

Lheidar

DevOps – Week 1

Task 2 – Terminal Exploration

Putty

Connection host:

ubuntu@ec2-35-176-22-231.eu-west-2.compute.amazonaws.com

Moving File Directories:

```
mv /home/ubuntu/QACLouy/sample.txt /home/ubuntu/QACLouy/MyProject/
```

Current Directory Location:

```
pwd
```

VI Editor:

I = Insert into VI text Editor

: enter to enter the command mode

wq! = Force save changes to file

View date within text file:

```
cat sample.txt
```

Create new text file:

```
cat> sample.txt
```

Task 3 – Creating a Script file

Force set username:

```
sudoadduser --force-badnameLheidar
```

Add user to sudo group:

```
sudoadduser<username>sudo
```

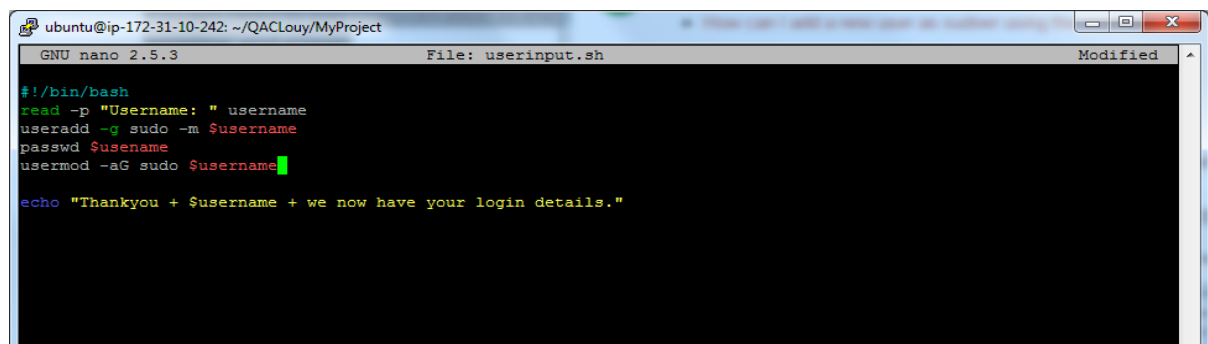
Read/write permission to file being accessed

```
chmod +x file_name
```

Add user to sudoers group

```
sudousermod -aGsudo<username>
```

Adding User Permission & Creating User



The screenshot shows a terminal window with the title bar 'ubuntu@ip-172-31-10-242: ~/QACLouy/MyProject'. The terminal is running the GNU nano 2.5.3 editor, editing a file named 'userinput.sh'. The script content is as follows:

```
#!/bin/bash
read -p "Username: " username
useradd -g sudo -m $username
passwd $username
usermod -aG sudo $username

echo "Thankyou + $username + we now have your login details."
```

Lheidar

Task 4 – Configuring the Linux Environment

<https://www.vultr.com/docs/how-to-install-apache-maven-on-ubuntu-16-04>

Java Installation:

```
louisheidar@ubuntu: ~  
louisheidar@ubuntu:~$ sudo nano /etc/profile.d/mavenenv.sh  
[sudo] password for louisheidar:  
louisheidar@ubuntu:~$ java -version  
java version "1.8.0_131"  
Java(TM) SE Runtime Environment (build 1.8.0_131-b11)  
Java HotSpot(TM) 64-Bit Server VM (build 25.131-b11, mixed mode)  
louisheidar@ubuntu:~$
```

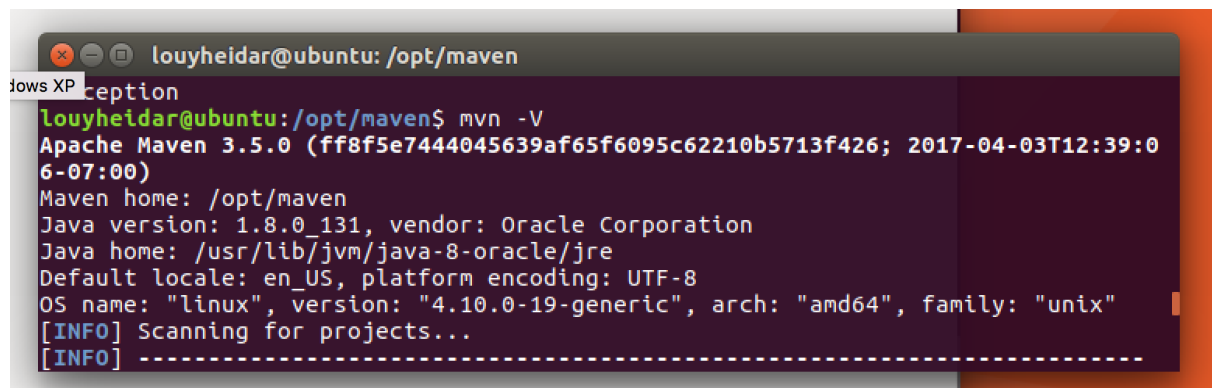
Maven Installation: Setting Environment Variables

```
GNU nano 2.7.4 File: /etc/profile.d/mavenenv.sh  
export M2_HOME=/home/louisheidar/Downloads/maven  
export PATH=$M2_HOME/bin:$PATH  
  
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify  
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Linter
```

Replacing sudo with source to run script file
source /etc/profile.d/mavenenv.sh

Lheidar

Maven Installation:

