|  |  |
| --- | --- |
|  | **2014** |
|  | The Company  Martien Huijsmans |

|  |
| --- |
| **[GIT]** |
| Installation, configuration and management |

Inhoud

[GIT on Ubuntu 1](#_Toc387436658)

[Typography 1](#_Toc387436659)

[Installation of a GIT server on Ubuntu 1](#_Toc387436660)

[Setup a repository to hold git projects 2](#_Toc387436661)

[Open issues 2](#_Toc387436662)

[Documentation 2](#_Toc387436663)

[GITWEB on Ubuntu 2](#_Toc387436664)

[Import project/repo 3](#_Toc387436665)

[SSH Server on Ubuntu 3](#_Toc387436666)

[Developer Machine 3](#_Toc387436667)

[Eclipse Standard 3](#_Toc387436668)

[MAVEN 3](#_Toc387436669)

[GIT 4](#_Toc387436670)

[Opening a MAVEN project from GIT repository 4](#_Toc387436671)

[SourceTree: GIT client (Mac & Windows) 4](#_Toc387436672)

[Project: TestApp3 4](#_Toc387436673)

[Ubuntu 4](#_Toc387436674)

[Developer machine 4](#_Toc387436675)

[Miscellaneous 5](#_Toc387436676)

[Bad file number 5](#_Toc387436677)

# GIT on Ubuntu

## Typography

Commands that are executed in a terminal are preceded with a $.  
For example: $ sudo apt-get install ….

## Installation of a GIT server on Ubuntu

Guide: <https://help.ubuntu.com/13.10/serverguide/git.html>  
Used sections: install, configuration, basic usage  
After this GIT runs at boot.

Also install gitk (UI for git)  
$ sudo apt-get install gitk

## Setup a repository to hold git projects

Choose: /home/martien/gitrepo

## Open issues

* Ability to send email. Requires SMTP server.
* Location of got-repository, e.g. in /opt/git
* Separate userid for git?
* Do all user use the same account to commit, or different accounts?
* GitServer needs a dns.name
* Workflow: <https://github.com/nvie/gitflow>
* A collection of useful .gitignore templates: <https://github.com/github/gitignore>
* Redmine as GIT frontend:  
  <http://stackoverflow.com/questions/438163/web-interface-for-git-repositories>

## Documentation

GIT commands:

* <http://www.siteground.com/tutorials/git/commands.htm>   
  This one also mentions, support tools like gitk and wbserver.
* Full list:  
  <https://www.kernel.org/pub/software/scm/git/docs/>

GIT best practices

* Use a single sandbox repo where you use directories to hold different projects
* Discussion of using different project & use of GIT Modules is described/ discussed here:  
  <http://stackoverflow.com/questions/5514739/best-practice-for-git-repositories-with-multiple-projects-in-traditional-n-tier?rq=1>

# GITWEB on Ubuntu

GIT web provides a web based interface for the GIT server.  
After installation it is started at boot.

* Installation: sudo apt-get install gitweb
  + This also installed apache (I think)
  + And it created file: /etc/apache2/conf.d/gitweb
* Created gitrepository: /home/martien/gitrepo
  + And 2 projects project1.git and project2.git in this dir
  + For each project:  
    cd <projectname>  
    git install –bare –shared
* sudo gedit /etc/apache2/conf.d/gitweb
  + Update: Alias /gitweb /usr/….  
    to: Alias /**git** /usr/……  
    Now gitrepo can be access under: [http://localhost/**git**](http://localhost/git)
* Sudo gedit /etc/gitweb.conf
  + Update $projectroot to point to git repo (see previous bullet).
* Restart apache: sudo /etc/init.d/apache2 force-reload
* Now access the git repo: <http://localhost/git>

## Import project/repo

ssh://<user>@guestserver.com/<project-name>

# SSH Server on Ubuntu

To access the git server via SSH, install a sshserver  
<https://help.ubuntu.com/community/SSH/OpenSSH/Configuring>   
I disables passwords as suggested.

