

Git – používateľ'ská príručka

Príručka je zameraná na potreby nášho tímu, teda nevenuje sa gitu komplexne.

1) Stiahnite si a nainštalujte Git:

a) návod pre Windows <http://help.github.com/win-set-up-git/>

b) návod pre Linux <http://help.github.com/linux-set-up-git/>

c) návod pre OSX <http://help.github.com/mac-set-up-git/>

Pri inštalácii si nezabudnite zaškrtnúť commitovanie Unix-style koncov riadkov, aby sme boli jednotní:

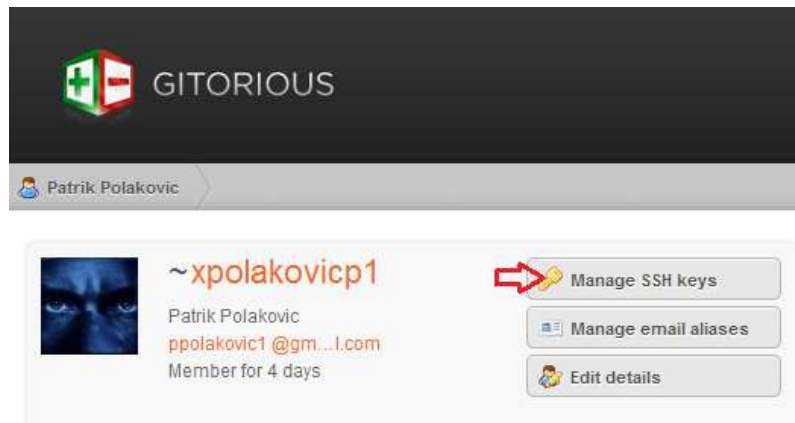


2) Pre autentifikáciu potrebujete podpisové kľúče, jeden privátny na podpisovanie a druhý verejný na overovanie. Kľúče vygenerujete zadaním nasledovného príkazu do Git bash konzoly:

```
ssh-keygen -t rsa -C "your_email@youremail.com"
```

Pre uloženie kľúča stlačte **Enter**, a kľúče sa vám uložia do `~/ .ssh` priečinka. Pre zvýšenie bezpečnosti môžete zadať `passphrase`.

3) Skopírujte obsah súboru `~/ .ssh/id_rsa.pub`. Prihláste sa na <http://gitbus.fiit.stuba.sk/> a v osobnom profile kliknite na **Manage ssh keys**:



Následne vyberte možnosť v pravom paneli Add SSH key.

Skopírujte obsah vášho verejného kľúča do textového pol'a a uložte kľúč:

Add a new public SSH key

Your public key

It is generally located in ~/.ssh/id_rsa.pub or ~/.ssh/id_dsa.pub. If you want to use multiple keys you will need to create a new key. The key should be in the format of:

ssh-`algorithm` `base64-content` `you@somehost`

```
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAujnVJZiSCWsKbP00Lgre/EF
/Bjqv6Pj8DV1HF4JI0tmJjA/8XD83DeAuoF2kk3sZbj5j5wLePKtcD
/45HijMwdNEK8ppUiqOayWX8UvwVklzSBkz9h1XFTo26F8v+S5WVm2no7zIRYUJ5
HKKEhYOotGeBhlcj85dQhg2upwkPdn3dn
/peL90Y8ZQYmInTkgIjaAlso30bJWohlri5SUI78I3qdz8NNT7AkBLroM4tMsPrRdlkMI
yjrMdrqVACBCUEsSYagyjW/epnfUWNjyKovgsq3yq3s3L294Q== timovy-projekt-
16@googlegroups.com
```

Save

4) Na gitbuse je už vytvorený repozitár `znalosti-a-zrucnosti-studentov`. Tento repozitár si naklonujte do vašich PC, do priečinku `~/` zadaním príkazu:

[git@gitbus.fiit.stuba.sk:znalosti-a-zrucnosti-studentov/znalosti-a-zrucnosti-studentov.git](https://gitbus.fiit.stuba.sk:znalosti-a-zrucnosti-studentov/znalosti-a-zrucnosti-studentov.git)

Tiež si vyplňte meno a mailovú adresu pre potreby commitovania:

```
git config --global user.name "Firstname Lastname"
git config --global user.email "your_email@youremail.com"
```

5) Teraz si vysvetlíme 2 pojmy. A to `master` a `origin`. `origin` môžeme chápať ako hlavné úložisko, teda server gitbus. `Master` je hlavná vetva nášho repozitáru. Vetvenie je dobré pri verziovaní projektu. Nová vetva sa dá vytvoriť príkazom:

```
(~..@MUCHA ~/znanosti-a-zrucnosti-studentov (master)
$ git branch mybranch
```