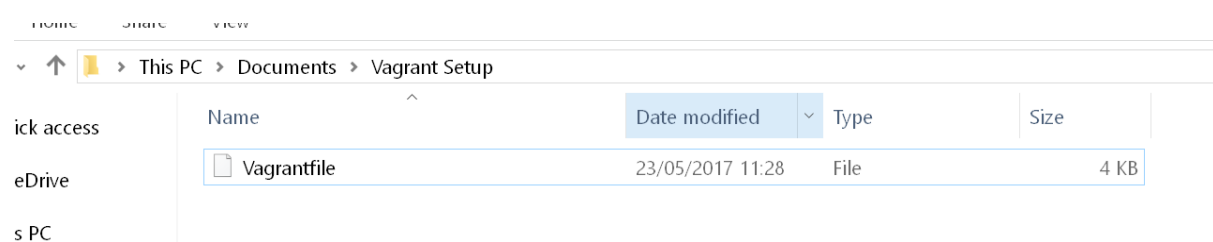


```
louis XPception
louyheidar@ubuntu: /opt/maven$ mvn -V
Apache Maven 3.5.0 (ff8f5e7444045639af65f6095c62210b5713f426; 2017-04-03T12:39:06-07:00)
Maven home: /opt/maven
Java version: 1.8.0_131, vendor: Oracle Corporation
Java home: /usr/lib/jvm/java-8-oracle/jre
Default locale: en_US, platform encoding: UTF-8
OS name: "linux", version: "4.10.0-19-generic", arch: "amd64", family: "unix"
[INFO] Scanning for projects...
[INFO] -----
```

Task 6 – Vagrant Scripting

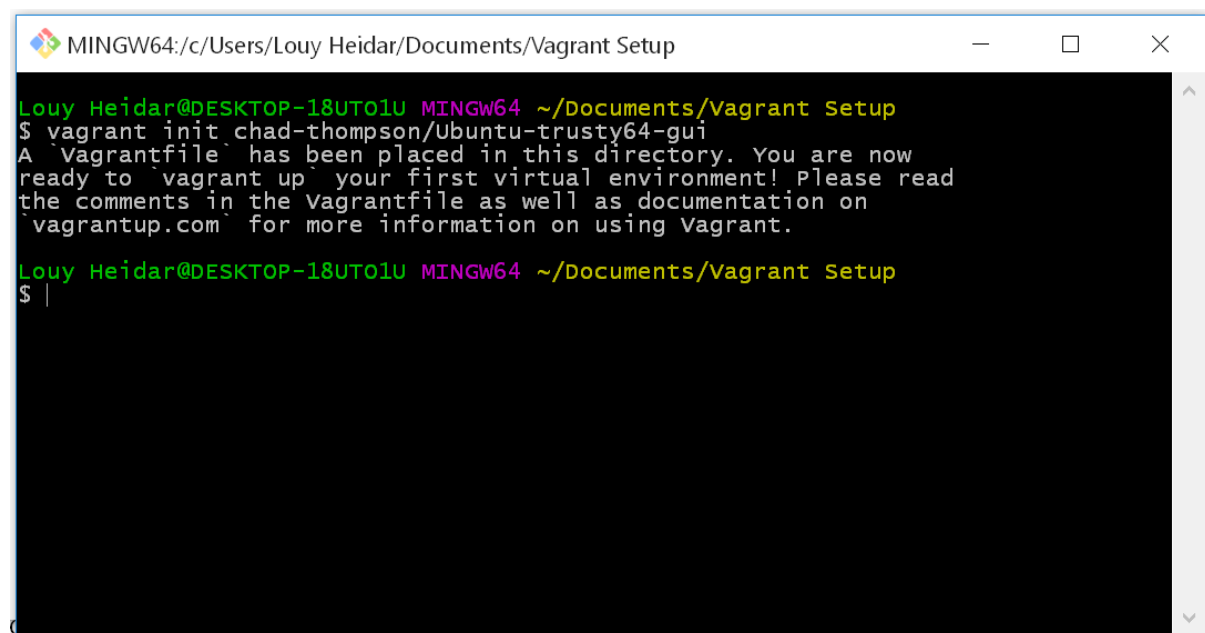
<http://tech.osteel.me/posts/2015/01/25/how-to-use-vagrant-on-windows.html>

