VIT Vidyalankar Institute of Technology
Accredited A+ by NAAC

DEPARTMENT OF INFORMATION TECHNOLOGY

1

Semester	BE Semester VIII – INFT Engineering
Subject	DevOps Lab
Subject Professor In-charge	Prof. Yash Shah
Laboratory	CC02

Student Name			
Roll Number		Division:	Batch:
Grade and Subject Teacher’s Signature			

Experiment Number	1	
Experiment Title	Version Control on software/website using GIT version control tool.	
Resources / Apparatus Required	Hardware: • Intel Core i3/i5/i7 Processor with Intel VT-X support. • 4 GB RAM • 500 GB Hard disk	Software: Operating systems: Windows or Linux Desktop OS for Client machines.

Theory/ Procedure/ Algorithm

Version Control:

Version control is a type of system that allows you to keep track of changes made to your code over time. As such, version control is useful because:

- You can revert back to specific ‘versions’ of your code.
- Collaboration on the same work is possible because specific changes and associated contributors are tracked.

As coding is an integral aspect of data science, it is best practice to use version control to maintain source code and databases. Changes can be recorded in a repository: a data structure that stores files and a record of changes made to those files.

GIT:

Git is one of the most popular version control systems. It is a distributed version control system.

It is designed to handle everything from small to very large projects with speed and efficiency.

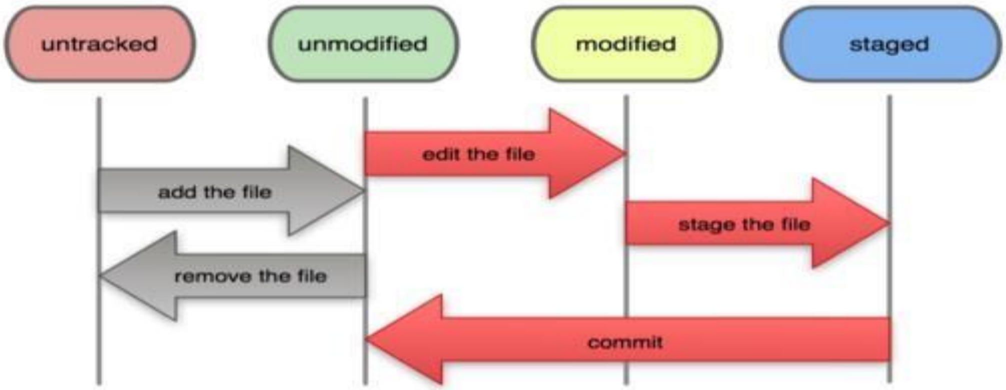


Fig: File Status Lifecycle

- **Initialize Git Repository:** git-init - Create an empty Git repository or reinitialize an existing one.

  Command: git init
- **Check File Status:** To check if any files were modified and not yet committed. Files that are not yet 'tracked' (staged and/or committed) will be noted here. If all files are tracked, then Git will note that there is nothing to commit and the working tree is clean.

  Command: git status
- **Stage File(s) Changes:** You can selectively stage modified files, adding them to the 'staging area' to prepare them to be committed. Modified files that are not added to the staging area will subsequently not be committed.

  Command:
  ```
  #to stage specific modified files
  git add filename
  ```
- **Commit File(s) Changes:** All staged files are then committed, essentially creating a 'screenshot' of those particular files at that particular moment. This effectively records a new change to the repository.

  Command: git commit -m 'describe change(s) made here'

  Each commit must be made with a message, describing the change(s) made. This is done in the present tense and it is best to be more descriptive. This will be helpful when reviewing the logs later on.
- **Using Past Commits**: It may be very useful to check past commits, whether to see what new changes were made since (to potentially identify the source of a new bug) or even to go back to a previous commit.

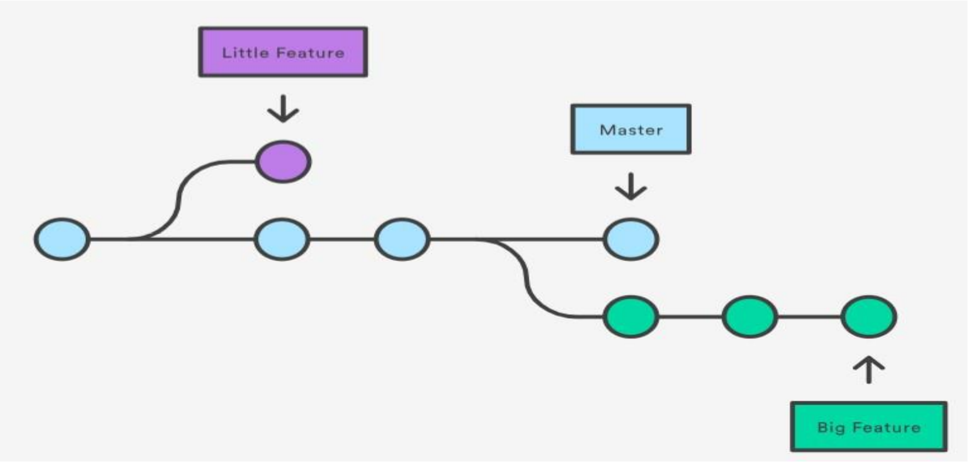
  To display a log of all commits made:

  Command: git log
- **To revert completely to a prior commit:**

  Command: git reset --hard commit-id

  This will essentially go back to the specified commit

Branching:

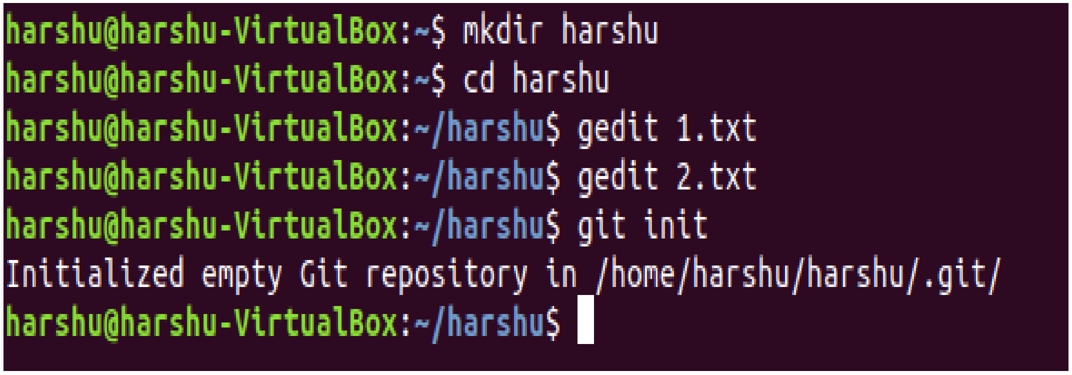


A branch is essentially a 'new' directory, on which you can work on a specific part or feature of a project, before you merge those changes to the main branch that contains all of your source code.

The default branch that you always start with is always called the master branch. The master branch contains the most updated, available source code. Always assume that the master branch is ready to be deployed. All the experimentation and changes, big and small, are made on other branches to be merged in later.

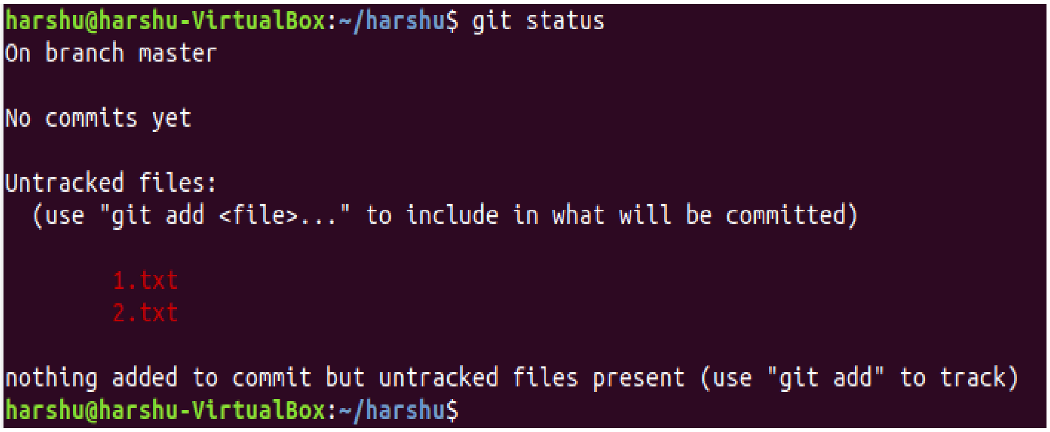
Commands/ Output:

mkdir harshu
cd harshu
gedit 1.txt
gedit 2.txt
git init



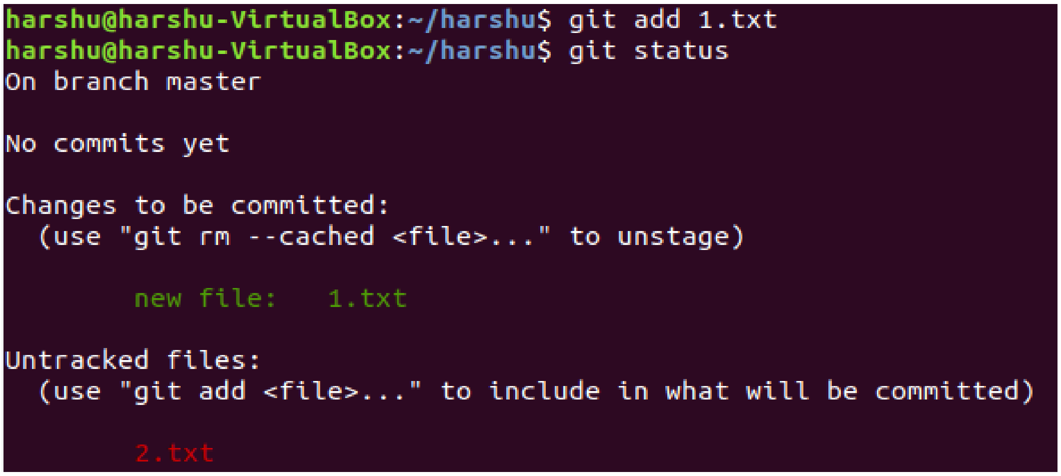
```
harshu@harshu-VirtualBox:~$ mkdir harshu
harshu@harshu-VirtualBox:~$ cd harshu
harshu@harshu-VirtualBox:~/harshu$ gedit 1.txt
harshu@harshu-VirtualBox:~/harshu$ gedit 2.txt
harshu@harshu-VirtualBox:~/harshu$ git init
Initialized empty Git repository in /home/harshu/harshu/.git/
harshu@harshu-VirtualBox:~/harshu$ 
```

git status



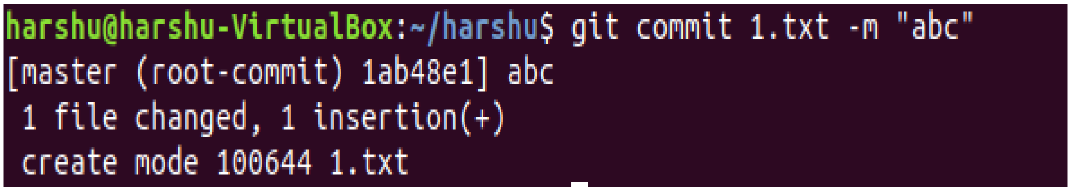
```
harshu@harshu-VirtualBox:~/harshu$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        1.txt
        2.txt

nothing added to commit but untracked files present (use "git add" to track)
harshu@harshu-VirtualBox:~/harshu$
```

git add 1.txt
git status



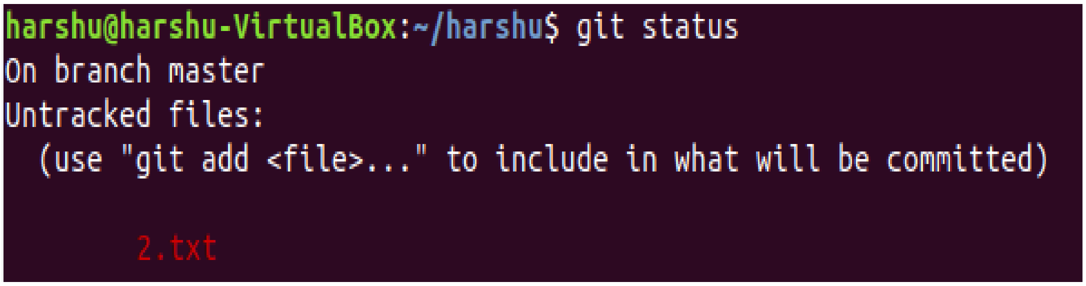
```
harshu@harshu-VirtualBox:~/harshu$ git add 1.txt
harshu@harshu-VirtualBox:~/harshu$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   1.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        2.txt
```

git commit 1.txt -m "abc"



```
harshu@harshu-VirtualBox:~/harshu$ git commit 1.txt -m "abc"
[master (root-commit) 1ab48e1] abc
 1 file changed, 1 insertion(+)
 create mode 100644 1.txt
```

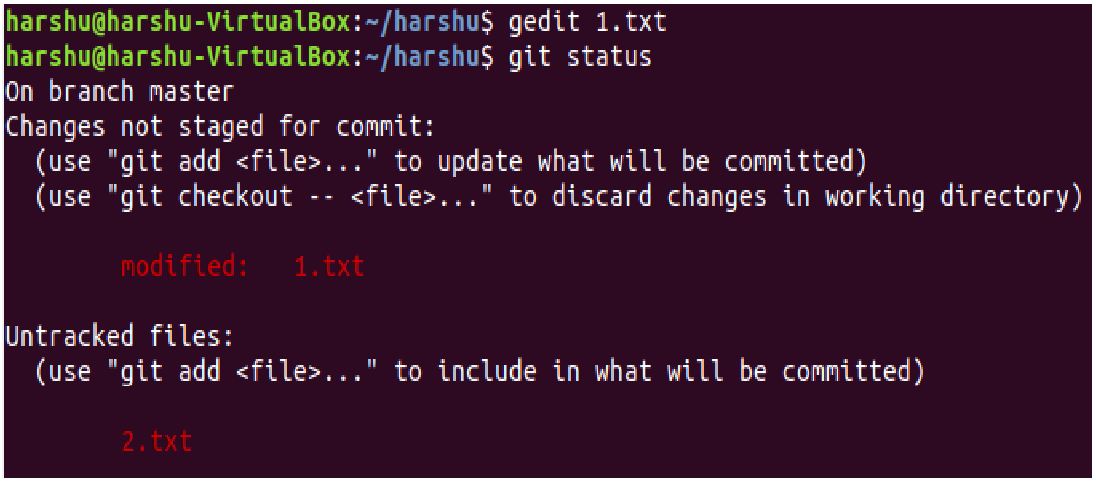
git status



```
harshu@harshu-VirtualBox:~/harshu$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        2.txt
```