Then I followed this guide to generate and install a ssh\_key (from my machine)on the VM  
<https://help.ubuntu.com/community/SSH/OpenSSH/Keys>   
One problem: (1) howto get my public\_key to VM and (2) where to store it.  
1) I use email  
2) I did “cat id\_rsa.pub >> authorized\_keys” as described in above link, updated the /etc/ssh/sshd\_config to uncomment the #AuthorizedKeysFile ……   
Note: the authorized\_key didn’t exist yet.

To install this software I needed to set the Virtual Image network to NAT.

# Developer Machine

## Eclipse Standard

EGIT for Eclipse  
Already installed on Java-Eclipse on windows.

## MAVEN

Maven 3.1.1 installed on Windows.

## GIT

Install Git for windows as recommended by gitbook:  
<http://msysgit.github.com/>   
Gives GUI & commandline.

Observation: I installed maven the windows way. But I can run maven command also from inside the Linux style GUI that is part of msysgit.

## Opening a MAVEN project from GIT repository

1. Opened GIT repository (on local disk).
2. Open project in from GIT repository (dir -> open repository -> import existing projects)
3. Next converted the project to maven   
   “Configure->Convert to Maven Project” and “Maven->Update Project”.
4. After this I could  
   run as: Java Application / JUnit test).
5. Used ref: <http://stackoverflow.com/questions/4869815/importing-a-maven-project-into-eclipse-from-git> , answer-0.

## SourceTree: GIT client (Mac & Windows)

Link: <http://sourcetreeapp.com/> (has expiry and requires registration).  
<https://code.google.com/p/tortoisegit/>   
<http://www.syntevo.com/smartgithg/>

# Project: TestApp3

This chapter described the client and server actions to define a new project.  
<http://stackoverflow.com/questions/315911/git-for-beginners-the-definitive-practical-guide?rq=1>

## Ubuntu

Create <project-name>.git in gitrepo.  
$ mkdir <project-name>.git  
$ cd <project-name>.git  
Create an empty shared directory  
$ git init --bare --shared

## Developer Machine Eclipse with EGIT

First created a workspace

Next opened GIT perspective and cloned project:  
ssh://<user>@guestserver.com/home/<home-user>/gitrepo/<project>.git  
I cloned this into my workspace.

Next I imported the maven projects into eclipse.

Note: an existing git project can be “imported”

* Open GIT perspective
* Open en local git repo
* Projects can be imported directly from the GIT perspective

## Developer machine Commandline

Used create\_jar.bat to created a maven project: TestApp3.  
  
Open “GIT Bash” in new directory  
Initialize git. Creates a .git dir. $ git init  
Note: “rm -rf .git/” wil delete the .git directory (repo) with all GIT data (not the sources)  
  
Add all file to stage: $ git add .  
Note the .  
  
Next (global settings)  
$ git config –global user.email “…”  
$ git config –global user.name “…”  
  
Commit changes to local git repo: git commit  
Open vim editor.  
Inset line with o or O  
Enter commit message, press <esc> when ready.  
Next quit & save: :wq

Remote git server:

* Define  
  $ git remote add gs [<user>@guestserver.com:/home/<user>/gitrepo/TestApp3.git](mailto:martien@guestserver.com:/home/martien/gitrepo/TestApp3.git)  
  Note: gs is the short name for the remote git repo.
* To list   
  $ git remote -v

Push to remote server (master branch)  
$ git push gs master  
Note: this time no logfile

# Miscellaneous

## Bad file number

$ git clone [<user>@guestserver.com:/home/<user>/testapp1.git](mailto:martien@guestserver.com:/home/mahu/testapp1.git)  
Cloning into 'testapp1'...  
ssh: connect to host guestserver.com port 22: Bad file number  
fatal: Could not read from remote repository.  
Please make sure you have the correct access rights and the repository exists.

Ref: <http://stackoverflow.com/questions/7144811/git-ssh-error-connect-to-host-bad-file-number>