Vagrant file added to Vagrant Setup directory



Vagrant file generated using vagrant init command

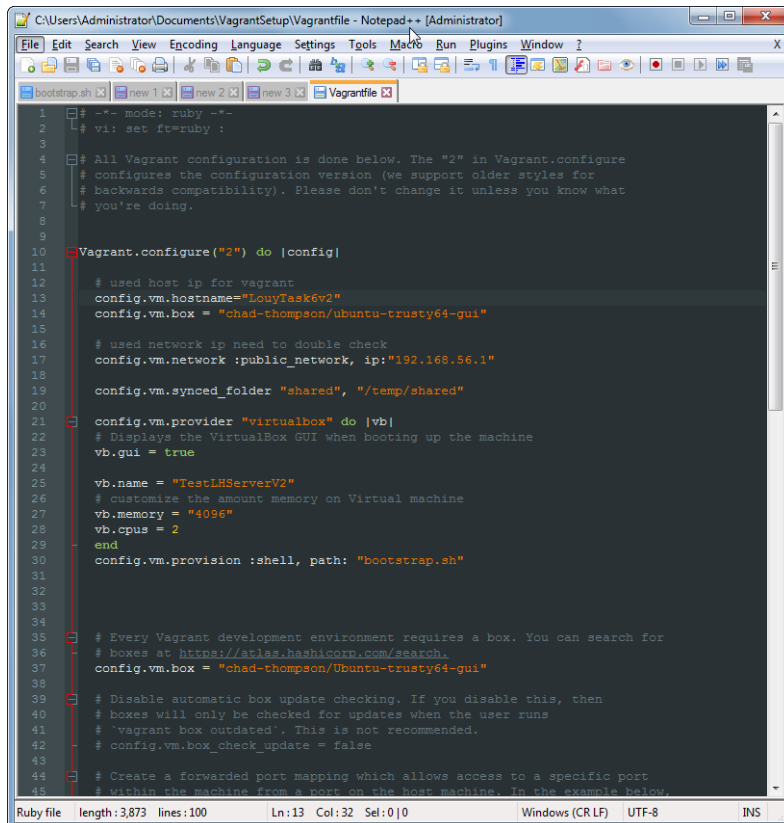
\$vagrant up



```
MINGW64:/c:/Users/Louy Heidar/Documents/Vagrant Setup
Louy Heidar@DESKTOP-18UTO1U MINGW64 ~/Documents/Vagrant Setup
$ vagrant init chad-thompson/Ubuntu-trusty64-gui
A Vagrantfile has been placed in this directory. You are now
ready to `vagrant up` your first virtual environment! Please read
the comments in the Vagrantfile as well as documentation on
`vagrantup.com` for more information on using Vagrant.
Louy Heidar@DESKTOP-18UTO1U MINGW64 ~/Documents/Vagrant Setup
$ |
```

Lheidar

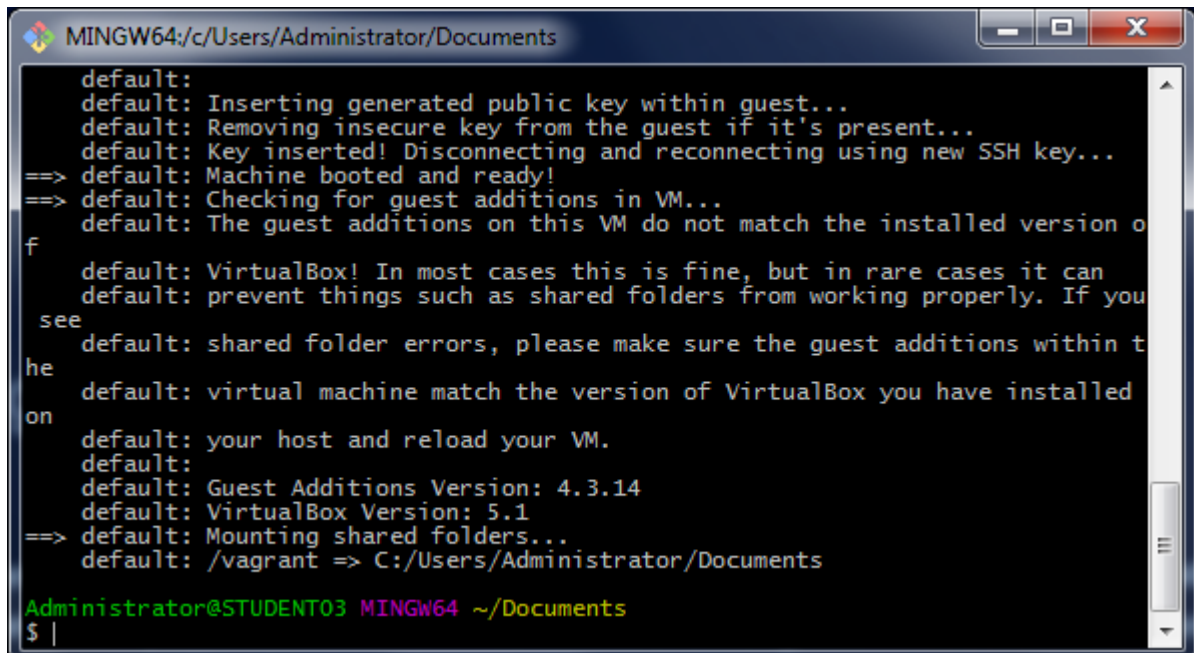
Configuring vagrant file set properties and copy content within shared folder:



The screenshot shows a Notepad++ window titled "C:\Users\Administrator\Documents\VagrantSetup\Vagrantfile - Notepad++ [Administrator]". The window contains a Vagrantfile configuration for a VirtualBox VM named "TestLHServerV2". The configuration includes settings for the VM's hostname, network, synced folders, provider, and memory. It also includes a provision script named "bootstrap.sh".

```
1 # -*- mode: ruby -*-
2 # vi: set ft=ruby :
3
4 # All Vagrant configuration is done below. The "2" in Vagrant.configure
5 # configures the configuration version (we support older styles for
6 # backwards compatibility). Please don't change it unless you know what
7 # you're doing.
8
9
10 Vagrant.configure("2") do |config|
11
12   # used host ip for vagrant
13   config.vm.hostname = "LouyTask6v2"
14   config.vm.box = "chad-thompson/ubuntu-trusty64-gui"
15
16   # used network ip need to double check
17   config.vm.network :public_network, ip: "192.168.56.1"
18
19   config.vm.synced_folder "shared", "/temp/shared"
20
21   config.vm.provider "virtualbox" do |vb|
22     # Displays the VirtualBox GUI when booting up the machine
23     vb.gui = true
24
25     vb.name = "TestLHServerV2"
26     # customize the amount memory on Virtual machine
27     vb.memory = "4096"
28     vb.cpus = 2
29   end
30   config.vm.provision :shell, path: "bootstrap.sh"
31
32
33
34
35   # Every Vagrant development environment requires a box. You can search for
36   # boxes at https://atlas.hashicorp.com/search.
37   config.vm.box = "chad-thompson/Ubuntu-trusty64-gui"
38
39   # Disable automatic box update checking. If you disable this, then
40   # boxes will only be checked for updates when the user runs
41   # 'vagrant box outdated'. This is not recommended.
42   # config.vm.box_check_update = false
43
44   # Create a forwarded port mapping which allows access to a specific port
45   # within the machine from a port on the host machine. In the example below,
```