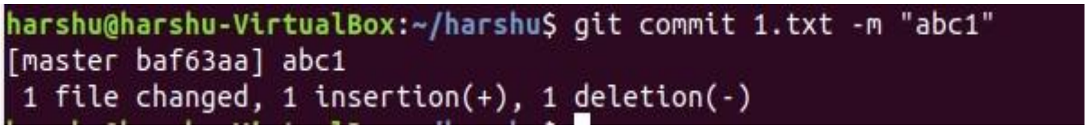
gedit 1.txt

git status



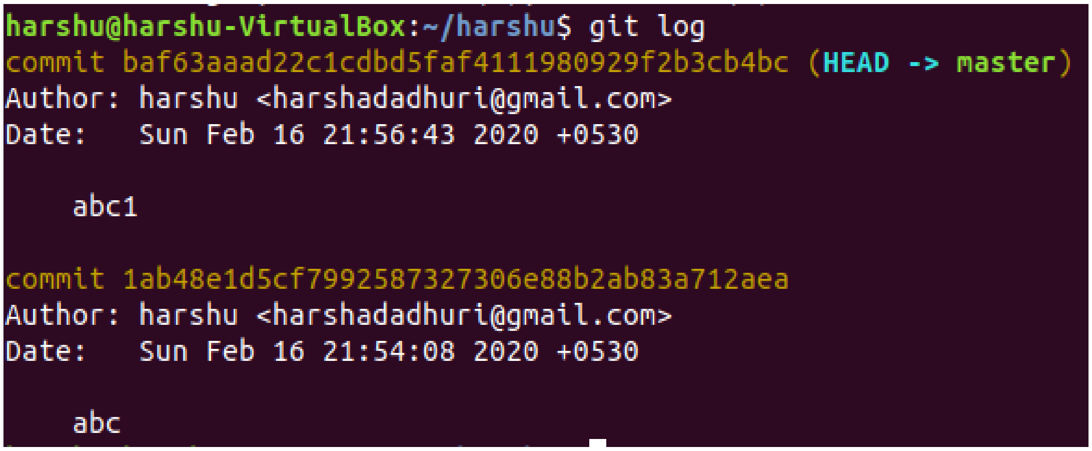
```
harshu@harshu-VirtualBox:~/harshu$ gedit 1.txt
harshu@harshu-VirtualBox:~/harshu$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   1.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        2.txt
```

git commit 1.txt -m "abcde"



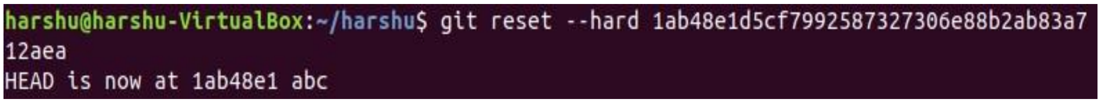
```
harshu@harshu-VirtualBox:~/harshu$ git commit 1.txt -m "abc1"
[master baf63aa] abc1
 1 file changed, 1 insertion(+), 1 deletion(-)
```

git log



```
harshu@harshu-VirtualBox:~/harshu$ git log
commit baf63aaad22c1cdbd5faf4111980929f2b3cb4bc (HEAD -> master)
Author: harshu <harshadadhuri@gmail.com>
Date:   Sun Feb 16 21:56:43 2020 +0530

    abc1

commit 1ab48e1d5cf7992587327306e88b2ab83a712aea
Author: harshu <harshadadhuri@gmail.com>
Date:   Sun Feb 16 21:54:08 2020 +0530

    abc
```

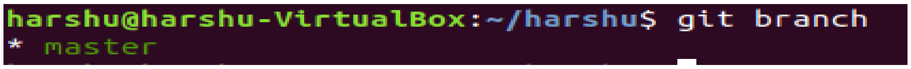
git reset --hard 62588343637090fc849f077bb0c05ea1042de88a



```
harshu@harshu-VirtualBox:~/harshu$ git reset --hard 1ab48e1d5cf7992587327306e88b2ab83a7
12aea
HEAD is now at 1ab48e1 abc
```

gedit 1.txt

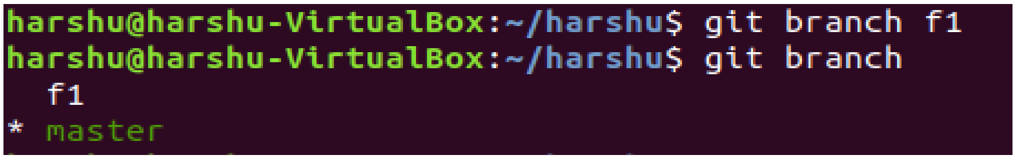
git branch



```
harshu@harshu-VirtualBox:~/harshu$ git branch
* master
```

git branch f1

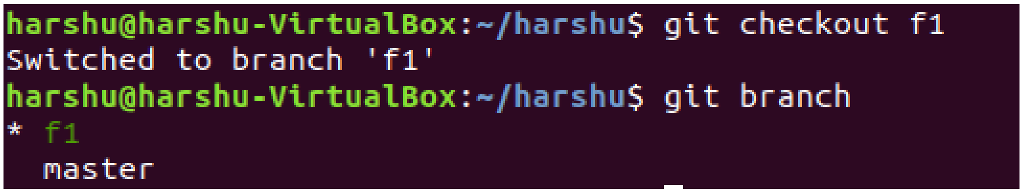
git branch



```
harshu@harshu-VirtualBox:~/harshu$ git branch f1
harshu@harshu-VirtualBox:~/harshu$ git branch
  f1
* master
```

git checkout f1

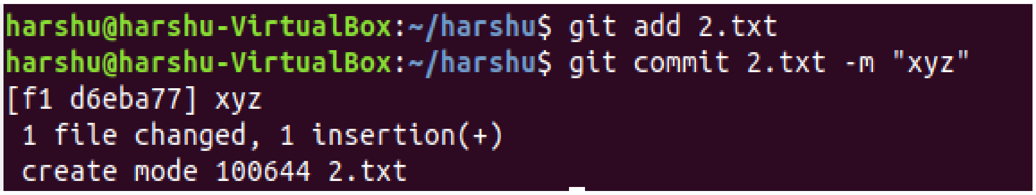
git branch



```
harshu@harshu-VirtualBox:~/harshu$ git checkout f1
Switched to branch 'f1'
harshu@harshu-VirtualBox:~/harshu$ git branch
* f1
  master
```

git add 2.txt

git commit 2.txt -m "xyz"

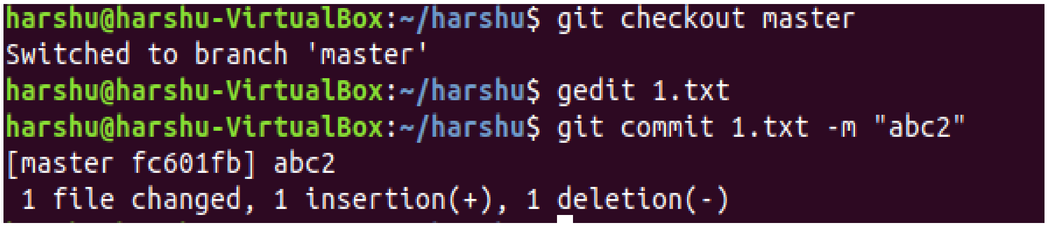


```
harshu@harshu-VirtualBox:~/harshu$ git add 2.txt
harshu@harshu-VirtualBox:~/harshu$ git commit 2.txt -m "xyz"
[f1 d6eba77] xyz
 1 file changed, 1 insertion(+)
 create mode 100644 2.txt
```

git checkout master

gedit 1.txt

git commit 1.txt -m "a"

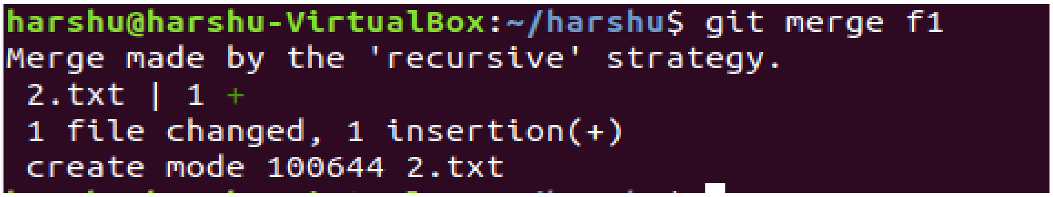


```
harshu@harshu-VirtualBox:~/harshu$ git checkout master
Switched to branch 'master'
harshu@harshu-VirtualBox:~/harshu$ gedit 1.txt
harshu@harshu-VirtualBox:~/harshu$ git commit 1.txt -m "abc2"
[master fc601fb] abc2
 1 file changed, 1 insertion(+), 1 deletion(-)
```

git checkout f1

git checkout master

git merge f1



```
harshu@harshu-VirtualBox:~/harshu$ git merge f1
Merge made by the 'recursive' strategy.
 2.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 2.txt
```

git checkout f1

git commit 2.txt -m "123"

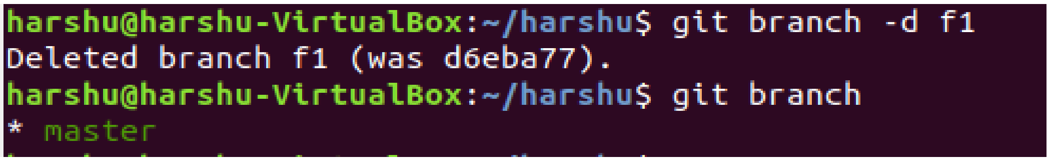
git checkout master

git commit 1.txt -m "1"

git merge f1

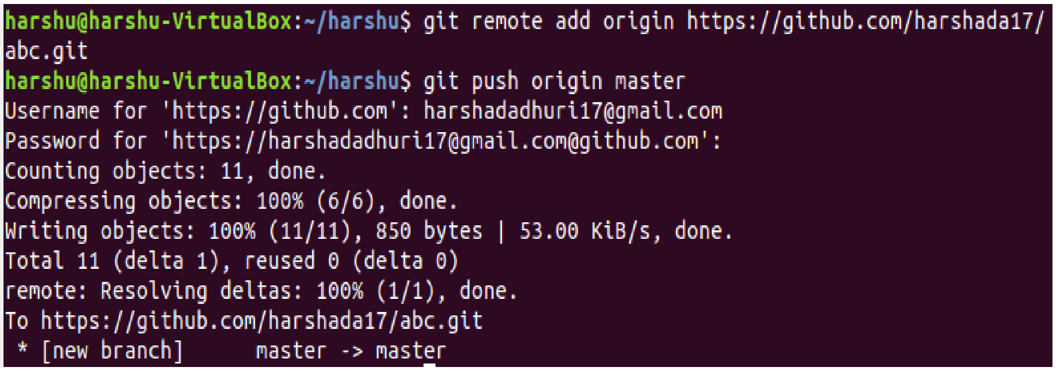
git branch -d f1

git branch



```
harshu@harshu-VirtualBox:~/harshu$ git branch -d f1
Deleted branch f1 (was d6eba77).
harshu@harshu-VirtualBox:~/harshu$ git branch
* master
```

git remote add origin https://github.com/17/abc.git git push
origin master



```
harshu@harshu-VirtualBox:~/harshu$ git remote add origin https://github.com/harshada17/
abc.git
harshu@harshu-VirtualBox:~/harshu$ git push origin master
Username for 'https://github.com': harshadadhuri17@gmail.com
Password for 'https://harshadadhuri17@gmail.com@github.com':
Counting objects: 11, done.
Compressing objects: 100% (6/6), done.
Writing objects: 100% (11/11), 850 bytes | 53.00 KiB/s, done.
Total 11 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/harshada17/abc.git
 * [new branch]      master -> master
```

git branch f2

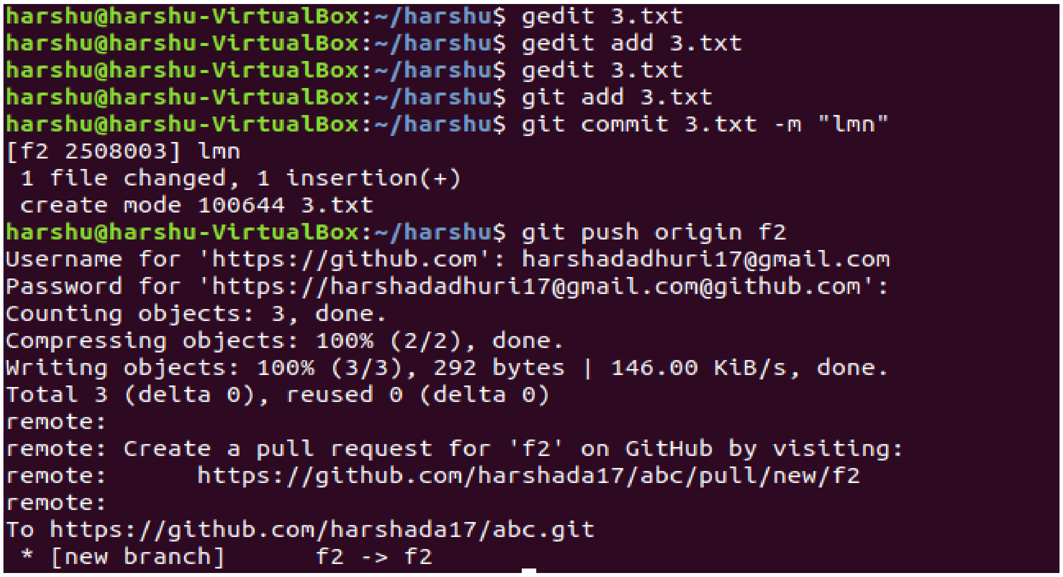
git checkout f2

gedit 3.txt

git add 3.txt

git commit 3.txt -m "9"

git push origin f2

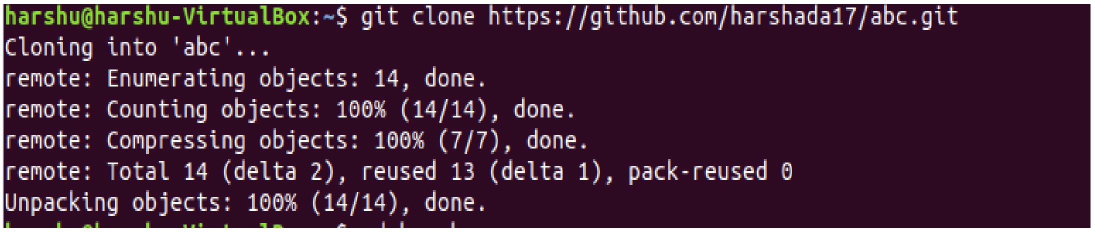


```
harshu@harshu-VirtualBox:~/harshu$ gedit 3.txt
harshu@harshu-VirtualBox:~/harshu$ gedit add 3.txt
harshu@harshu-VirtualBox:~/harshu$ gedit 3.txt
harshu@harshu-VirtualBox:~/harshu$ git add 3.txt
harshu@harshu-VirtualBox:~/harshu$ git commit 3.txt -m "lmn"
[f2 2508003] lmn
 1 file changed, 1 insertion(+)
 create mode 100644 3.txt
harshu@harshu-VirtualBox:~/harshu$ git push origin f2
Username for 'https://github.com': harshadadhuri17@gmail.com
Password for 'https://harshadadhuri17@gmail.com@github.com':
Counting objects: 3, done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 292 bytes | 146.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'f2' on GitHub by visiting:
remote:      https://github.com/harshada17/abc/pull/new/f2
remote:
To https://github.com/harshada17/abc.git
 * [new branch]      f2 -> f2
```

<table>
<tr>
<td></td>
<td>

cd

git clone https://github.com/17/abc.git



```
harshu@harshu-VirtualBox:~$ git clone https://github.com/harshada17/abc.git
Cloning into 'abc'...
remote: Enumerating objects: 14, done.
remote: Counting objects: 100% (14/14), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 14 (delta 2), reused 13 (delta 1), pack-reused 0
Unpacking objects: 100% (14/14), done.
```

