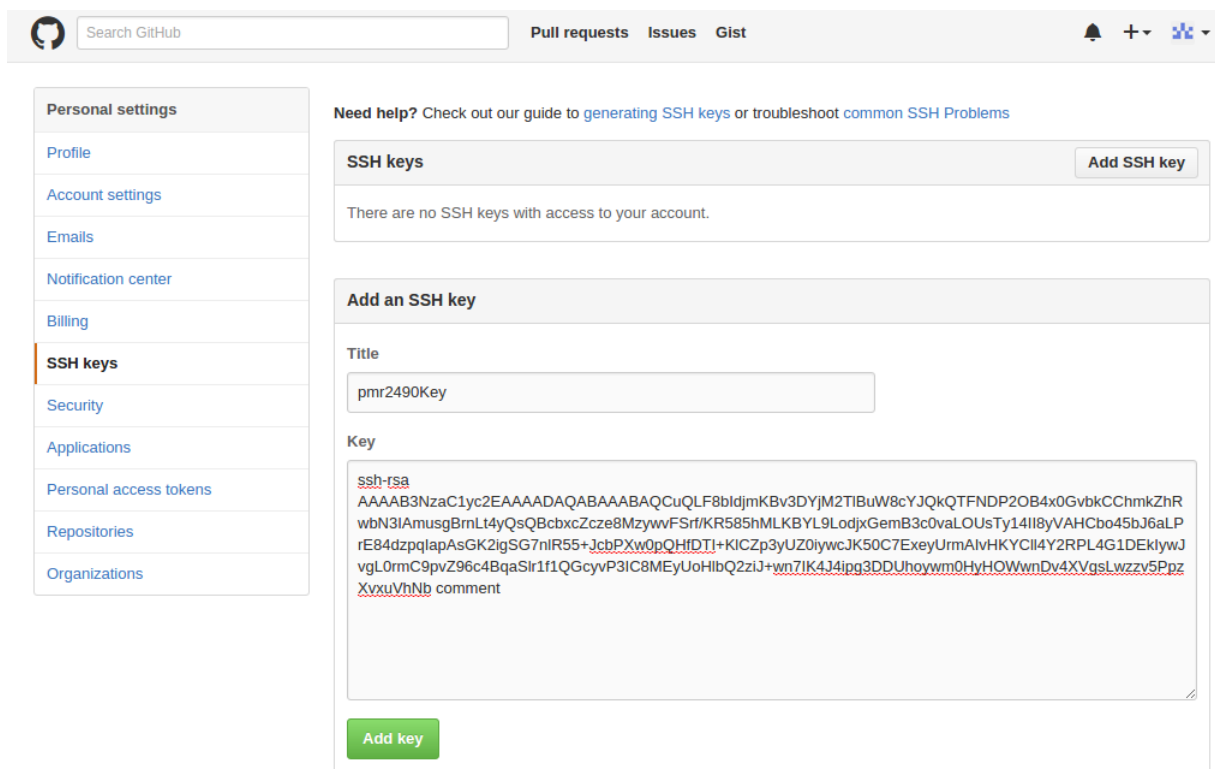


8. Uma vez logado, acessar <https://github.com/account> e clicar em "SSH Public Keys" e depois em "add another public key". Como a cópia da chave precisa ser exata sugere-se primeiro instalar o **xclip**: `sudo apt-get install xclip`; e depois fazer a copia digitando na linha de comando `cat /usr/local/pmr2490Key.pub | xclip -sel clip`