Virtual Machine Created and added to virtual box

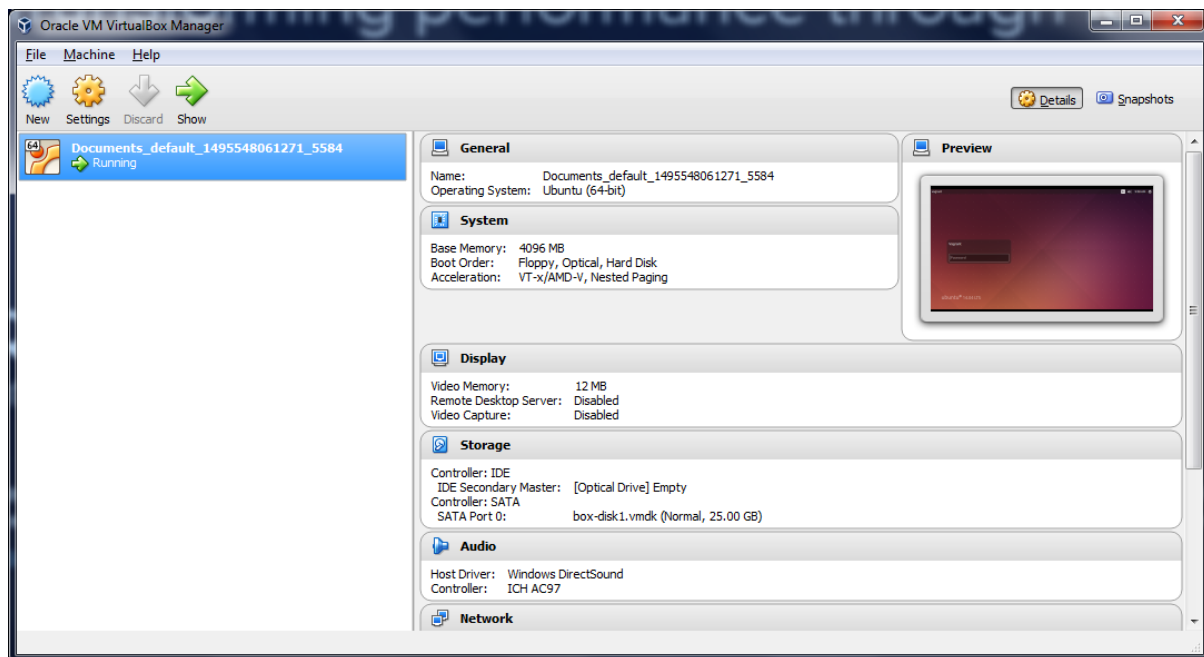


The screenshot shows a MINGW64 terminal window titled "MINGW64:/c:/Users/Administrator/Documents". The terminal displays the output of a Vagrant command, showing the process of inserting a public key, removing an insecure key, and mounting shared folders. The output indicates that the VM is ready and the shared folders are mounted.

```
default:
default: Inserting generated public key within guest...
default: Removing insecure key from the guest if it's present...
default: Key inserted! Disconnecting and reconnecting using new SSH key...
==> default: Machine booted and ready!
==> default: Checking for guest additions in VM...
default: The guest additions on this VM do not match the installed version of
f
default: VirtualBox! In most cases this is fine, but in rare cases it can
default: prevent things such as shared folders from working properly. If you
see
default: shared folder errors, please make sure the guest additions within t
he
default: virtual machine match the version of VirtualBox you have installed
on
default: your host and reload your VM.
default:
default: Guest Additions Version: 4.3.14
default: VirtualBox Version: 5.1
==> default: Mounting shared folders...
default: /vagrant => C:/Users/Administrator/Documents

Administrator@STUDENT03 MINGW64 ~/Documents
$ |
```

Lheidar



Installing Java, Maven, Git and Jenkins using a Shell script

The screenshot shows a Notepad++ window titled 'C:\Users\Administrator\Documents\VagrantSetup\bootstrap.sh - Notepad++ [Administrator]'. The script is a shell script that performs the following steps:

```
1 # 1) Copying the files from the shared directory
2
3 echo "Copying files from shared directory"
4 cd /temp/shared
5 sudo scp java.tar.gz /opt/
6 sudo scp maven.tar.gz /opt/
7 sudo scp jenkins_2.1_all.deb /home/vagrant/Desktop
8 sudo scp jira.bin /opt/
9 sudo scp response.varfile /opt/
10 cd /opt/
11
12 # 2) Installing Java and Maven
13 echo "Installing Java and Maven..."
14 sudo apt-get update
15 sudo tar zxvf java.tar.gz
16 sudo tar zxvf maven.tar.gz
17 sudo update-alternative --install /user/bin/java java /opt/jdk1.8.0_45/bin/java 100
18 sudo update-alternative --install /user/bin/javac javac /opt/jdk1.8.0_45/bin/java 100
19 sudo update-alternative --install /user/bin/mvn mvn /opt/apache-maven-3.3.9/bin/mvn 100
20
21 # 3) Setting Environment vairables for Maven
22 sudo echo "export M2_HOME=/optapache-maven-3.3.9" >> /etc/profile
23 sudo echo "export M2=$M2_HOME/bin" >> /etc/profile
24 sudo echo "export PATH=$M2:PATH" >> /etc/profile
25 #refresh profile to activate environment vairable
26 source /etc/profile
27
28 #4) Installing Git
29 echo "Installing Git..."
30 sudo apt-get install -y git
31
32 #5) Install Jenkins
33 echo "Installing Jenkins..."
34 cd /home/vagrant/Desktop
35 sudo dpkg -i jenkins_2.1_all.deb
36 sudo apt-get install -y -f
37 sudo apt-get install -y jenkins
38 sudo service jenkins start
39
40 #6) Installing Jira
41 echo "Installing Jira"
42 cd /opt/
43 sudo chmod a+x jira.bin
44 sudo ./jira.bin -q -varfile response.varfile
45
```