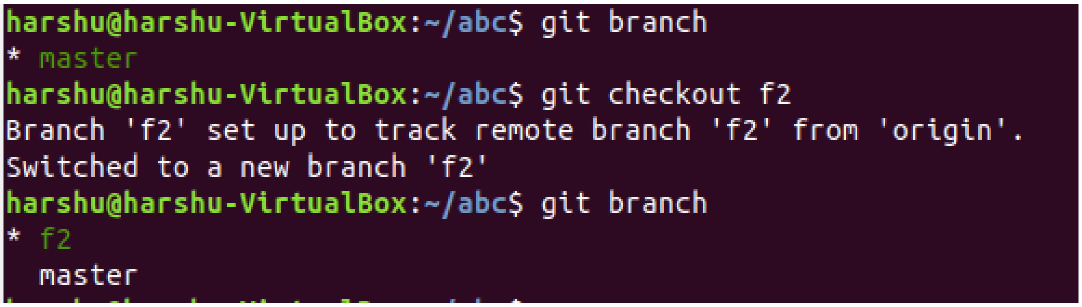
cd harshu

ls

git branch

git checkout f2

git branch



```
harshu@harshu-VirtualBox:~/abc$ git branch
* master
harshu@harshu-VirtualBox:~/abc$ git checkout f2
Branch 'f2' set up to track remote branch 'f2' from 'origin'.
Switched to a new branch 'f2'
harshu@harshu-VirtualBox:~/abc$ git branch
* f2
  master
```

</td>
</tr>
<tr>
<td>Observations/ Conclusion</td>
<td>We come to know about how to use different commands of GIT version-control tool. Version control is “a system that records changes to a file or set of files over time so that we can recall specific versions later.”</td>
</tr>
</table>

VIT | Vidyalankar Institute of Technology
Accredited A+ by NAAC

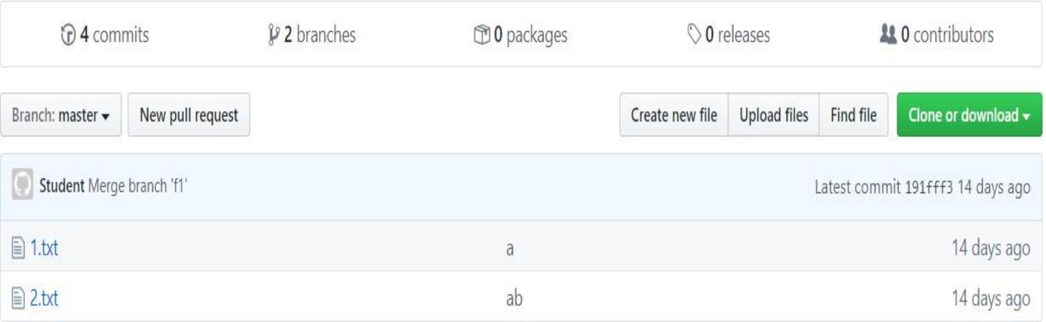
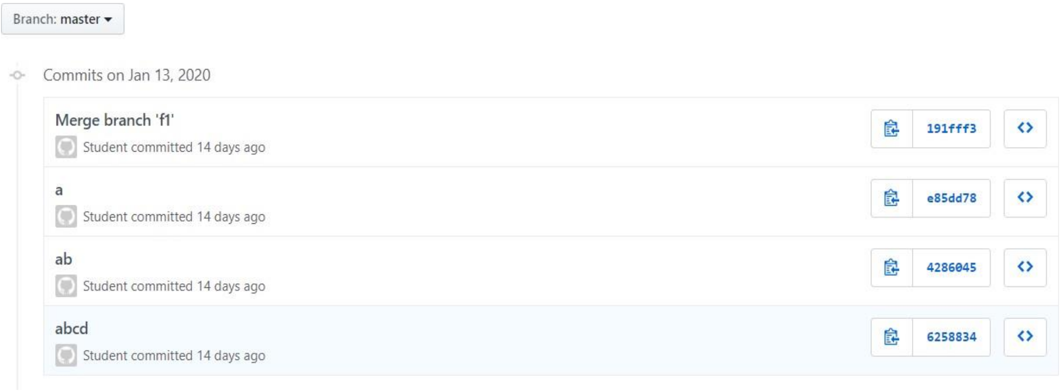
DEPARTMENT OF INFORMATION TECHNOLOGY

2

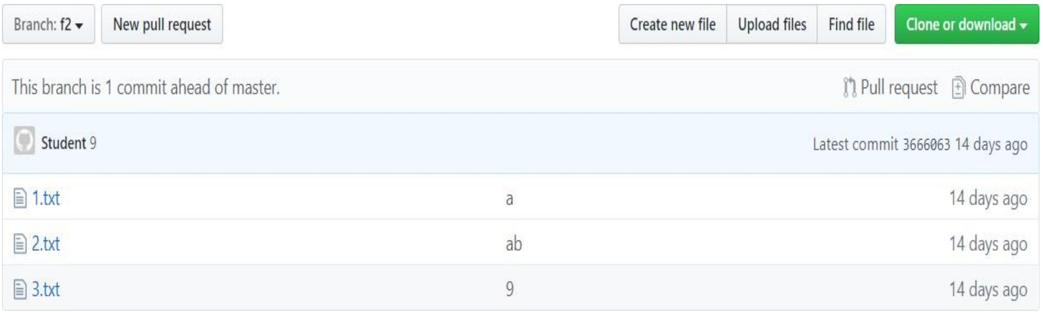
Semester	BE Semester VIII – INFT Engineering
Subject	DevOps Lab
Subject Professor In-charge	Prof. Yash Shah
Laboratory	CC02

Student Name			
Roll Number		Division:	Batch:
Grade and Subject Teacher’s Signature			

Experiment Number	2	
Experiment Title	To perform version control using a GitHub.	
Resources / Apparatus Required	Hardware: • Intel Core i3/i5/i7 Processor with Intel VT-X support. • 4 GB RAM • 500 GB Hard disk	Software: Operating systems: Windows or Linux Desktop OS for Client machines.
Theory/ Procedure/ Algorithm	**GitHub:** • GitHub is a web-based version-control and collaboration platform for software developers. • GitHub allows developers to change, adapt and improve software from its public repositories for free, but it charges for private repositories, offering various paid plans. Each public or private repository contains all of a project's files, as well as each file's revision history. Repositories can have multiple collaborators and can be either public or private. • GitHub facilitates social coding by providing a web interface to the Git code repository and management tools for collaboration. GitHub can be thought of as a serious social networking site for software developers. Members can follow each other, rate each other's work, receive updates for specific projects and communicate publicly or privately. • Three important terms used by developers in GitHub are fork, pull request and merge. A fork, also known as a branch, is simply a repository that has been copied from one member's account to another member's account. Forks and	

	branches allow a developer to make modifications without affecting the original code. If the developer would like to share the modifications, she can send a pull request to the owner of the original repository. If, after reviewing the modifications, the original owner would like to pull the modifications into the repository, she can accept the modifications and merge them with the original repository.
Commands	git remote add origin https://github.com/17/harshu.git git push origin master git branch f2 git checkout f2 gedit 3.txt git add 3.txt git commit 3.txt -m "9" git push origin f2 cd git clone https://github.com/17/harshu.git cd harshu ls git branch git checkout f2 git branch
Output:	Master branch:  4 commits 2 branches 0 packages 0 releases 0 contributors Branch: master ▾ New pull request Create new file Upload files Find file Clone or download ▾ Student Merge branch 'f1' Latest commit 191fff3 14 days ago 1.txt a 14 days ago 2.txt ab 14 days ago Commits on Master branch:  Branch: master ▾ Commits on Jan 13, 2020 Merge branch 'f1' Student committed 14 days ago 191fff3 a Student committed 14 days ago e85dd78 ab Student committed 14 days ago 4286045 abcd Student committed 14 days ago 6258834

f2 branch:

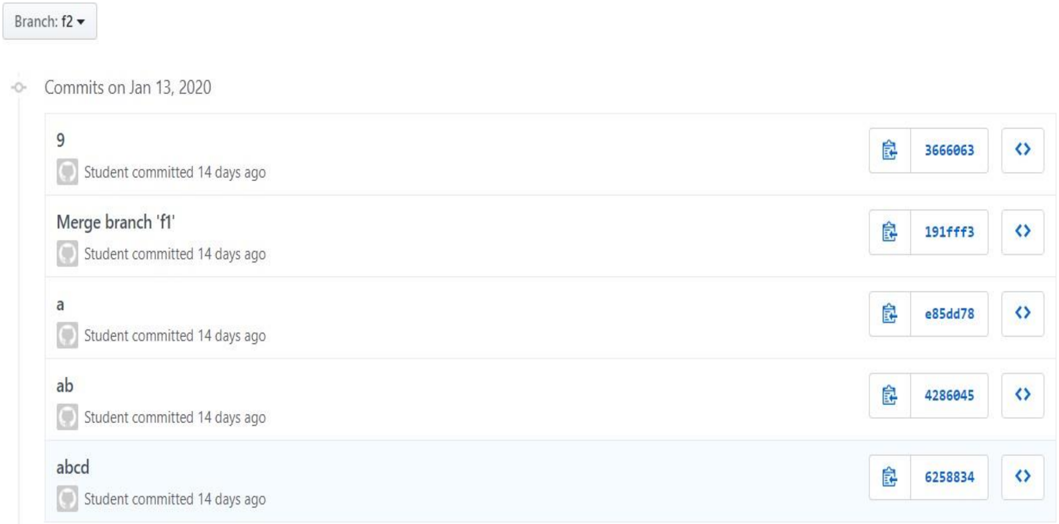


This branch is 1 commit ahead of master. Pull request Compare

Student 9 — Latest commit 3666063 14 days ago

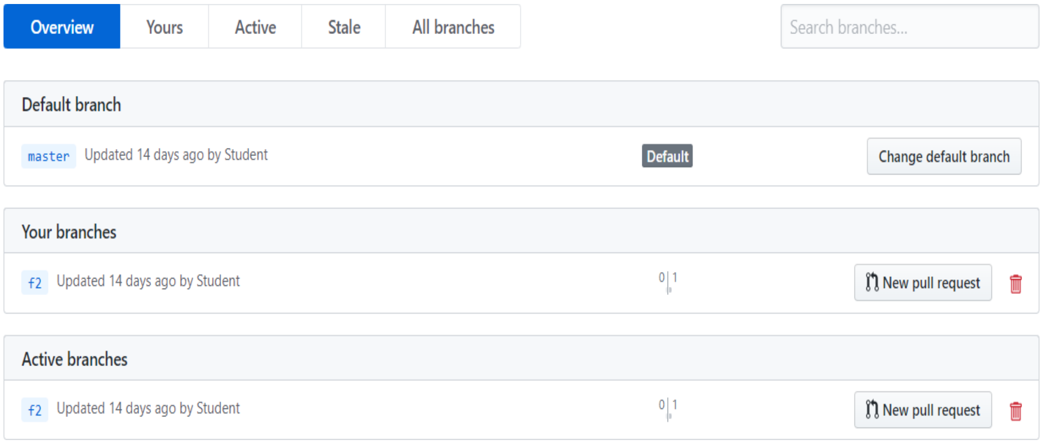
1.txt	a	14 days ago
2.txt	ab	14 days ago
3.txt	9	14 days ago

Commits on f2 branch:



Commits on Jan 13, 2020

9 Student committed 14 days ago	3666063
Merge branch 'f1' Student committed 14 days ago	191fff3
a Student committed 14 days ago	e85dd78
ab Student committed 14 days ago	4286045
abcd Student committed 14 days ago	6258834



Default branch

master Updated 14 days ago by Student — Default — Change default branch

Your branches

f2 Updated 14 days ago by Student — 0 | 1 — New pull request

Active branches

f2 Updated 14 days ago by Student — 0 | 1 — New pull request

Observations/ Conclusion	We come to know about how to use GitHub for version-control. GitHub is a web-based version-control and collaboration platform for software developers.

VIT | Vidyalankar Institute of Technology
Accredited A+ by NAAC

DEPARTMENT OF INFORMATION TECHNOLOGY

3

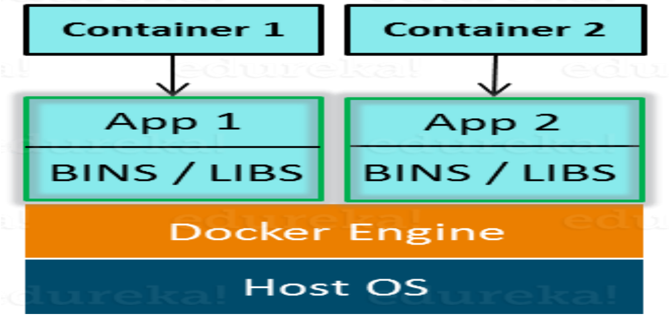
Semester	BE Semester VIII – INFT Engineering
Subject	DevOps Lab
Subject Professor In-charge	Prof. Yash Shah
Laboratory	CC02

Student Name			
Roll Number		Division:	Batch:
Grade and Subject Teacher’s Signature			

Experiment Number	3	
Experiment Title	To install and Configure Docker for creating Containers of operating system.	
Resources / Apparatus Required	Hardware: • Intel Core i3/i5/i7 Processor with Intel VT-X support. • 4 GB RAM • 500 GB Hard disk	Software: Operating systems: Windows or Linux Desktop OS for Client machines.
Theory/ Procedure/ Algorithm	**What is Containerization?** Containerization is the technique of bringing virtualization to the operating system level. While Virtualization brings abstraction to the hardware, Containerization brings abstraction to the operating system. Do note that Containerization is also a type of Virtualization. Containerization is however more efficient because there is no guest OS here and utilizes a host’s operating system, share relevant libraries & resources as and when needed unlike virtual machines. Application specific binaries and libraries of containers run on the host kernel, which makes processing and execution very fast. **Advantages of Containerization:** • Containers on the same OS kernel are lighter and smaller • Better resource utilization compared to VMs • Boot-up process is short and takes few seconds	

Docker

Docker is a containerization platform that packages application and all its dependencies together in the form of Containers to ensure that application works seamlessly in any environment.



As you can see in the diagram on the right, each application will run on a separate container and will have its own set of libraries and dependencies. This also ensures that there is process level isolation, meaning each application is independent of other applications, giving developers surety that they can build applications that will not interfere with one another.

Benefits of Docker

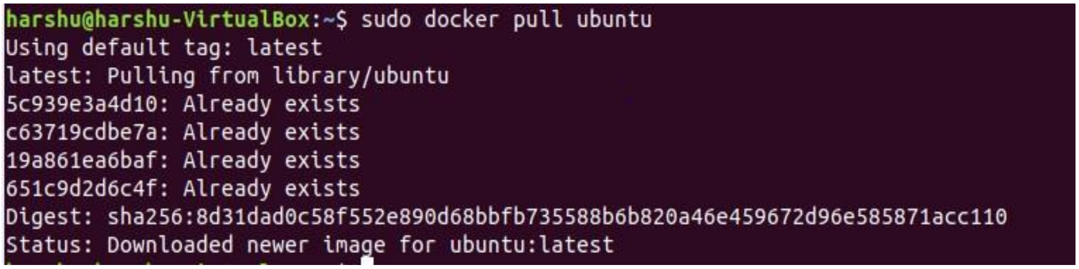
The QA team need not install all the dependent software and applications to test the code and this helps them save lots of time and energy. This also ensures that the working environment is consistent across all the individuals involved in the process, starting from development to deployment. The number of systems can be scaled up easily and the code can be deployed on them effortlessly.