Prepínanie medzi vetvami je nasledovné (zmení aktuálnu vetvu na `mybranch`):

```
(~..@MUCHA ~/znanosti-a-zrucnosti-studentov (master)
$ git checkout mybranch
Switched to branch 'mybranch'
(~..@MUCHA ~/znanosti-a-zrucnosti-studentov (mybranch)
```

6) Pridávanie zmien do vetvy bude prebiehať nasledovne:

Zmodifikujem ľubovoľný súbor a potvrdím prijatie zmien pre všetky zmodifikované súbory:

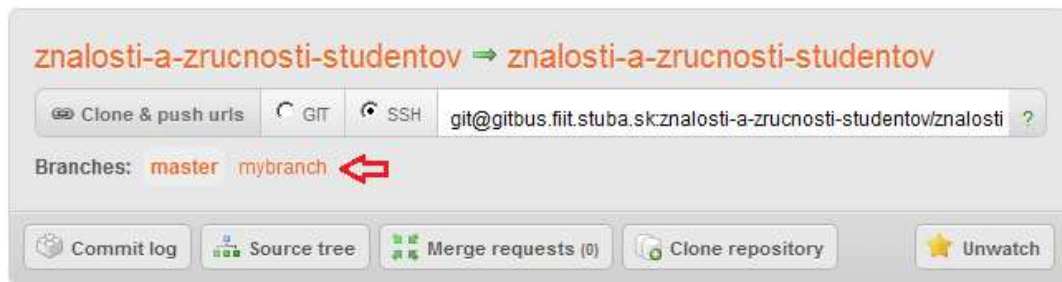
```
(~..@MUCHA ~/znanosti-a-zrucnosti-studentov (mybranch)
$ echo "mybranch test" >> README
```

```
(~..@MUCHA ~/znanosti-a-zrucnosti-studentov (mybranch)
$ git commit -a -m "modifikacia README suboru"
warning: LF will be replaced by CRLF in README.
The file will have its original line endings in your working directory.
[mybranch warning: LF will be replaced by CRLF in README.
The file will have its original line endings in your working directory.
d6cc3ee] modifikacia README suboru
warning: LF will be replaced by CRLF in README.
The file will have its original line endings in your working directory.
1 files changed, 1 insertions(+), 0 deletions(-)
```

Potom odošlem zmeny na server:

```
(~..@MUCHA ~/znanosti-a-zrucnosti-studentov (mybranch)
$ git push origin mybranch
Counting objects: 5, done.
Writing objects: 100% (3/3), 270 bytes, done.
Total 3 (delta 0), reused 0 (delta 0)
remote: => Syncing Gitorious... [OK]
To git@gitbus.fiit.stuba.sk:znanosti-a-zrucnosti-studentov/znanosti-a-zrucnosti-studentov.git
1a6fdf6..d6cc3ee mybranch -> mybranch
```

Zmeny sa mi okamžite prejavajú na gitbuse:



Odstránenie remote vetvy:

```
(~..@MUCHA ~/znanosti-a-zrucnosti-studentov (master)
$ git push origin :mybranch
remote: => Syncing Gitorious... [OK]
To git@gitbus.fiit.stuba.sk:znanosti-a-zrucnosti-studentov/znanosti-
studentov.git
- [deleted]          mybranch
```

7) Aktualizovanie lokálnej vetvy. To znamená, že ak niekto vykoná zmeny na projekte a odošle ich, tak pre aktualizáciu vašej vetvy musíte tieto zmeny uplatniť u vás nasledovným príkazom:

```
(~..@MUCHA ~/znanosti-a-zrucnosti-studentov (master)
$ git pull
Already up-to-date.
```

Pri konfliktoch je lepšie používať kombináciu **fetch/merge**, kedy si môžete manuálne riešiť vzniknuté konflikty (príkaz **merge** aplikuje zmeny projektu na zadanú vetvu - **master**):

```
(~..@MUCHA ~/znanosti-a-zrucnosti-studentov (master)
$ git fetch origin

(~..@MUCHA ~/znanosti-a-zrucnosti-studentov (master)
$ git merge origin/master
Already up-to-date.
```

8) GUI. Git pre Windows aj Mac má aj GUI ktoré poskytuje všetky vyššie opísané veci a mnoho ďalších, navyše sa v ňom veľmi ľahko orientuje.

Poznámky:

Nepopísal som úplne všetky príkazy gitu, ktoré možno pri projekte využijeme. Pri nepochopení niektorých príkazov resp. ich nesprávneho fungovania a všetky vzniknuté problémy spojené pri používaní gitu navrhujem riešiť "on the fly".