The status bar at the bottom of the Notepad++ window shows 'Unix script file', 'length: 1,391 lines: 45', 'Ln: 16 Col: 23 Sel: 0 | 0', 'Windows (CR LF)', 'UTF-8', and 'INS'.

Task 7 – Repository Management



1) Creating New Repository on GitHub

<https://github.com/new> ☆

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner **Repository name**

 **louyiheidar** / **hello-world** 

Great repository names are short and memorable. Need inspiration? How about [effective-octo-guide](#).

Description (optional)

Task 7 – Repository Management

☒ **Public**
Anyone can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.

☐ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** | Add a license: **None** ⓘ

Create repository

2) Creating new repository within terminal (command line)

```
louyiheidar$ echo "# hello-world" >> README.md
louyiheidar$ git init
Initialized empty Git repository in /Users/louyiheidar/.git/
louyiheidar$ git add README.md
louyiheidar$ git commit -m "first commit"
[master (root-commit) 248c9ca] first commit
Committer: Louy Heidar <louyiheidar@Louys-MacBook-Pro.local>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

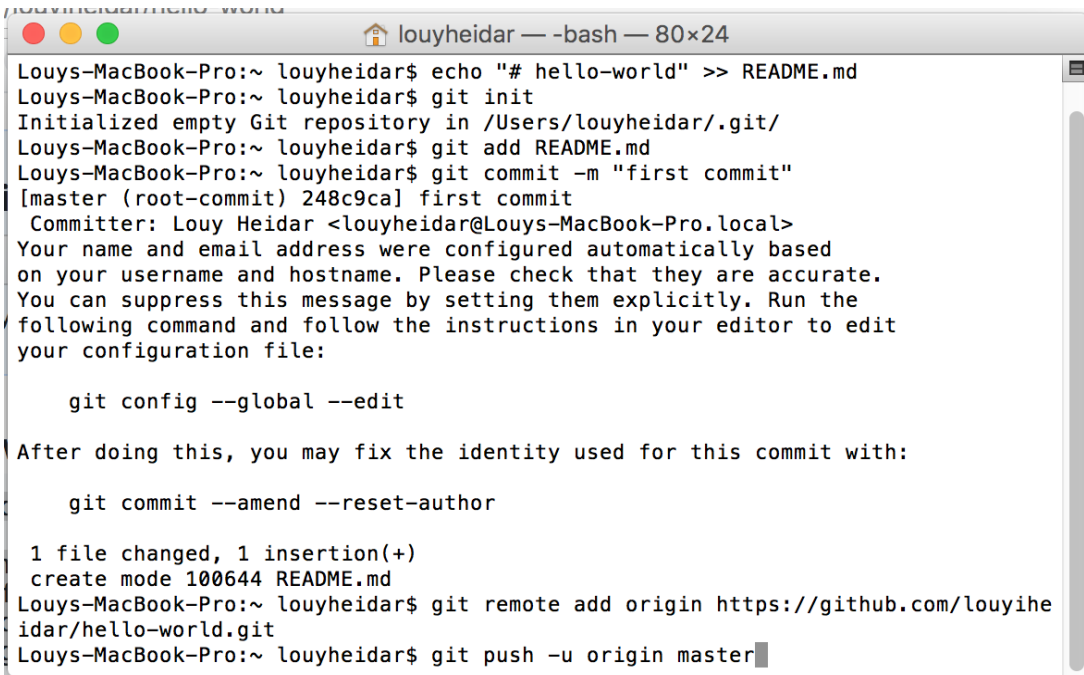
After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

1 file changed, 1 insertion(+)
create mode 100644 README.md
louyiheidar$ git remote add origin https://github.com/louyiheidar/hello-world.git
louyiheidar$ git push -u origin master
```