Commands/ Output

sudo apt install docker.io

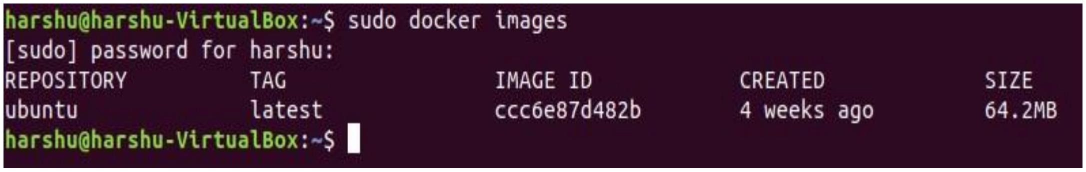
sudo docker --version

sudo docker pull ubuntu



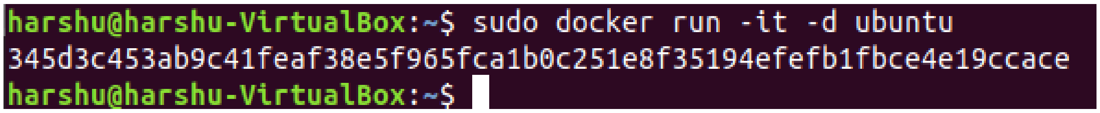
```
harshu@harshu-VirtualBox:~$ sudo docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
5c939e3a4d10: Already exists
c63719cdbe7a: Already exists
19a861ea6baf: Already exists
651c9d2d6c4f: Already exists
Digest: sha256:8d31dad0c58f552e890d68bbfb735588b6b820a46e459672d96e585871acc110
Status: Downloaded newer image for ubuntu:latest
```

sudo docker images



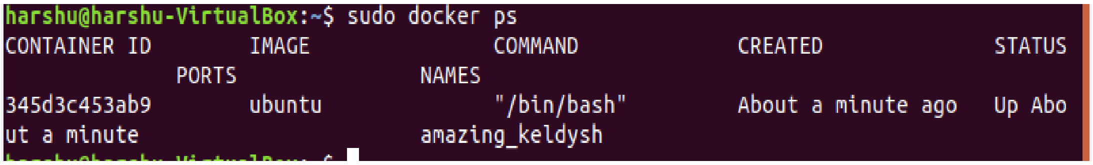
```
harshu@harshu-VirtualBox:~$ sudo docker images
[sudo] password for harshu:
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
ubuntu              latest              ccc6e87d482b        4 weeks ago         64.2MB
harshu@harshu-VirtualBox:~$
```

sudo docker run -it -d ubuntu



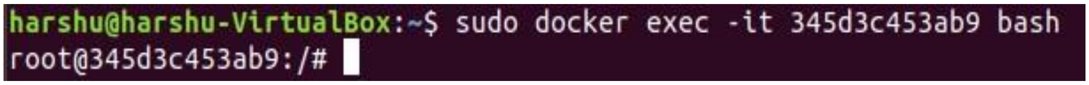
```
harshu@harshu-VirtualBox:~$ sudo docker run -it -d ubuntu
345d3c453ab9c41feaf38e5f965fca1b0c251e8f35194efefb1fbce4e19ccace
harshu@harshu-VirtualBox:~$
```

sudo docker ps



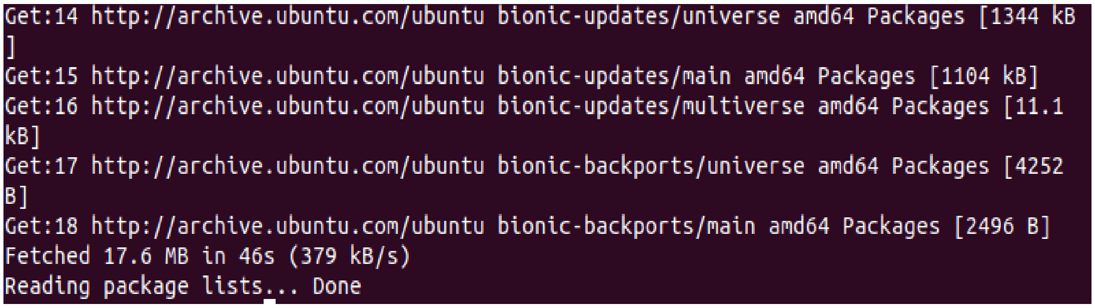
```
harshu@harshu-VirtualBox:~$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS
              PORTS               NAMES
345d3c453ab9        ubuntu              "/bin/bash"         About a minute ago  Up Abo
ut a minute                       amazing_keldysh
harshu@harshu-VirtualBox:~$
```

sudo docker exec -it 345d3c453ab9 bash



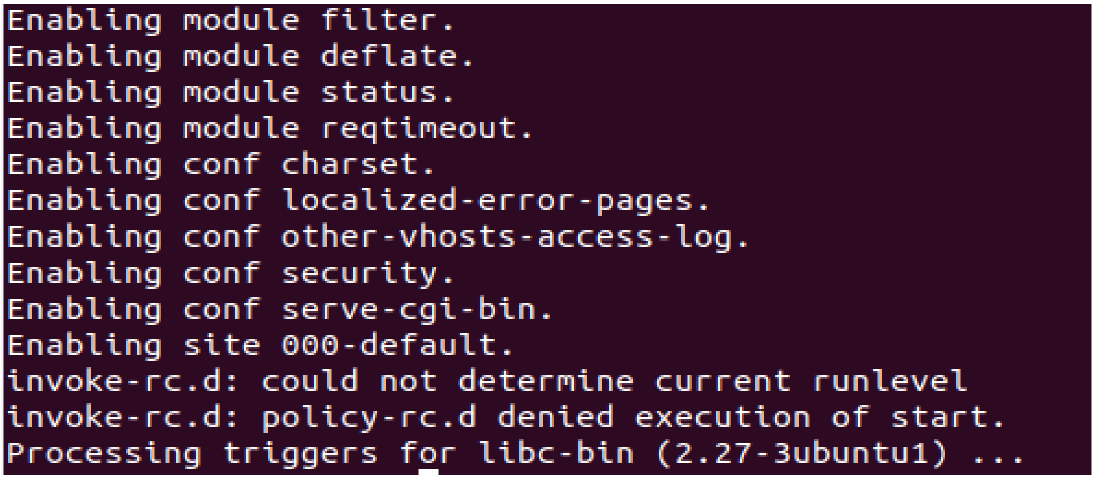
```
harshu@harshu-VirtualBox:~$ sudo docker exec -it 345d3c453ab9 bash
root@345d3c453ab9:/#
```

apt-get update



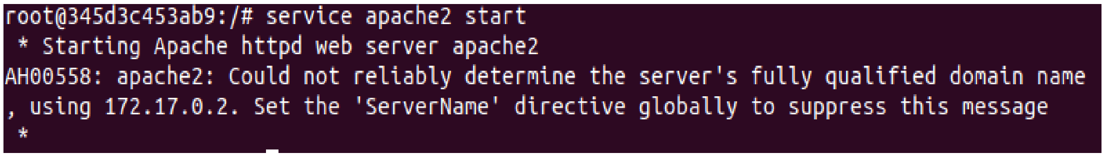
```
Get:14 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [1344 kB
]
Get:15 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [1104 kB]
Get:16 http://archive.ubuntu.com/ubuntu bionic-updates/multiverse amd64 Packages [11.1
kB]
Get:17 http://archive.ubuntu.com/ubuntu bionic-backports/universe amd64 Packages [4252
B]
Get:18 http://archive.ubuntu.com/ubuntu bionic-backports/main amd64 Packages [2496 B]
Fetched 17.6 MB in 46s (379 kB/s)
Reading package lists... Done
```

apt-get install apache2



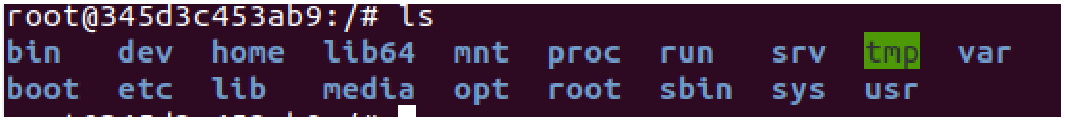
```
Enabling module filter.
Enabling module deflate.
Enabling module status.
Enabling module reqtimeout.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
invoke-rc.d: could not determine current runlevel
invoke-rc.d: policy-rc.d denied execution of start.
Processing triggers for libc-bin (2.27-3ubuntu1) ...
```

service apache2 start



```
root@345d3c453ab9:/# service apache2 start
 * Starting Apache httpd web server apache2
AH00558: apache2: Could not reliably determine the server's fully qualified domain name
, using 172.17.0.2. Set the 'ServerName' directive globally to suppress this message
 *
```

ls

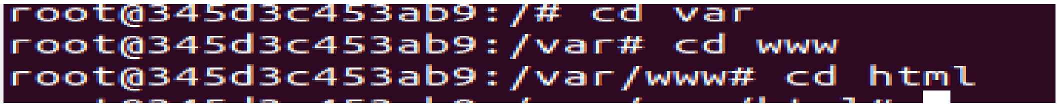


```
root@345d3c453ab9:/# ls
bin   dev  home  lib64  mnt  proc  run   srv  tmp  var
boot  etc  lib   media  opt  root  sbin  sys  usr
```

cd var

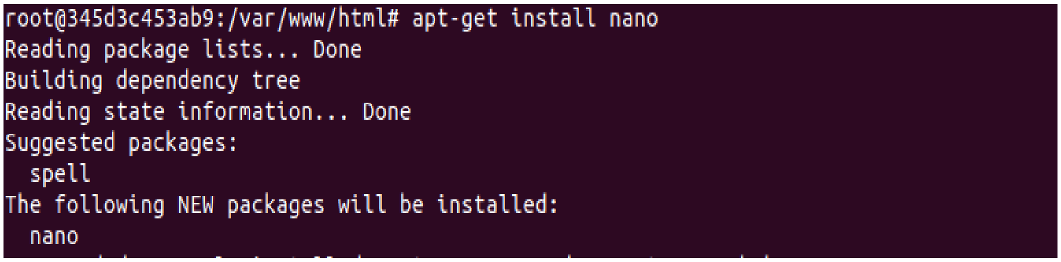
cd www

cd html



```
root@345d3c453ab9:/# cd var
root@345d3c453ab9:/var# cd www
root@345d3c453ab9:/var/www# cd html
```

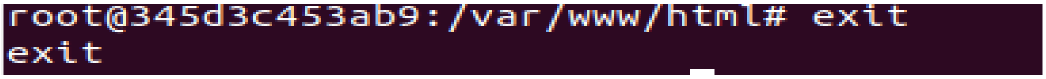
apt-get install nano



```
root@345d3c453ab9:/var/www/html# apt-get install nano
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  spell
The following NEW packages will be installed:
  nano
```

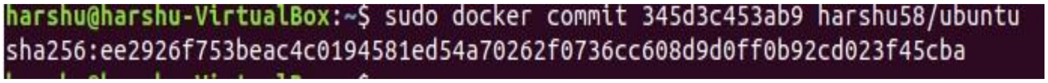
nano harshu.html

exit



```
root@345d3c453ab9:/var/www/html# exit
exit
```

sudo docker commit 345d3c453ab9 harshu58/ubuntu

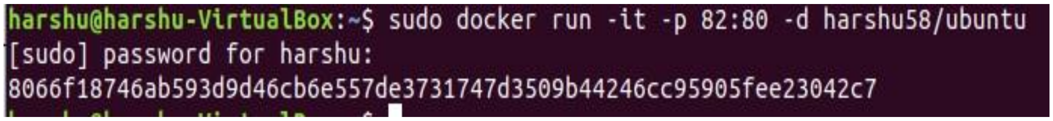


```
harshu@harshu-VirtualBox:~$ sudo docker commit 345d3c453ab9 harshu58/ubuntu
sha256:ee2926f753beac4c0194581ed54a70262f0736cc608d9d0ff0b92cd023f45cba
```

sudo docker ps

sudo docker images

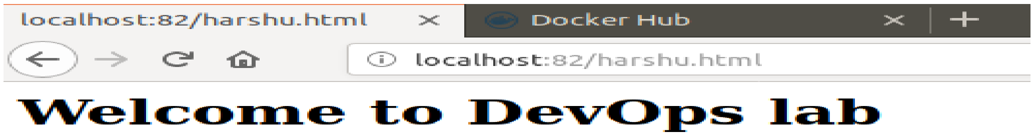
sudo docker run -it -d -p 82:80 harshu58/ubuntu



```
harshu@harshu-VirtualBox:~$ sudo docker run -it -p 82:80 -d harshu58/ubuntu
[sudo] password for harshu:
8066f18746ab593d9d46cb6e557de3731747d3509b44246cc95905fee23042c7
```

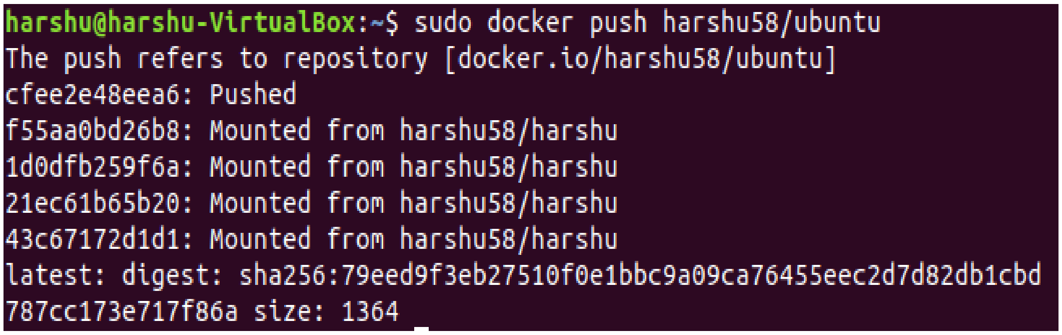
sudo docker exec –it 8066f18746ab bash

service apache2 start

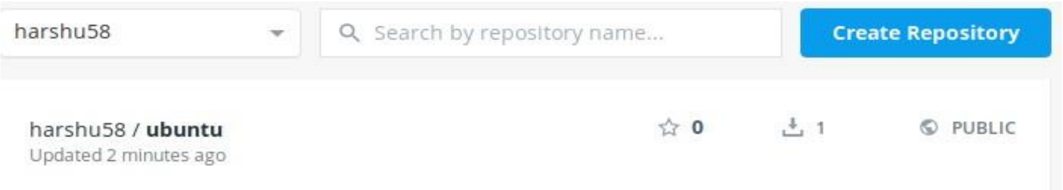


Welcome to DevOps lab

sudo docker login
sudo docker push harshu58/Ubuntu

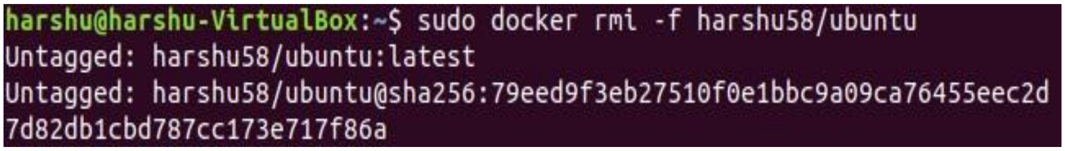


```
harshu@harshu-VirtualBox:~$ sudo docker push harshu58/ubuntu
The push refers to repository [docker.io/harshu58/ubuntu]
cfee2e48eea6: Pushed
f55aa0bd26b8: Mounted from harshu58/harshu
1d0dfb259f6a: Mounted from harshu58/harshu
21ec61b65b20: Mounted from harshu58/harshu
43c67172d1d1: Mounted from harshu58/harshu
latest: digest: sha256:79eed9f3eb27510f0e1bbc9a09ca76455eec2d7d82db1cbd
787cc173e717f86a size: 1364
```