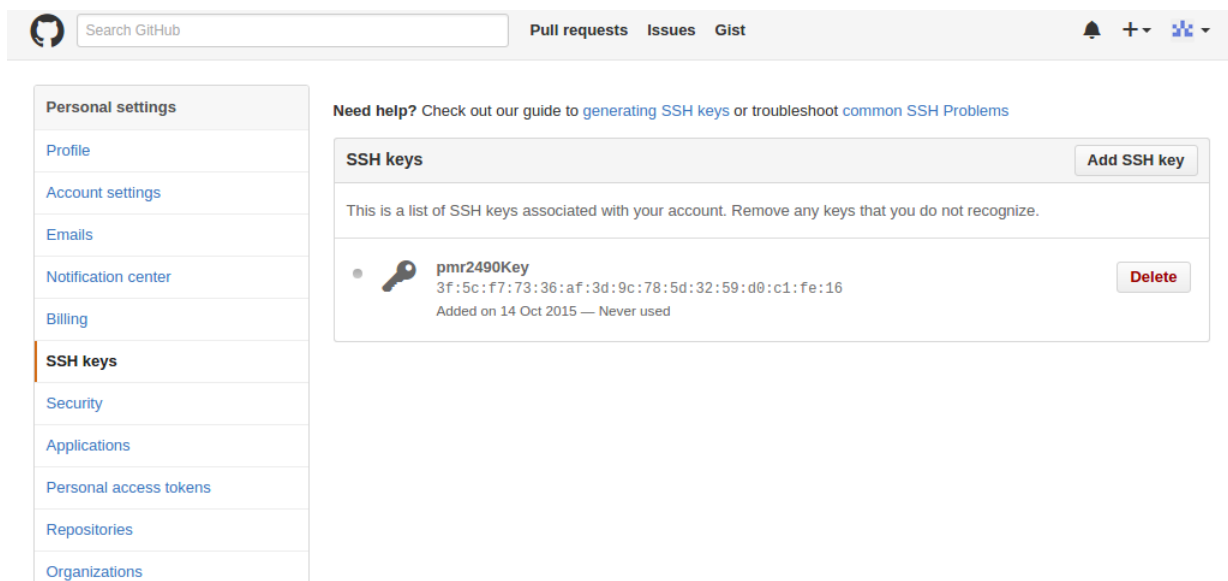
```
javier@javier-desktop: /usr/local
javier@javier-desktop:/usr/local$ sudo apt-get install xclip
```

```
javier@javier-desktop: /usr/local
javier@javier-desktop:/usr/local$ sudo cat /usr/local/pmr2490Key.pub | xclip -sel clip
```

9. Para acrescentar chave no github só resta fazer `ctrl+v`:



10. O GitHub confirmará a inserção da nova chave, sendo também notificado via email .

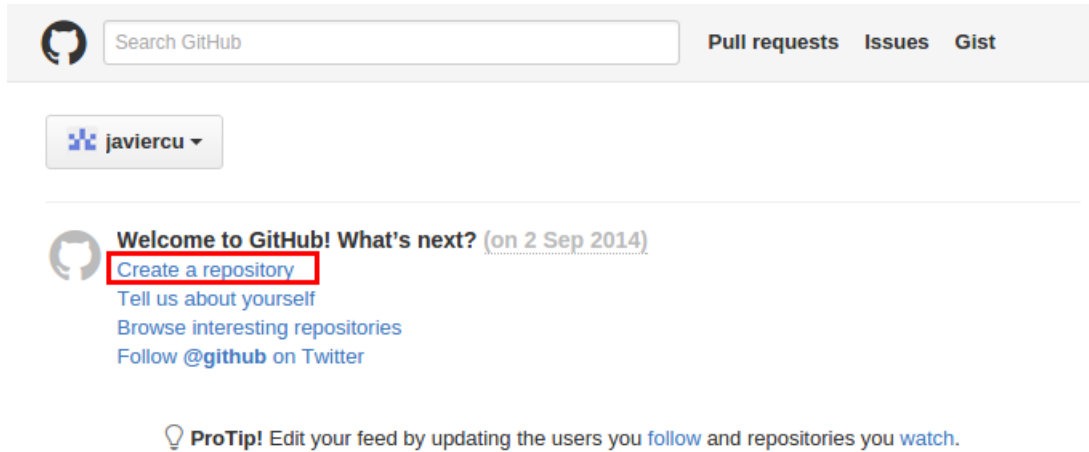


11. Por padrão o git vai pegar o usuário do sistema, para que seu nome de usuário do github apareça corretamente use os comandos:

```
javier-desktop: /usr/local
javier@javier-desktop:/usr/local$ git config --global user.name "Your Name"
javier@javier-desktop:/usr/local$ git config --global user.email codexico@gmail.com
```

1.2 Criando o nosso primeiro Projeto no github

1. Poderíamos criar um novo projeto ou simplesmente trabalhar sobre um já existente. Para criar um novo vá até o github e no alto da página clique em *"Dashboard"* e depois em *"Create a repository"* .