Lheidar

3) Committing changed and pushing to master branch



```
louyheidar — -bash — 80x24
Louys-MacBook-Pro:~ louyheidar$ echo "# hello-world" >> README.md
Louys-MacBook-Pro:~ louyheidar$ git init
Initialized empty Git repository in /Users/louyheidar/.git/
Louys-MacBook-Pro:~ louyheidar$ git add README.md
Louys-MacBook-Pro:~ louyheidar$ git commit -m "first commit"
[master (root-commit) 248c9ca] first commit
Committer: Louy Heidar <louyheidar@Louys-MacBook-Pro.local>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

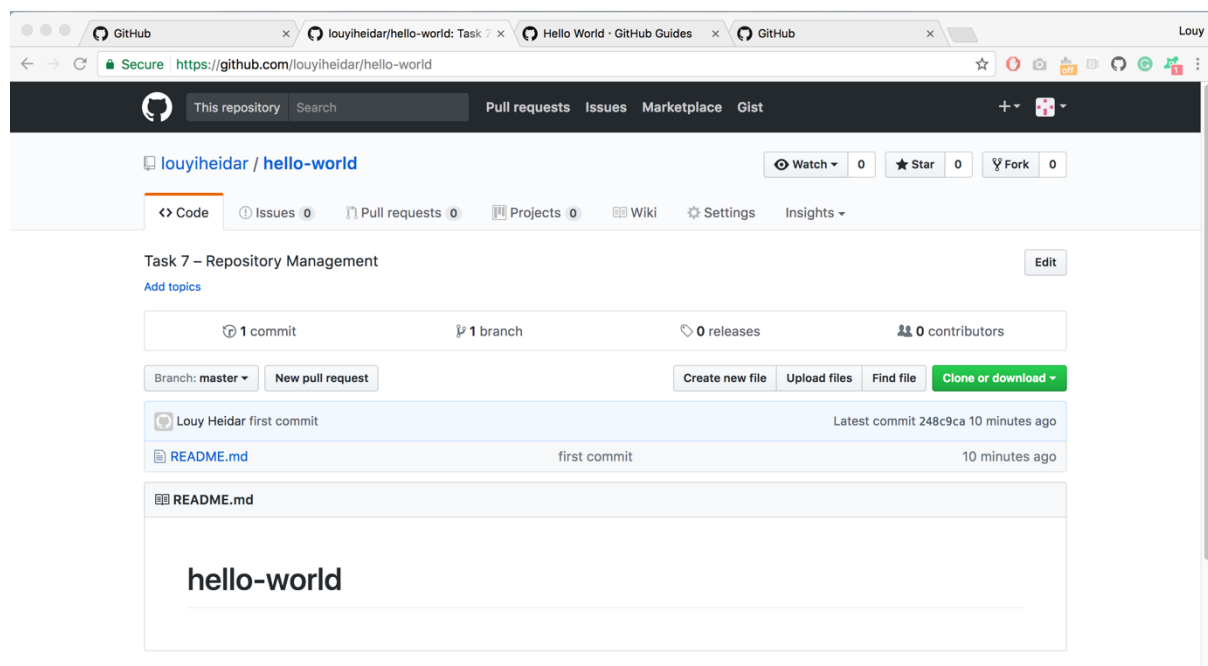
    git config --global --edit

After doing this, you may fix the identity used for this commit with:

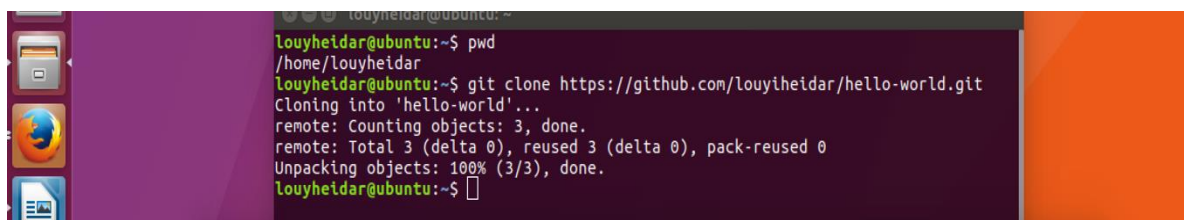
    git commit --amend --reset-author

1 file changed, 1 insertion(+)
create mode 100644 README.md
Louys-MacBook-Pro:~ louyheidar$ git remote add origin https://github.com/louyiheidar/hello-world.git
Louys-MacBook-Pro:~ louyheidar$ git push -u origin master
```

Result of first Commit change:

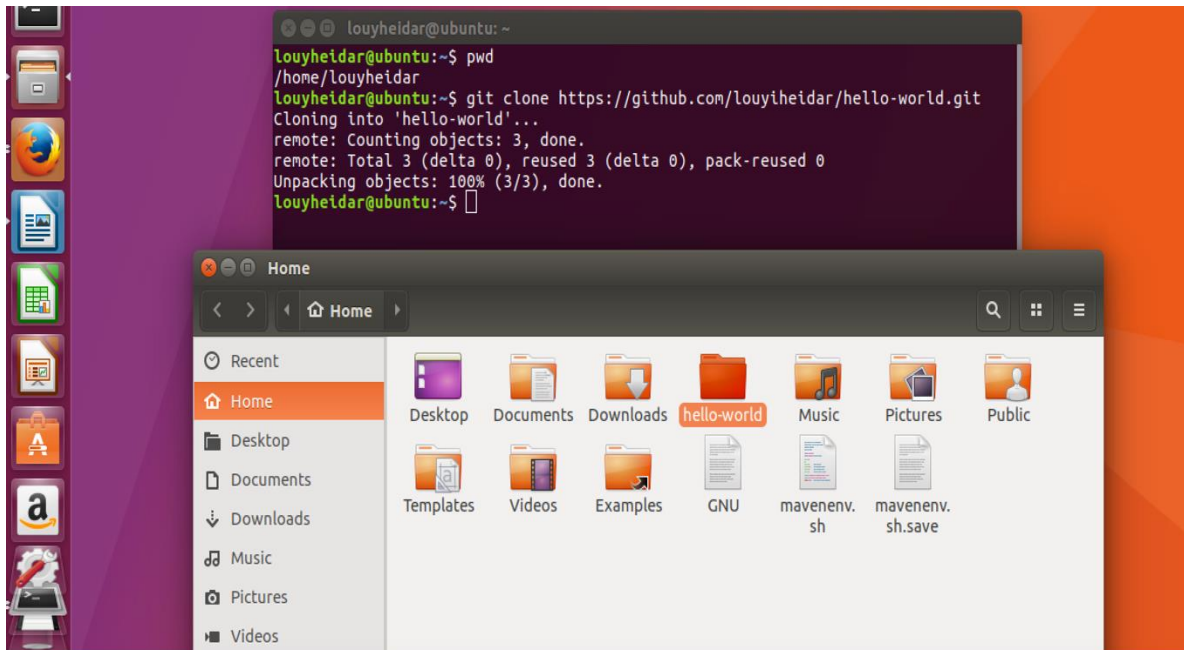


4) Cloning created repository to Ubuntu

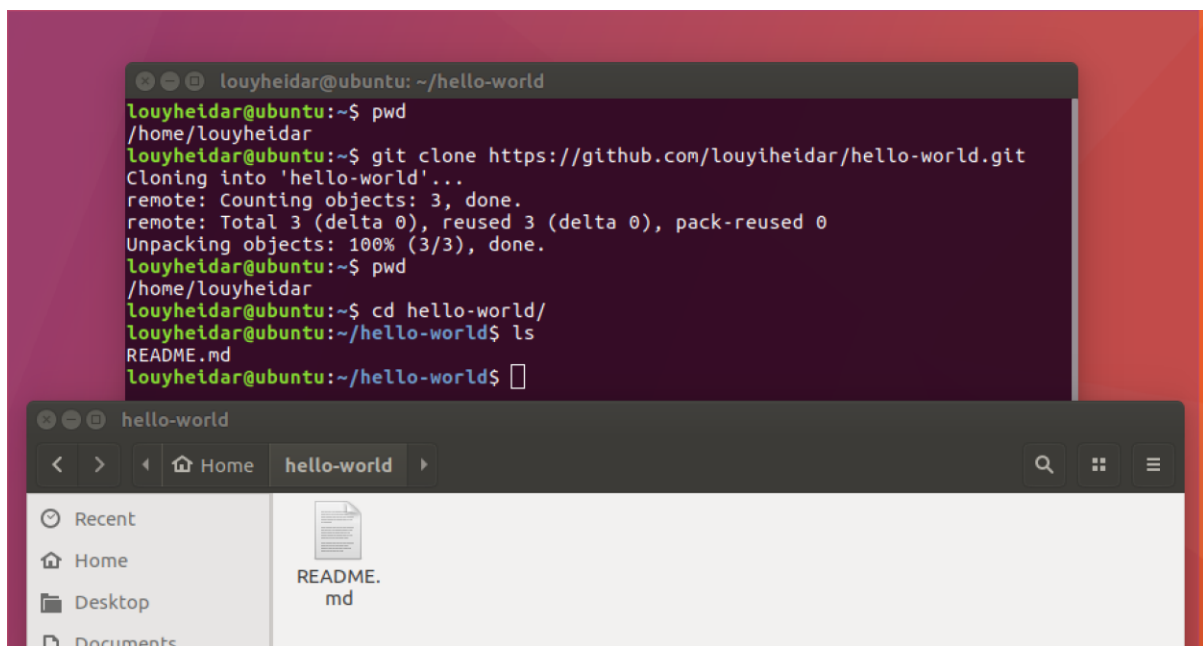


Lheidar

File cloned and display with specified directory
git clone https://github.com/louyiheidar/hello-world.git



Navigation to directory using terminal and verifying content



5) Pushing changes from local directory to GitHub



Local changes pushed to GitHub

Task 7 – Repository Management
Edit

Add topics

2 commits
1 branch
0 releases
0 contributors

Branch: master
New pull request
Create new file
Upload files
Find file
Clone or download

Louy Heidar first commit
Latest commit be56c77 3 minutes ago

README.md
first commit
3 minutes ago

README.md

hello-world 2

File change from Ubuntu VM

- 1)git init
- 2)git add *
- 3)git commit -m "first commit"
- 4)git remote add origin https://github.com/louyiheidar/hello-world.git
- 5)git push -u origin master

Changing and pushing change locally from MacOS

```
//cd to directory
//commands used
```

```
git add README.md
git commit -m "first commit"
git push -u origin master
git pull
```

Task 7 – Repository Management
Edit

Add topics

3 commits
1 branch
0 releases
0 contributors

Branch: master
New pull request
Create new file
Upload files
Find file
Clone or download

Louy Heidar first commit
Latest commit 63f78d6 13 minutes ago

README.md
first commit
13 minutes ago

README.md

hello-world 2

File change from Ubuntu VM

Change made within MacOS

Lheidar

Task 7 – Repository Management Edit

[Add topics](#)

4 commits 1 branch 0 releases 0 contributors

Branch: master New pull request Create new file Upload files Find file Clone or download