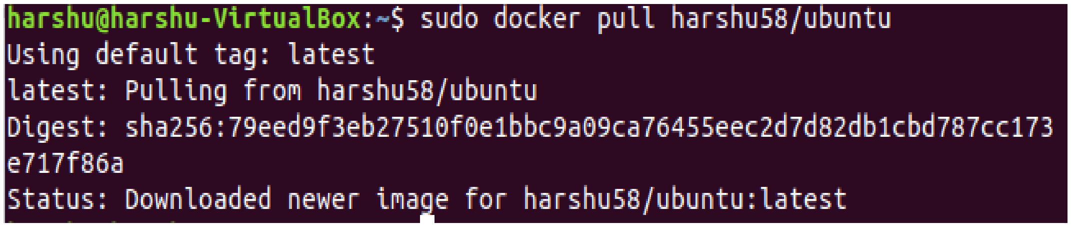
harshu58 / **ubuntu**
Updated 2 minutes ago
☆ 0 1 PUBLIC

sudo docker rmi -f harshu58/Ubuntu



```
harshu@harshu-VirtualBox:~$ sudo docker rmi -f harshu58/ubuntu
Untagged: harshu58/ubuntu:latest
Untagged: harshu58/ubuntu@sha256:79eed9f3eb27510f0e1bbc9a09ca76455eec2d
7d82db1cbd787cc173e717f86a
```

sudo docker images
sudo docker pull harshu58/ubuntu



```
harshu@harshu-VirtualBox:~$ sudo docker pull harshu58/ubuntu
Using default tag: latest
latest: Pulling from harshu58/ubuntu
Digest: sha256:79eed9f3eb27510f0e1bbc9a09ca76455eec2d7d82db1cbd787cc173
e717f86a
Status: Downloaded newer image for harshu58/ubuntu:latest
```

Observations/ Conclusion	We come to know about how to configure Docker for creating containers of OS. Docker is a containerization platform that packages application and all its dependencies together in the form of Containers to ensure that application works seamlessly in any environment.

VIT Vidyalankar Institute of Technology
Accredited A+ by NAAC

DEPARTMENT OF INFORMATION TECHNOLOGY

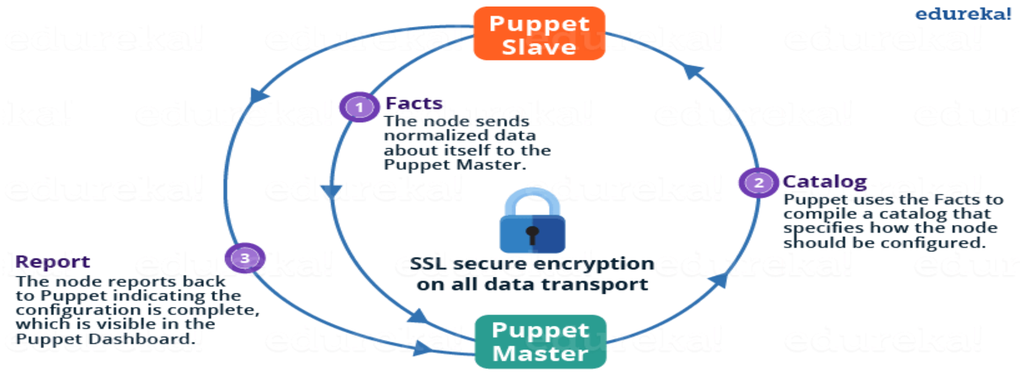
4

Semester	BE Semester VIII – INFT Engineering
Subject	DevOps Lab
Subject Professor In-charge	Prof. Yash Shah
Laboratory	CC02

Student Name			
Roll Number		Division:	Batch:
Grade and Subject Teacher’s Signature			

Experiment Number	4	
Experiment Title	To install and perform Software Configuration Management using Puppet.	
Resources / Apparatus Required	Hardware: • Intel Core i3/i5/i7 Processor with Intel VT-X support. • 4 GB RAM • 500 GB Hard disk	Software: Operating systems: Windows or Linux Desktop OS for Client machines.
Theory	**Puppet:** Puppet is a Configuration Management tool that is used for deploying, configuring and managing servers. It's most commonly used on Linux and Windows to pull the strings on multiple application servers at once. It performs the following functions: • Defining distinct configurations for each and every host, and continuously checking and confirming whether the required configuration is in place and is not altered (if altered Puppet will revert back to the required configuration) on the host. • Dynamic scaling-up and scaling-down of machines. • Providing control over all your configured machines, so a centralized (master-server or repo-based) change gets propagated to all, automatically.	

Architecture of Puppet:

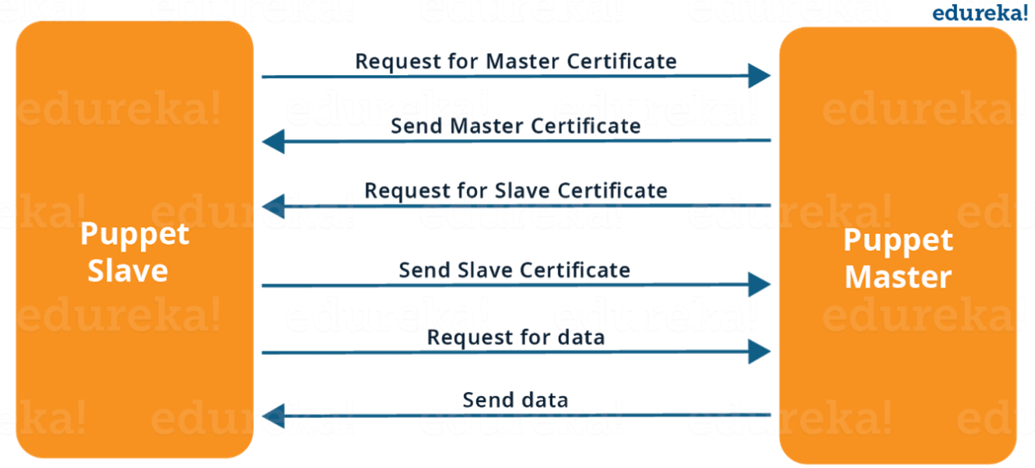

edureka!
Puppet Slave
1
Facts
The node sends normalized data about itself to the Puppet Master.
2
Catalog
Puppet uses the Facts to compile a catalog that specifies how the node should be configured.
3
Report
The node reports back to Puppet indicating the configuration is complete, which is visible in the Puppet Dashboard.
SSL secure encryption on all data transport
Puppet Master

The following functions are performed in the above image:

- The Puppet Agent sends the Facts to the Puppet Master. Facts are basically key/value data pair that represents some aspect of Slave state, such as its IP address, up-time, operating system, or whether it's a virtual machine. I will explain Facts in detail later in the blog.
- Puppet Master uses the facts to compile a Catalog that defines how the Slave should be configured. Catalog is a document that describes the desired state for each resource that Puppet Master manages on a Slave. I will explain catalogs and resources in detail later.

Puppet Master and Slave Communication

Puppet Master and Slave communicates through a secure encrypted channel with the help of SSL. The diagram below depicts the same:



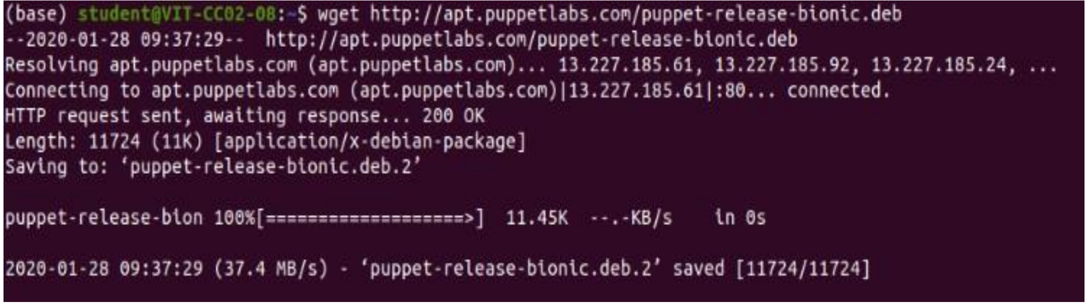
As you can see from the above Image:

- Puppet Slave asks for Puppet Master certificate.
- After receiving Puppet Master certificate, Master requests for Slave certificate.
- Once Master has signed the Slave certificate, Slave requests for configuration/data.
- Finally, Puppet Master will send the configuration to Puppet Slave.

Commands and Output

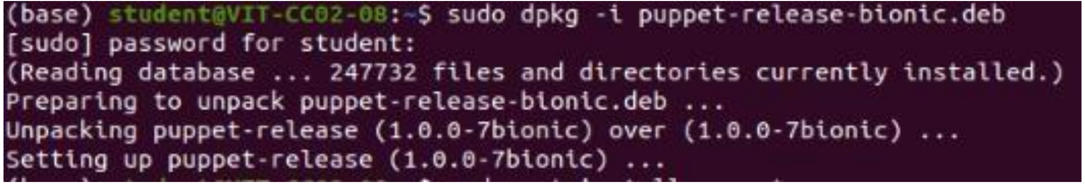
Puppet Agent:

wget http://apt.puppetlabs.com/puppet-release-bionic.deb



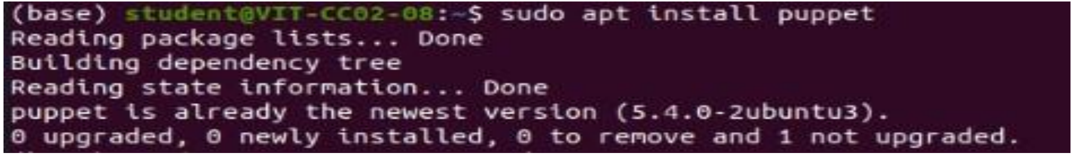
```
(base) student@VIT-CC02-08:~$ wget http://apt.puppetlabs.com/puppet-release-bionic.deb
--2020-01-28 09:37:29--  http://apt.puppetlabs.com/puppet-release-bionic.deb
Resolving apt.puppetlabs.com (apt.puppetlabs.com)... 13.227.185.61, 13.227.185.92, 13.227.185.24, ...
Connecting to apt.puppetlabs.com (apt.puppetlabs.com)|13.227.185.61|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11724 (11K) [application/x-debian-package]
Saving to: ‘puppet-release-bionic.deb.2’

puppet-release-bion 100%[===================>]  11.45K  --.-KB/s    in 0s

2020-01-28 09:37:29 (37.4 MB/s) - ‘puppet-release-bionic.deb.2’ saved [11724/11724]
```

sudo dpkg –i puppet-release-bionic.deb



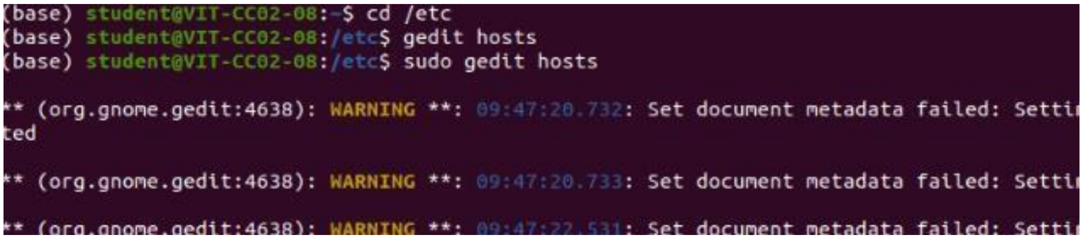
```
(base) student@VIT-CC02-08:~$ sudo dpkg -i puppet-release-bionic.deb
[sudo] password for student:
(Reading database ... 247732 files and directories currently installed.)
Preparing to unpack puppet-release-bionic.deb ...
Unpacking puppet-release (1.0.0-7bionic) over (1.0.0-7bionic) ...
Setting up puppet-release (1.0.0-7bionic) ...
```

sudo apt install puppet



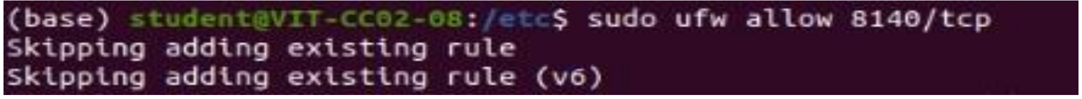
```
(base) student@VIT-CC02-08:~$ sudo apt install puppet
Reading package lists... Done
Building dependency tree
Reading state information... Done
puppet is already the newest version (5.4.0-2ubuntu3).
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
```

Set IP address in hosts file

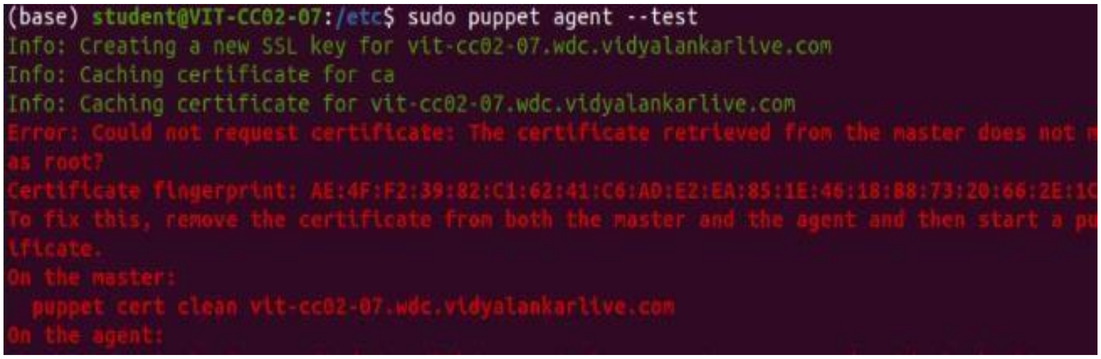


```
(base) student@VIT-CC02-08:~$ cd /etc
(base) student@VIT-CC02-08:/etc$ gedit hosts
(base) student@VIT-CC02-08:/etc$ sudo gedit hosts

** (org.gnome.gedit:4638): WARNING **: 09:47:20.732: Set document metadata failed: Setti
ted

** (org.gnome.gedit:4638): WARNING **: 09:47:20.733: Set document metadata failed: Setti

** (org.gnome.gedit:4638): WARNING **: 09:47:22.531: Set document metadata failed: Setti
```