- (a) Criar um espaço de trabalho para o projeto em nosso computador (pmr2490Project):

```
javier-desktop: ~/Documentos/pmr2490Project
javier@javier-desktop:~/Documentos$ mkdir pmr2490Project
javier@javier-desktop:~/Documentos$ cd pmr2490Project/
javier@javier-desktop:~/Documentos/pmr2490Project$ █
```

2. Iniciar um git neste diretório digitando *git init*:

```
javier-desktop: ~/Documentos/pmr2490Project
javier@javier-desktop:~/Documentos/pmr2490Project$ git init
```

3. A saída de aplicar este comando seria:

```
javier@javier-desktop:~/Documentos/pmr2490Project$ git init
Initialized empty Git repository in /home/javier/Documentos/pmr2490Project/.git/
```

(a) Um diretório oculto `“.git”` deve aparecer. Neste `.git` ficam as configurações que serão usadas para este projeto. Para conferir digite o comando `ll`:

```
javier@javier-desktop:~/Documentos/pmr2490Project$ ll
total 12
drwxrwxr-x 3 javier javier 4096 Out 14 14:26 ./
drwxr-xr-x 7 javier javier 4096 Out 14 14:37 ../
drwxrwxr-x 7 javier javier 4096 Out 14 14:26 .git/
javier@javier-desktop:~/Documentos/pmr2490Project$
```

4. Acessando ao conteúdo deste diretório de configuração aparecerá:

```
javier@javier-desktop:~/Documentos/pmr2490Project$ ls .git/
branches  config  description  HEAD  hooks  info  objects  refs
javier@javier-desktop:~/Documentos/pmr2490Project$
```

5. Neste ponto podemos adicionar o repositório. No exemplo usaremos o repositório criado no passo 1. É possível acessar ao endereço na página do projeto no GitHub (neste caso <https://github.com/javiercu/pmr2490.git>):

```
javier@javier-desktop:~/Documentos/pmr2490Project$ git remote add origin git@github.com:javiercu/pmr2490.git
```

Formato do comando:

- *"git remote add"*: Adiciona um repositório ao git que foi iniciado neste diretório.
- *"origin"*: É o apelido para o projeto,
- *"https://github.com/javiercu/pmr2490.git"* é o endereço do projeto.

6. Digitando `cat .git/config` é possível consultar as configurações do projeto.

```
javier@javier-desktop:~/Documentos/pmr2490Project$ cat .git/config
[core]
  repositoryformatversion = 0
  filemode = true
  bare = false
  logallrefupdates = true
[remote "origin"]
  url = https://github.com/javiercu/pmr2490.git
  fetch = +refs/heads/*:refs/remotes/origin/*
javier@javier-desktop:~/Documentos/pmr2490Project$
```

7. Para baixar (pull=puxar) o repositório para nosso projeto local: `git pull origin master`

```
javier@javier-desktop:~/Documentos/pmr2490Project$ git pull origin master
remote: Counting objects: 54, done.
remote: Compressing objects: 100% (8/8), done.
remote: Total 54 (delta 11), reused 5 (delta 5), pack-reused 41
Unpacking objects: 100% (54/54), done.
From github.com:codexico/tutorial-github
 * branch                master       -> FETCH_HEAD
 * [new branch]         master       -> origin/master
javier@javier-desktop:~/Documentos/pmr2490Project$ ~
```

1.3 Usando o git

1. Criar um documento `README.md`:

```
javier@javier-desktop:~/Documentos/pmr2490Project$ echo "# pmr2490" >> README.md
```

2. A seguir acrescentamos o documento em nosso repositório: `git add README.md`, fazemos o commit `git commit -m "first commit"` e depois o `push` definitivo:

(a) Criar um documento `README.md`:

```
javier@javier-desktop:~/Documentos/pmr2490Project$ echo "# pmr2490" >> README.md
javier@javier-desktop:~/Documentos/pmr2490Project$ git init
Reinitialized existing Git repository in /home/javier/Documentos/JavGitHub/.git/
javier@javier-desktop:~/Documentos/pmr2490Project$ git add README.md
javier@javier-desktop:~/Documentos/pmr2490Project$ git commit -m "first commit"
[master (root-commit) e06311e] first commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.md
javier@javier-desktop:~/Documentos/pmr2490Project$ git remote add origin git@github.com:javiercu/pmr2490.git
fatal: remote origin already exists.
javier@javier-desktop:~/Documentos/pmr2490Project$ git push -u origin master
Counting objects: 3, done.
Writing objects: 100% (3/3), 217 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To git@github.com:javiercu/pmr2490.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin.
```

A sequencia de comandos correta é:

- `git add README.md`
 - `git commit -m "first commit"`
 - `git remote add origin https://github.com/javiercu/pmr2490.git`
 - `git push -u origin master`
3. Se durante o tempo em que é feito o `pull` e o `push` feito por uma outra pessoa que também participe do projeto fez alterações o `push` será rejeitado. Então é necessário atualizar o projeto local antes de enviar novas alterações: `git fetch origin`.
Uma boa prática é atualizar antes de enviar é uma boa prática. No caso do git a é obrigatória.
 4. Pronto, para conferir as alterações no navegador acessando o endereço do projeto (<https://github.com/javiercu/p> neste exemplo).