Louy Heidar first commit Latest commit 019757a 2 minutes ago

README.md first commit 2 minutes ago

README.md

```
hello-world 2
```

```
File change from Ubuntu VM
```

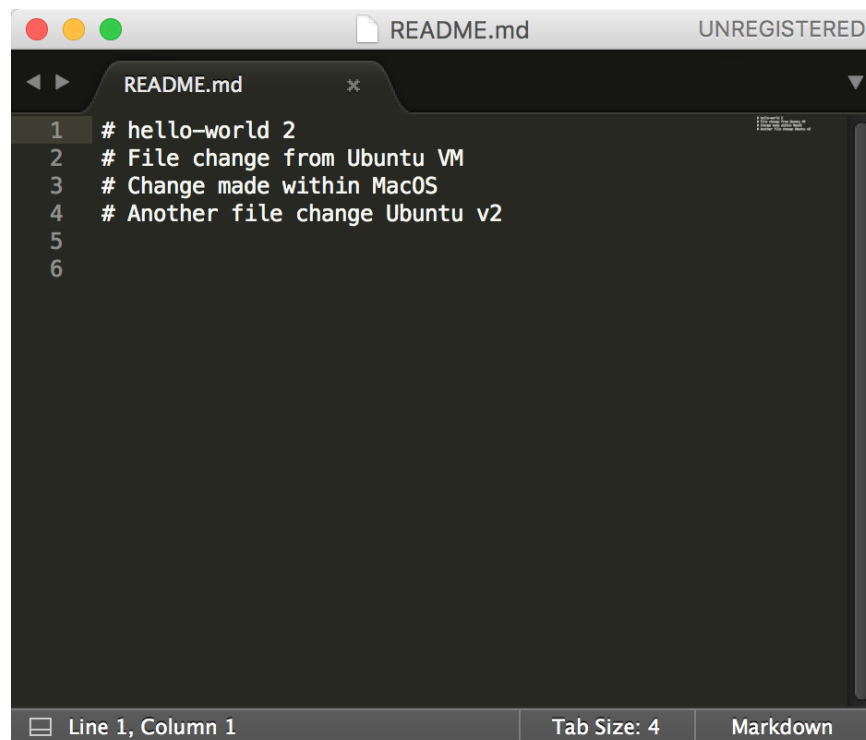
```
Change made within MacOS
```

```
Another file change Ubuntu v2
```

File change have been committed and local and server file have been updated

//cd to directory
//commands used

git pull



The screenshot shows a code editor window with a tab labeled 'README.md'. The editor displays the following content:

```
1 # hello-world 2
2 # File change from Ubuntu VM
3 # Change made within MacOS
4 # Another file change Ubuntu v2
5
6
```

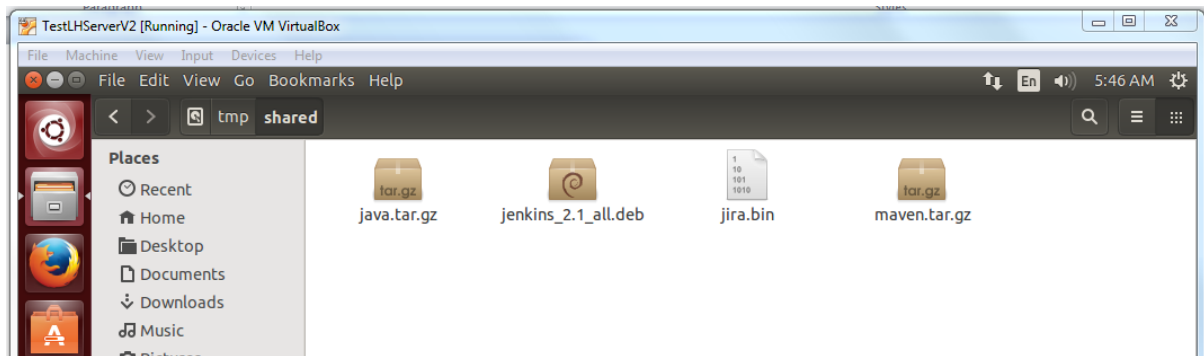
The status bar at the bottom indicates 'Line 1, Column 1', 'Tab Size: 4', and 'Markdown'.

Lheidar

Tooling

Task 2 - Setting up Jira

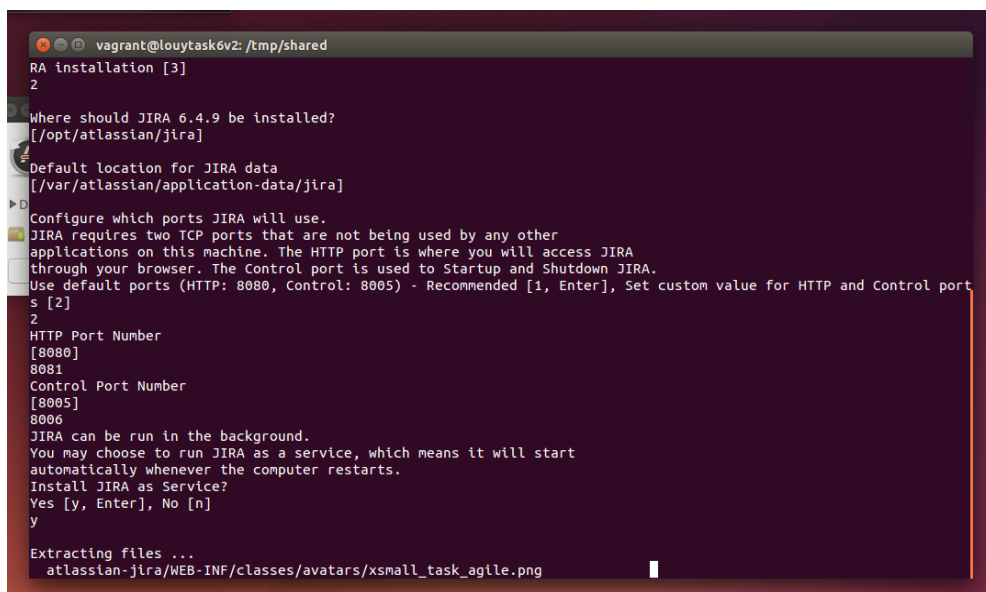
1) Shared folder Mounted via using shell script with all accessible installation files



2) Using custom install to configure set ports

HTTP Port = 8081

Control Port = 8006



1) Jira setup complete and running on the specified ports