Open port 8140

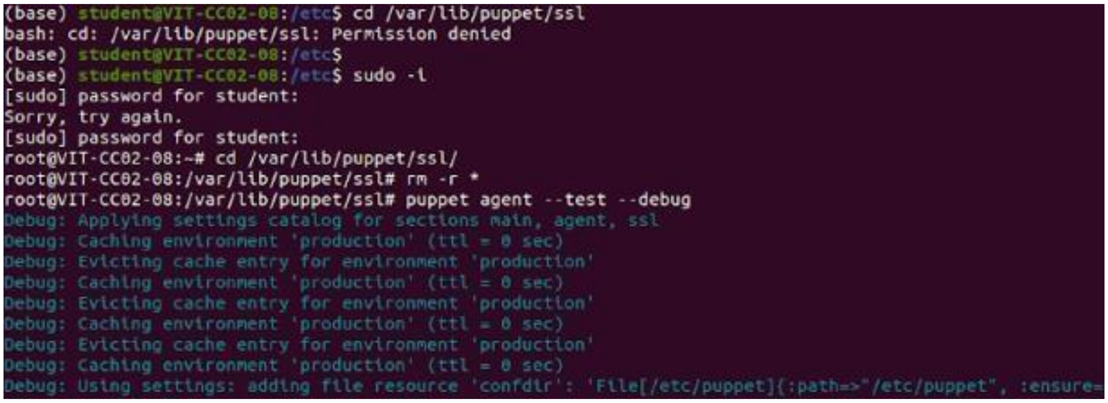


```
(base) student@VIT-CC02-08:/etc$ sudo ufw allow 8140/tcp
Skipping adding existing rule
Skipping adding existing rule (v6)
```

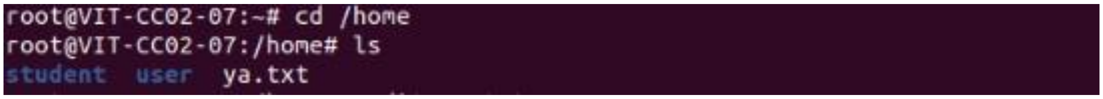


```
(base) student@VIT-CC02-07:/etc$ sudo puppet agent --test
Info: Creating a new SSL key for vit-cc02-07.wdc.vidyalankarlive.com
Info: Caching certificate for ca
Info: Caching certificate for vit-cc02-07.wdc.vidyalankarlive.com
Error: Could not request certificate: The certificate retrieved from the master does not m
as root?
Certificate fingerprint: AE:4F:F2:39:82:C1:62:41:C6:AD:E2:EA:85:1E:46:18:88:73:20:66:2E:1C
To fix this, remove the certificate from both the master and the agent and then start a pu
ificate.
On the master:
  puppet cert clean vit-cc02-07.wdc.vidyalankarlive.com
On the agent:
```

Remove all certificates in ssl folder

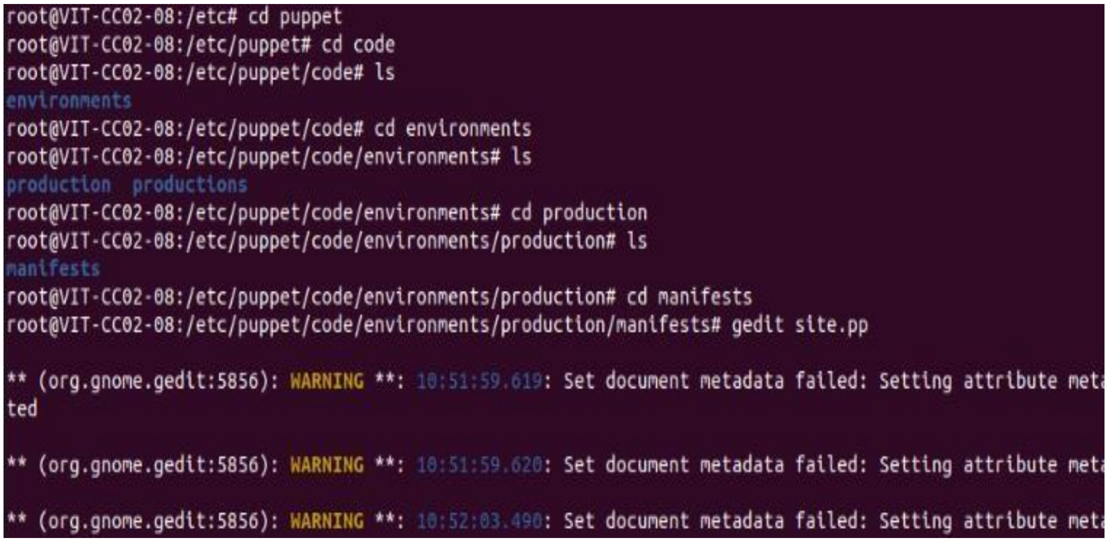


```
(base) student@VIT-CC02-08:/etc$ cd /var/lib/puppet/ssl
bash: cd: /var/lib/puppet/ssl: Permission denied
(base) student@VIT-CC02-08:/etc$
(base) student@VIT-CC02-08:/etc$ sudo -i
[sudo] password for student:
Sorry, try again.
[sudo] password for student:
root@VIT-CC02-08:~# cd /var/lib/puppet/ssl/
root@VIT-CC02-08:/var/lib/puppet/ssl# rm -r *
root@VIT-CC02-08:/var/lib/puppet/ssl# puppet agent --test --debug
Debug: Applying settings catalog for sections main, agent, ssl
Debug: Caching environment 'production' (ttl = 0 sec)
Debug: Evicting cache entry for environment 'production'
Debug: Caching environment 'production' (ttl = 0 sec)
Debug: Evicting cache entry for environment 'production'
Debug: Caching environment 'production' (ttl = 0 sec)
Debug: Evicting cache entry for environment 'production'
Debug: Caching environment 'production' (ttl = 0 sec)
Debug: Using settings: adding file resource 'confdir': 'File[/etc/puppet]{:path=>"/etc/puppet", :ensure=
```



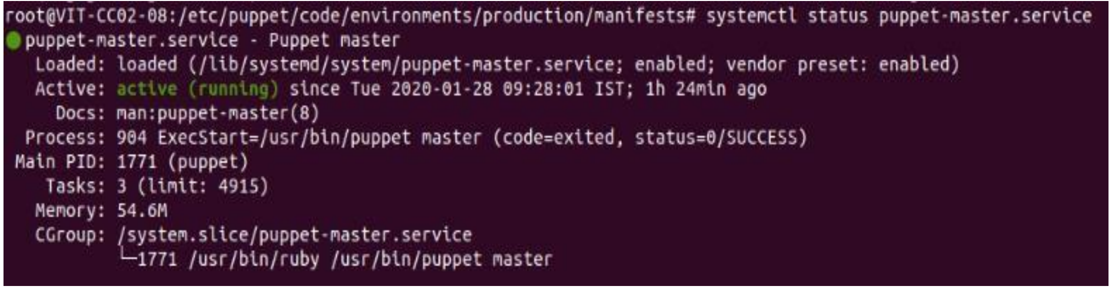
```
root@VIT-CC02-07:~# cd /home
root@VIT-CC02-07:/home# ls
student  user  ya.txt
```

Puppet Master:



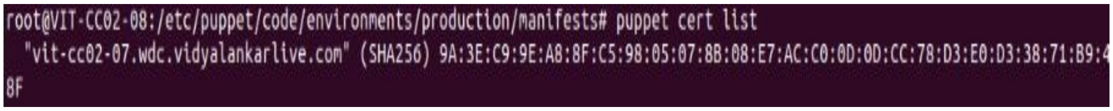
```
root@VIT-CC02-08:/etc# cd puppet
root@VIT-CC02-08:/etc/puppet# cd code
root@VIT-CC02-08:/etc/puppet/code# ls
environments
root@VIT-CC02-08:/etc/puppet/code# cd environments
root@VIT-CC02-08:/etc/puppet/code/environments# ls
production  productions
root@VIT-CC02-08:/etc/puppet/code/environments# cd production
root@VIT-CC02-08:/etc/puppet/code/environments/production# ls
manifests
root@VIT-CC02-08:/etc/puppet/code/environments/production# cd manifests
root@VIT-CC02-08:/etc/puppet/code/environments/production/manifests# gedit site.pp

** (org.gnome.gedit:5856): WARNING **: 10:51:59.619: Set document metadata failed: Setting attribute meta
ted

** (org.gnome.gedit:5856): WARNING **: 10:51:59.620: Set document metadata failed: Setting attribute meta

** (org.gnome.gedit:5856): WARNING **: 10:52:03.490: Set document metadata failed: Setting attribute meta
```

Check the status of puppet master

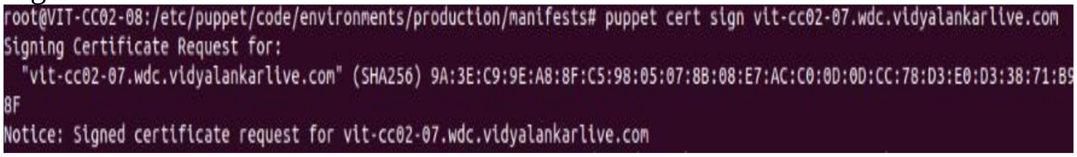


```
root@VIT-CC02-08:/etc/puppet/code/environments/production/manifests# systemctl status puppet-master.service
● puppet-master.service - Puppet master
   Loaded: loaded (/lib/systemd/system/puppet-master.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2020-01-28 09:28:01 IST; 1h 24min ago
     Docs: man:puppet-master(8)
  Process: 904 ExecStart=/usr/bin/puppet master (code=exited, status=0/SUCCESS)
 Main PID: 1771 (puppet)
    Tasks: 3 (limit: 4915)
   Memory: 54.6M
   CGroup: /system.slice/puppet-master.service
           └─1771 /usr/bin/ruby /usr/bin/puppet master
```

Certificates List



```
root@VIT-CC02-08:/etc/puppet/code/environments/production/manifests# puppet cert list
  "vit-cc02-07.wdc.vidyalankarlive.com" (SHA256) 9A:3E:C9:9E:A8:8F:C5:98:05:07:8B:08:E7:AC:C0:0D:0D:CC:78:D3:E0:D3:38:71:B9:4
8F
```

	Sign Certificates  root@VIT-CC02-08:/etc/puppet/code/environments/production/manifests# puppet cert sign vit-cc02-07.wdc.vidyalankarlive.com Signing Certificate Request for: "vit-cc02-07.wdc.vidyalankarlive.com" (SHA256) 9A:3E:C9:9E:A8:8F:C5:98:05:07:8B:08:E7:AC:C0:0D:0D:CC:78:D3:E0:D3:38:71:B9 8F Notice: Signed certificate request for vit-cc02-07.wdc.vidyalankarlive.com
Observations/ Conclusion	We come to know about how to used puppet for software configuration management.

VIT Vidyalankar Institute of Technology
Accredited A+ by NAAC

DEPARTMENT OF INFORMATION TECHNOLOGY

5

Semester	BE Semester VIII – INFT Engineering
Subject	DevOps Lab
Subject Professor In-charge	Prof. Yash Shah
Laboratory	CC02

Student Name		
Roll Number		
Grade and Subject Teacher’s Signature		

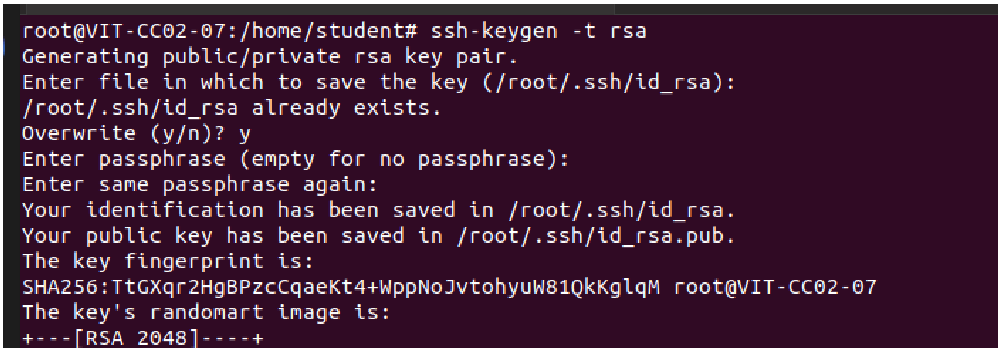
Experiment Number	5	
Experiment Title	To perform Software Configuration Management and provisioning using Ansible.	
Resources / Apparatus Required	Hardware: • Intel Core i3/i5/i7 Processor with Intel VT-X support. • 4 GB RAM • 500 GB Hard Disk	Software: Operating systems: Windows or Linux Desktop OS for Client machines.
Theory	**Ansible:** • Ansible is an automation and orchestration tool popular for its simplicity of installation, ease of use in what concerns the connectivity to clients, its lack of agent for ansible clients and the multitude of skills. • Ansible functions by connecting via SSH to the clients, so it doesn't need a special agent on the client-side, and by pushing modules to the clients. The modules are then executed locally, on the client-side, and the output is pushed back to the Ansible server. • Since it uses SSH, it can very easily connect to clients using SSH-Keys, simplifying though the whole process. Client details, like hostnames or IP addresses and SSH ports, are stored in files called inventory files. Once you have created an inventory file and populated it, ansible can use it. **Important terms used in Ansible** • Ansible server: The machine where Ansible is installed and from which all tasks and playbooks will be ran • Module: Basically, a module is a command or set of similar commands meant to be executed on the client-side • Task: A task is a section that consists of a single procedure to be completed • Role: A way of organizing tasks and related files to be later called in a playbook • Fact: Information fetched from the client system from the global variables with the gather-facts operation	

- Inventory: File containing data about the ansible client servers. Defined in later examples as hosts file
- Play: Execution of a playbook
- Handler: Task which is called only if a notifier is present
- Notifier: Section attributed to a task which calls a handler if the output is changed
- Tag: Name set to a task which can be used later on to issue just that specific task or group of tasks.

Commands and Output

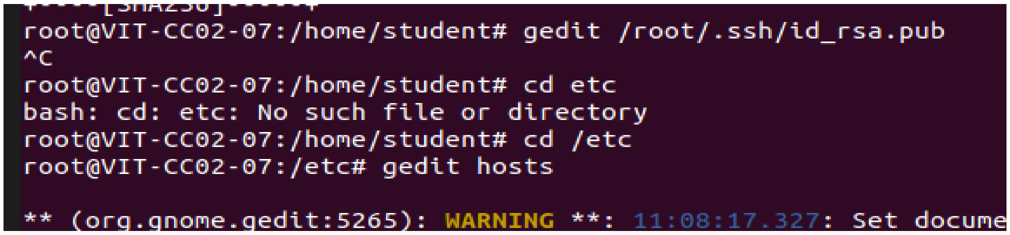
On Controller machine

ssh-keygen -t rsa



```
root@VIT-CC02-07:/home/student# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
/root/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:TtGXqr2HgBPzcCqaeKt4+WppNoJvtohyuW81QkKglqM root@VIT-CC02-07
The key's randomart image is:
+---[RSA 2048]----+
```

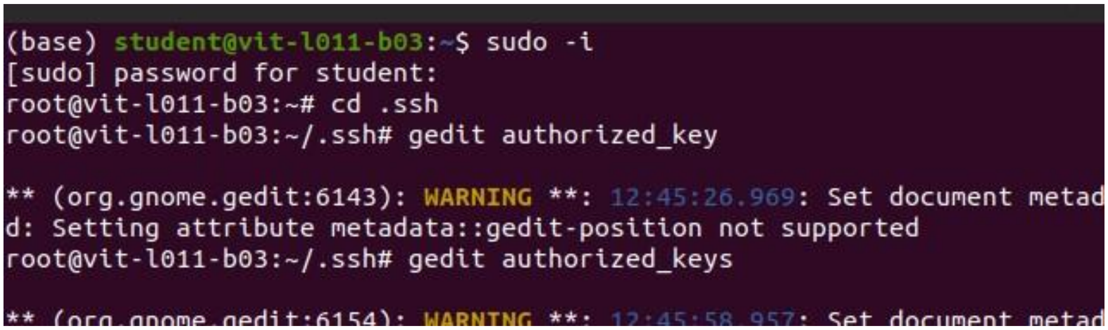
gedit /root/.ssh/id_rsa.pub



```
+----[SHA256]-----+
root@VIT-CC02-07:/home/student# gedit /root/.ssh/id_rsa.pub
^C
root@VIT-CC02-07:/home/student# cd etc
bash: cd: etc: No such file or directory
root@VIT-CC02-07:/home/student# cd /etc
root@VIT-CC02-07:/etc# gedit hosts

** (org.gnome.gedit:5265): WARNING **: 11:08:17.327: Set docume
```

On Controlled machine

sudo -i
mkdir .ssh
cd .ssh
gedit authorized_keys

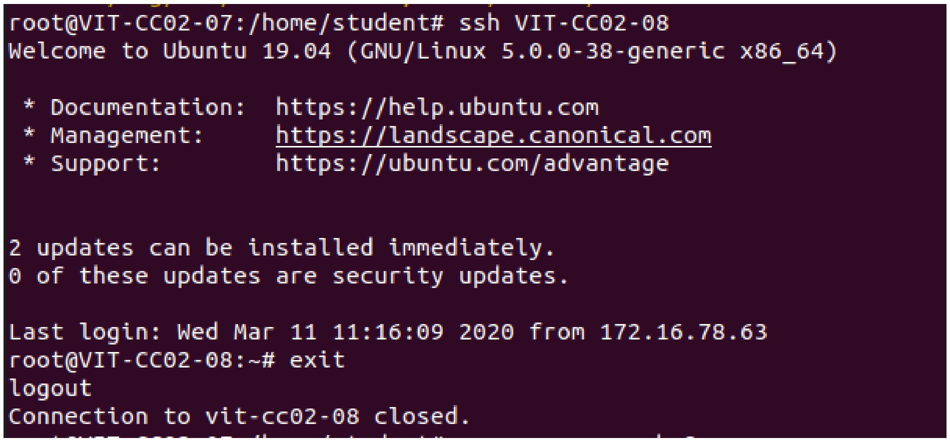


```
(base) student@vit-l011-b03:~$ sudo -i
[sudo] password for student:
root@vit-l011-b03:~# cd .ssh
root@vit-l011-b03:~/.ssh# gedit authorized_key

** (org.gnome.gedit:6143): WARNING **: 12:45:26.969: Set document metad
d: Setting attribute metadata::gedit-position not supported
root@vit-l011-b03:~/.ssh# gedit authorized_keys

** (org.gnome.gedit:6154): WARNING **: 12:45:58.957: Set document metad
```

```
** (org.gnome.gedit:6154): WARNING **: 12:46:01.017: Set docume
d: Setting attribute metadata::gedit-position not supported
root@vit-l011-b03:~/.ssh# ifconfig
```

On Controller machine

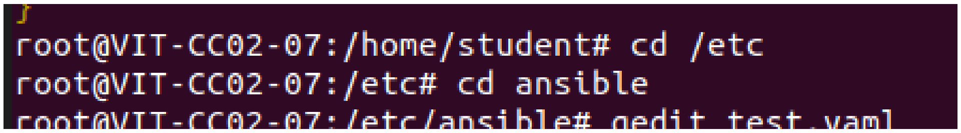
gedit hosts (enter ip address and host name of controlled machine)
ssh VIT-CC02-08



```
root@VIT-CC02-07:/home/student# ssh VIT-CC02-08
Welcome to Ubuntu 19.04 (GNU/Linux 5.0.0-38-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage


2 updates can be installed immediately.
0 of these updates are security updates.

Last login: Wed Mar 11 11:16:09 2020 from 172.16.78.63
root@VIT-CC02-08:~# exit
logout
Connection to vit-cc02-08 closed.
```

Ansible
On Controller machine

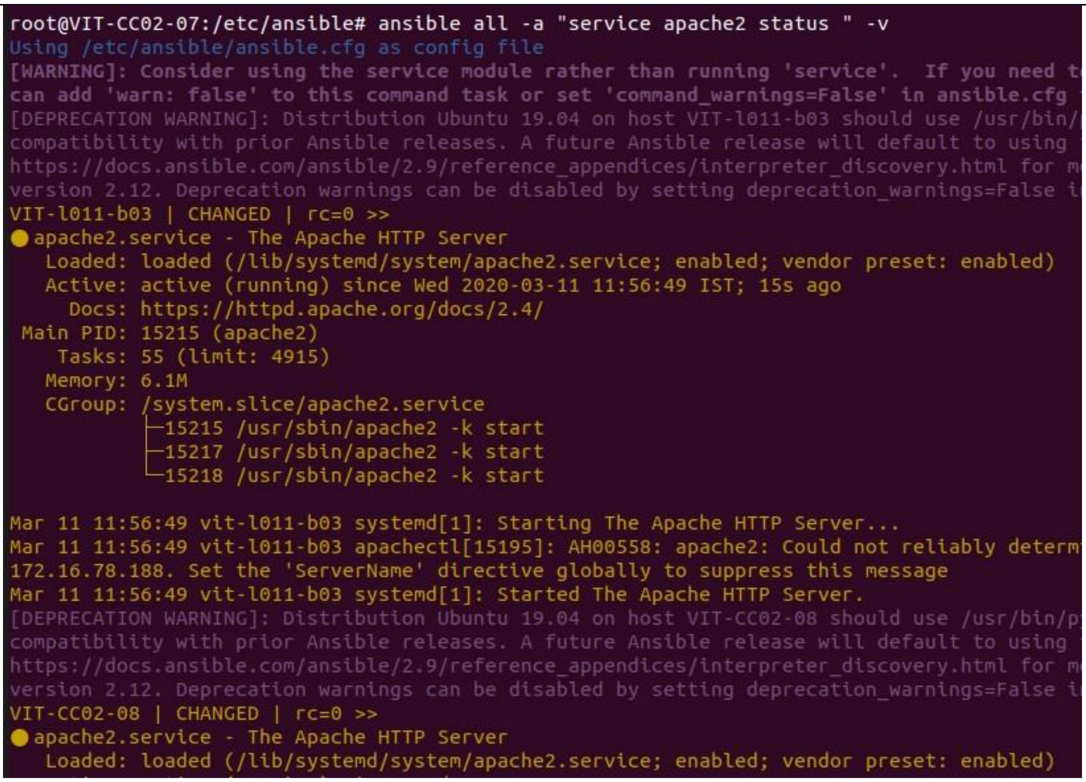
apt-get update
apt-get install ansible
cd /etc
cd ansible



```
root@VIT-CC02-07:/home/student# cd /etc
root@VIT-CC02-07:/etc# cd ansible
root@VIT-CC02-07:/etc/ansible# gedit test.yaml
```

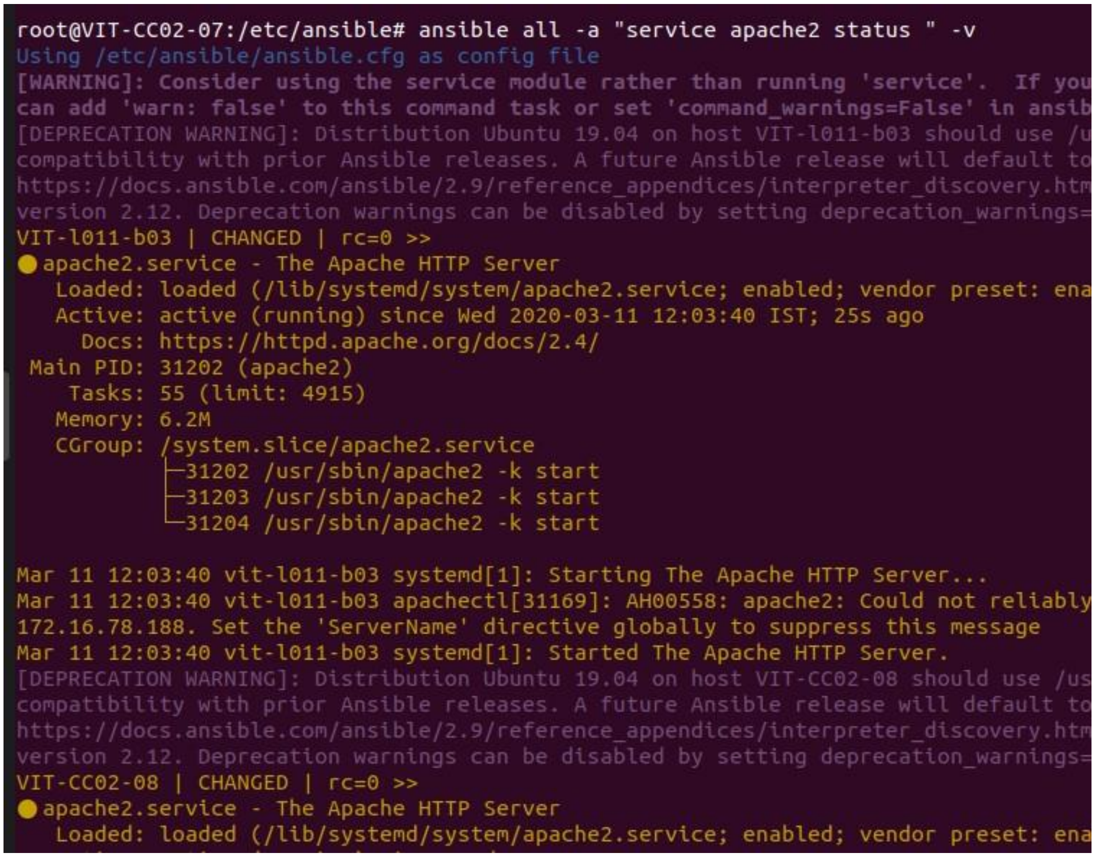
Create group

gedit hosts (enter group name, hostname)
ansible all -a “service apache2 status” -v



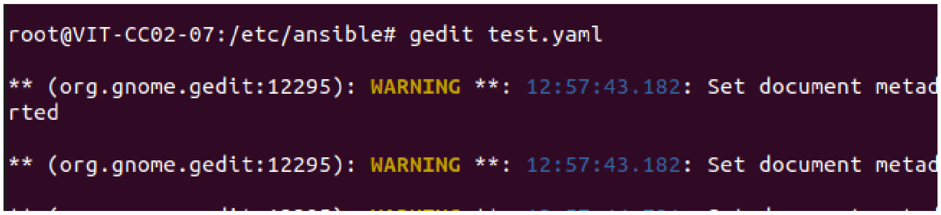
```
root@VIT-CC02-07:/etc/ansible# ansible all -a "service apache2 status " -v
Using /etc/ansible/ansible.cfg as config file
[WARNING]: Consider using the service module rather than running 'service'.  If you need t
can add 'warn: false' to this command task or set 'command_warnings=False' in ansible.cfg 
[DEPRECATION WARNING]: Distribution Ubuntu 19.04 on host VIT-l011-b03 should use /usr/bin/
compatibility with prior Ansible releases. A future Ansible release will default to using 
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for m
version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False i
VIT-l011-b03 | CHANGED | rc=0 >>
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2020-03-11 11:56:49 IST; 15s ago
     Docs: https://httpd.apache.org/docs/2.4/
 Main PID: 15215 (apache2)
    Tasks: 55 (limit: 4915)
   Memory: 6.1M
   CGroup: /system.slice/apache2.service
           ├─15215 /usr/sbin/apache2 -k start
           ├─15217 /usr/sbin/apache2 -k start
           └─15218 /usr/sbin/apache2 -k start

Mar 11 11:56:49 vit-l011-b03 systemd[1]: Starting The Apache HTTP Server...
Mar 11 11:56:49 vit-l011-b03 apachectl[15195]: AH00558: apache2: Could not reliably determ
172.16.78.188. Set the 'ServerName' directive globally to suppress this message
Mar 11 11:56:49 vit-l011-b03 systemd[1]: Started The Apache HTTP Server.
[DEPRECATION WARNING]: Distribution Ubuntu 19.04 on host VIT-CC02-08 should use /usr/bin/p
compatibility with prior Ansible releases. A future Ansible release will default to using 
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for m
version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False i
VIT-CC02-08 | CHANGED | rc=0 >>
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
```

ansible all -a “service apache2 status” -v

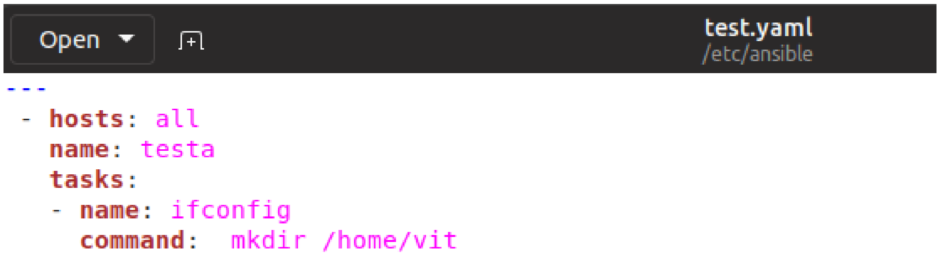


```
root@VIT-CC02-07:/etc/ansible# ansible all -a "service apache2 status " -v
Using /etc/ansible/ansible.cfg as config file
[WARNING]: Consider using the service module rather than running 'service'.  If you
can add 'warn: false' to this command task or set 'command_warnings=False' in ansib
[DEPRECATION WARNING]: Distribution Ubuntu 19.04 on host VIT-l011-b03 should use /u
compatibility with prior Ansible releases. A future Ansible release will default to
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.htm
version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=
VIT-l011-b03 | CHANGED | rc=0 >>
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: ena
   Active: active (running) since Wed 2020-03-11 12:03:40 IST; 25s ago
     Docs: https://httpd.apache.org/docs/2.4/
 Main PID: 31202 (apache2)
    Tasks: 55 (limit: 4915)
   Memory: 6.2M
   CGroup: /system.slice/apache2.service
           ├─31202 /usr/sbin/apache2 -k start
           ├─31203 /usr/sbin/apache2 -k start
           └─31204 /usr/sbin/apache2 -k start

Mar 11 12:03:40 vit-l011-b03 systemd[1]: Starting The Apache HTTP Server...
Mar 11 12:03:40 vit-l011-b03 apachectl[31169]: AH00558: apache2: Could not reliably
172.16.78.188. Set the 'ServerName' directive globally to suppress this message
Mar 11 12:03:40 vit-l011-b03 systemd[1]: Started The Apache HTTP Server.
[DEPRECATION WARNING]: Distribution Ubuntu 19.04 on host VIT-CC02-08 should use /us
compatibility with prior Ansible releases. A future Ansible release will default to
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.htm
version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=
VIT-CC02-08 | CHANGED | rc=0 >>
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: ena
```

gedit test.yaml

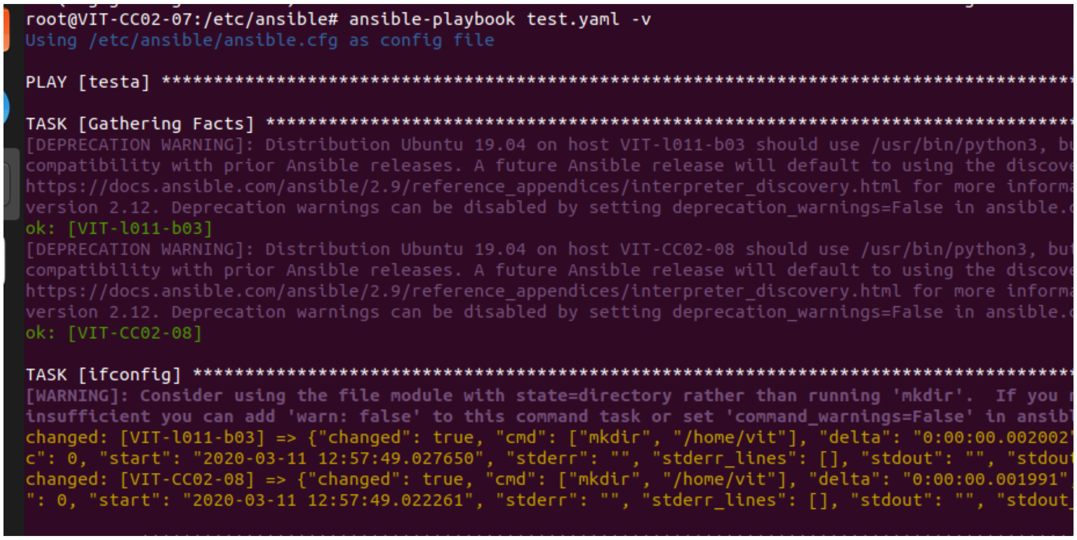


```
root@VIT-CC02-07:/etc/ansible# gedit test.yaml

** (org.gnome.gedit:12295): WARNING **: 12:57:43.182: Set document metad
rted

** (org.gnome.gedit:12295): WARNING **: 12:57:43.182: Set document metad
```



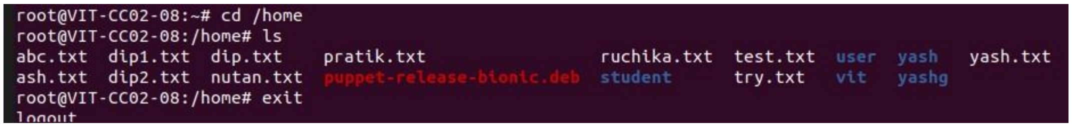
```
---
 - hosts: all
   name: testa
   tasks:
   - name: ifconfig
     command:  mkdir /home/vit
```

ansible-playbook test.yaml -v



```
root@VIT-CC02-07:/etc/ansible# ansible-playbook test.yaml -v
Using /etc/ansible/ansible.cfg as config file

PLAY [testa] ****************************************************************************

TASK [Gathering Facts] ******************************************************************
[DEPRECATION WARNING]: Distribution Ubuntu 19.04 on host VIT-l011-b03 should use /usr/bin/python3, bu
compatibility with prior Ansible releases. A future Ansible release will default to using the discove
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more informa
version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.c
ok: [VIT-l011-b03]
[DEPRECATION WARNING]: Distribution Ubuntu 19.04 on host VIT-CC02-08 should use /usr/bin/python3, but
compatibility with prior Ansible releases. A future Ansible release will default to using the discove
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more informa
version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.c
ok: [VIT-CC02-08]

TASK [ifconfig] *************************************************************************
[WARNING]: Consider using the file module with state=directory rather than running 'mkdir'.  If you r
insufficient you can add 'warn: false' to this command task or set 'command_warnings=False' in ansibl
changed: [VIT-l011-b03] => {"changed": true, "cmd": ["mkdir", "/home/vit"], "delta": "0:00:00.002002"
c": 0, "start": "2020-03-11 12:57:49.027650", "stderr": "", "stderr_lines": [], "stdout": "", "stdout
changed: [VIT-CC02-08] => {"changed": true, "cmd": ["mkdir", "/home/vit"], "delta": "0:00:00.001991",
": 0, "start": "2020-03-11 12:57:49.022261", "stderr": "", "stderr_lines": [], "stdout": "", "stdout_
```

cd /home
ls



```
root@VIT-CC02-08:~# cd /home
root@VIT-CC02-08:/home# ls
abc.txt  dip1.txt  dip.txt    pratik.txt                 ruchika.txt  test.txt  user  yash   yash.txt
ash.txt  dip2.txt  nutan.txt  puppet-release-bionic.deb  student      try.txt   vit   yashg
root@VIT-CC02-08:/home# exit
logout
```

Observations/ Conclusion	We come to know about how to used Ansible for software configuration management.

VIT | Vidyalankar Institute of Technology
Accredited A+ by NAAC

DEPARTMENT OF INFORMATION TECHNOLOGY   Assignment 1

Semester	Semester VIII – INFT Engineering		
Subject	DevOps Lab		
Subject Professor In-charge	Prof. Yash Shah		
Student Name			
Roll Number		Division:	Batch:

Title	Case Study on Selenium

Selenium

Selenium is an open source tool which is used for automating the tests carried out on web browsers (Web applications are tested using any web browser). only testing of web applications is possible with Selenium. We can neither test any desktop (software) application nor test any mobile application using Selenium.

Since Selenium is open-source, there is no licensing cost involved, which is a major advantage over other testing tools.

Other reasons behind Selenium's ever growing popularity are:

- Test scripts can be written in any of these programming languages: Java, Python, C#, PHP, Ruby, Perl & .Net
- Tests can be carried out in any of these OS: Windows, Mac or Linux
- Tests can be carried out using any browser: Mozilla Firefox, Internet Explorer, Google Chrome, Safari or Opera
- It can be integrated with tools such as TestNG & JUnit for managing test cases and generating reports
- It can be integrated with Maven, Jenkins & Docker to achieve Continuous Testing

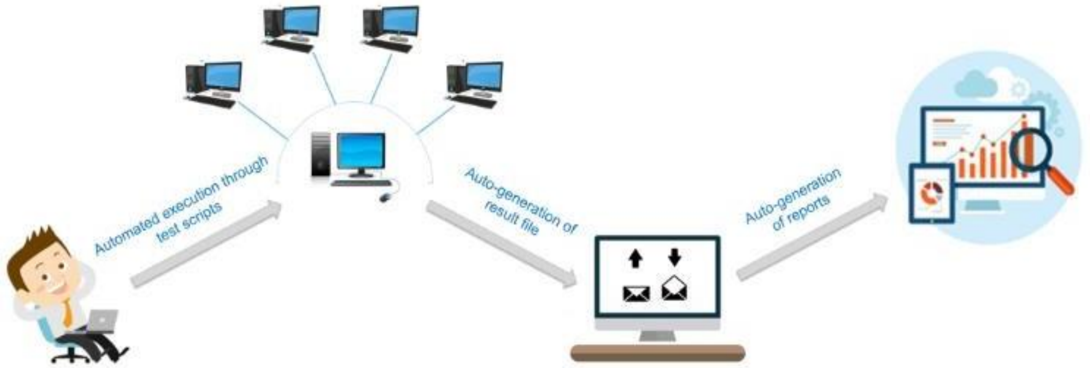
Challenges with Manual Testing

Manual testing means the (web) application is tested manually by QA testers. Tests need to be performed manually in every environment, using a different data set and the success/ failure rate of every transaction should be recorded.

Automation Testing Beats Manual Testing

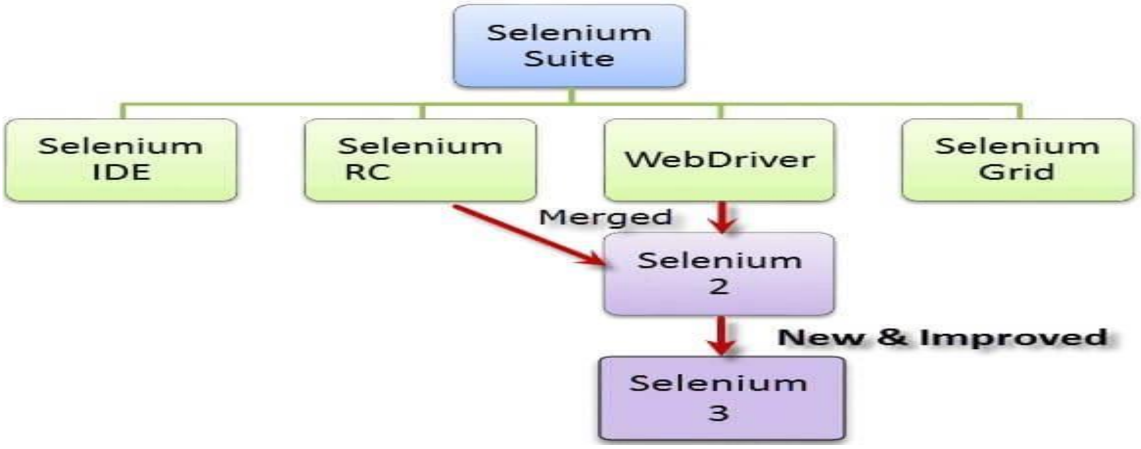
Automation testing beats manual testing every time. Because it is faster, needs less investment in human resource, it is not prone to errors, frequent execution of tests is possible, supports lights out execution, supports regression testing and also functional testing.

Let’s example, Suppose there is a login page and we need to verify if all the login attempts are successful, then it will be really easy to write a piece of code which will validate if all the transaction/ login attempts are a success or not (automated test case execution). Moreover, these tests can be configured in such a way that they are tested in different environments and web browsers.
The key point is that automation testing makes a tester’s job a whole lot simpler.


Automated execution through test scripts
Auto-generation or result file
Auto-generation of reports

Selenium Software is not just a single tool but a suite of software, each piece catering to different testing needs of an organization.
Here is the list of tools:

- Selenium RC (Now depreciated)
- Selenium Integrated Development Environment (IDE)
- Selenium Grid
- Selenium WebDriver



Selenium RC (Remote Control):

- Selenium Core was the first tool. But, Selenium Core hit a roadblock in terms of cross- domain testing because of the same origin policy. Same origin policy prohibits JavaScript code from accessing web elements which are hosted on a different domain compared to where the JavaScript was launched.
- To overcome the same origin policy issue, testers needed to install local copies of both Selenium Core (a JavaScript program) and the web server containing the web application being tested so they would belong to the same domain. This lead to the birth of Selenium RC, which is accredited to then ThoughtWork's engineer, Paul Hammant.
- RC overcame the problem by involving an HTTP proxy server to "trick" the browser into believing that Selenium Core and the web application being tested come from the same domain. Thus making RC a two-component tool.
    1. Selenium RC Server
    2. Selenium RC Client – Library containing your programming language code

Selenium IDE (Integrated Development Environment):

- In 2006, Shinya Kastani from Japan had donated his Selenium IDE prototype to Apache's Selenium project. It was a Firefox plugin for faster creation of test cases. IDE implemented a record and playback model wherein, test cases are created by recording the interactions which the user had with the web browser. These tests can then be played back any number of times.
- The advantage with Selenium IDE is that, tests recorded via the plugin can be exported in different programming languages like: Java, Ruby, Python etc. Check out the below screenshot of Firefox's IDE plugin.

Selenium Grid:

- Selenium Grid was developed by Patrick Lightbody and initially called HostedQA (initially a part of Selenium v1) and it was used in combination with RC to run tests on remote machines. In fact, with Grid, multiple test scripts can be executed at the same time on multiple machines.
- Parallel execution is achieved with the help of Hub-Node architecture. One machine will assume the role of Hub and the others will be the Nodes. Hub controls the test scripts running on various browsers inside various operating systems. Test scripts being executed on different Nodes can be written in different programming languages.
- Grid is still in use and works with both WebDriver and RC. However, maintaining a grid with all required browsers and operating systems is a challenge.

Selenium WebDriver:

- Founded by Simon Stewart in 2006, Selenium WebDriver was the first cross platform testing framework that could control the browser from OS level. In contrast to IDE, Selenium WebDriver provides a programming interface to create and execute test cases. Test cases are written such that, web elements on web pages are identified and then actions are performed on those elements.

- WebDriver is an upgrade to RC because it is much faster. It is faster because it makes direct calls to the browser. RC on the other hand needs an RC server to interact with the web browser. Each browser has its own driver on which the application runs.

The different WebDrivers are:

- Firefox Driver (Gecko Driver)
- Chrome Driver
- Internet Explorer Driver
- Opera Driver
- Safari Driver and
- HTM Unit Driver

DEPARTMENT OF INFORMATION TECHNOLOGY Assignment 2

Semester	Semester VIII – INFT Engineering		
Subject	DevOps Lab		
Subject Professor In-charge	Prof. Yash Shah		
Student Name			
Roll Number		Division:	Batch:
Title	PPT on Project Management Technique		

PROJECT MANAGEMENT TECHNIQUE

HARSHADA DHURI 17101C2058

TABLE OF CONTENT

- What is project management
- Approaches of project management

  Agile project management approach
- **ODHTIO£L** R 8 It t21JS GS
- Benefits

PROJECT MANAGEMENT

- A project is a unique, temporary endeavor with u defiiiite beguuiing and end that’s lindotaken to fulfill a specific set of goals and objectives. As suck the Project ManRgement Institute defines project management as ‘be applicahon of knowledge, akills, tools, and techniques to project activities to meet the project requirements.”

  There are n wide variety of projects and niganizudonal structures, so there’B no one-size-fits-all FI [l£ORch to project management, For project management to be effective, it must be tailored to your organization’s protect typea, human resource cRpabililies, and company

APPROACHES OF PROJECT MANAGEMENT

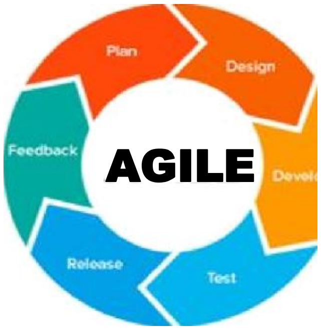
- Trnditinnal Prnject Management
- Waterfall Project Management
- Agile Project Management
- Rational Unified R ncess
- PHRT Project Management
- Critical Path R-oject ManRgement
- Critical Chain Project Management
- Extreme Project Management

AGILE PROJECT MANAGEMENT APPROACHES

Agile Project Management (APM) is am iterahve approach to plannuig and binding project processes.

Agile is a project management inethndolngy that uses short develnpoient cycles called "sprints" to focus on continuous improvement in the development of a product or service.

The main benefit of Agile Project Mrinagenient is its ability to respond tn isaues as they arise throughout the course of the project.



COMMON CHARACTERISTICS

Incremental and Iterafive development

- Agile Methods:
  - c Ernbrac% change and Business value
- Ag_ile Project lenient
  - c Etaipuncal process
- Lean thinking

BENEFITS

- More rapid deployirient of solutions
- Reduced «aste through minimization of resources
- Increased flexibilin' and adaptabilih' to chanee
- Increased success throush nore focused eBorts
- Faster tiimaround ñmes
- Faster detection of issues and defects
- Opnmized development processes
- .Slighter u'eigbt framez'oik
- timai pfoject control
- Increased focus on specific custotiier needs
- Increased frequent,' o:I collabnration and feedback

DRAWBACKS

- As with any other methodology, agile is not well-suited for every project, and sufficient due diligence ia ahvays recommended to identify the beat methodology fm each unique situation.
- Agile may not wnrk as intended if a customer la not clear on goala, the project manager or team ia inexperienced, or if they do not funchon well imder significant preBB .
- Throughout the development process, agile favors the developers, project teams and customer goala, but user's experience. end not necessarily the
- Due to its lesa formal and more dexible prncesseB, agile may not always be easily absorbed within larger more traditional organizations where there are significant amounts of rigidity or flexibility within processes, policies, or teams.
- It may also face problems being used with customers who B larly have rigid processes or operating