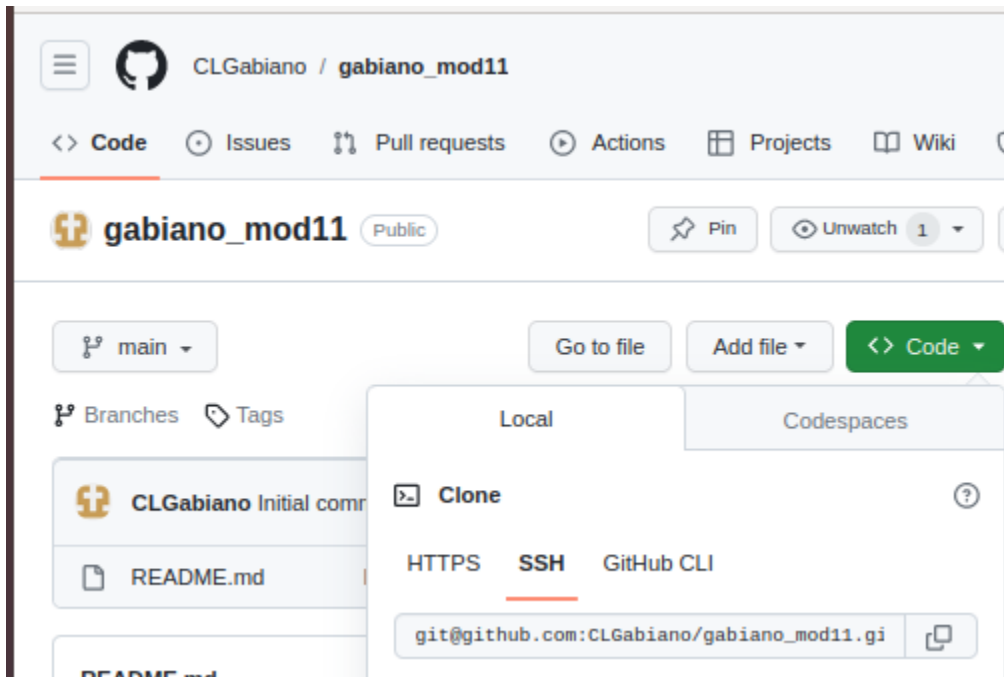


<b>Name: Gabiano, Chris Leonard A.</b>	<b>Date Performed: Nov 16, 2023</b>
<b>Course/Section: CPE31s6</b>	<b>Date Submitted: Nov 16, 2023</b>
<b>Instructor: Engr. Jonathan Taylar</b>	<b>Semester and SY:</b>
<b>Activity 11: Containerization</b>	
<b>1. Objectives</b>	
Create a Dockerfile and form a workflow using Ansible as Infrastructure as Code (IaC) to enable Continuous Delivery process	
<b>2. Discussion</b>	
<p>Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your infrastructure in the same ways you manage your applications. By taking advantage of Docker's methodologies for shipping, testing, and deploying code quickly, you can significantly reduce the delay between writing code and running it in production.</p> <p>Source: <a href="https://docs.docker.com/get-started/overview/">https://docs.docker.com/get-started/overview/</a></p> <p>You may also check the difference between containers and virtual machines. Click the link given below.</p> <p>Source: <a href="https://docs.microsoft.com/en-us/virtualization/windowscontainers/about/containers-vs-vm">https://docs.microsoft.com/en-us/virtualization/windowscontainers/about/containers-vs-vm</a></p>	
<b>3. Tasks</b>	
<ol style="list-style-type: none"> <li>1. Create a new repository for this activity.</li> <li>2. Install Docker and enable the docker socket.</li> <li>3. Add to Docker group to your current user.</li> <li>4. Create a Dockerfile to install web and DB server.</li> <li>5. Install and build the Dockerfile using Ansible.</li> <li>6. Add, commit and push it to your repository.</li> </ol>	
<b>4. Output</b> (screenshots and explanations)	

7. Create a new repository for this activity.



clone the repository to the workstation

```
leonard@workstation:~$ git clone git@github.com:CLGabiano/gabiano_mod11.git
Cloning into 'gabiano_mod11'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
leonard@workstation:~$
```

Install Docker and enable the docker socket.

```

leonard@workstation:~/gabiano_mod11$ systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset:
   Active: active (running) since Thu 2023-11-16 16:44:25 PST; 11min ago
     Docs: https://docs.docker.com
   Main PID: 1090 (dockerd)
      Tasks: 13
     CGroup: /system.slice/docker.service
             └─1090 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/contai

Nov 16 16:44:24 workstation dockerd[1090]: time="2023-11-16T16:44:24.696497805+
Nov 16 16:44:24 workstation dockerd[1090]: time="2023-11-16T16:44:24.696508103+
Nov 16 16:44:24 workstation dockerd[1090]: time="2023-11-16T16:44:24.696512119+
Nov 16 16:44:24 workstation dockerd[1090]: time="2023-11-16T16:44:24.697610993+
Nov 16 16:44:25 workstation dockerd[1090]: time="2023-11-16T16:44:25.498236799+
Nov 16 16:44:25 workstation dockerd[1090]: time="2023-11-16T16:44:25.626834346+
Nov 16 16:44:25 workstation dockerd[1090]: time="2023-11-16T16:44:25.810861985+
Nov 16 16:44:25 workstation dockerd[1090]: time="2023-11-16T16:44:25.811936069+
Nov 16 16:44:25 workstation systemd[1]: Started Docker Application Container En
Nov 16 16:44:25 workstation dockerd[1090]: time="2023-11-16T16:44:25.861834162+

```

Add to Docker group to your current user.

```

leonard@workstation:~/gabiano_mod11$ sudo usermod -aG docker leonard
leonard@workstation:~/gabiano_mod11$
leonard@workstation:~/gabiano_mod11$
leonard@workstation:~/gabiano_mod11$ sudo systemctl restart docker

```

Create a Dockerfile to install the web and DB server.

\*dockerfile

```

GNU nano 2.9.3                                dockerfile

FROM ubuntu
MAINTAINER leonard <qclagabiano@tip.edu.ph>

ARG DEBIAN_FRONTEND=noninteractive

RUN apt-get -y update

RUN apt packages; apt dist-upgrade -y

RUN apt install -y apache2 mariadb-server

ENTRYPOINT apache2ctl -D FOREGROUND

```

Install and build the Dockerfile using Ansible

\*yml file

```
- hosts: web_servers
  become: true
  pre_tasks:

    - name: dpkg for Ubuntu
      shell:
        dpkg --configure -a
      when: ansible_distribution == "Ubuntu"

    - name: Install Docker (Ubuntu)
      apt:
        name: docker
        state: latest
        when: ansible_distribution == "Ubuntu"

    - name: Install SDK (Ubuntu)
      shell:
        pip install docker-py

    - name: Adding group to Docker
      shell:
        usermod -aG docker leonard
```

```
- name: Enable/Restart Docker (Ubuntu)
  service:
    name: docker
    state: started
    enabled: true

- name: Creating Directory for Dockerfile
  file:
    path: ./root/demo-dockerfile
    state: directory
    owner: root
    group: root
```

```
group: root
mode: '0755'

- hosts: db_servers
  become: true
  pre_tasks:

    - name: Install required packages
      yum:
        name:
          - yum-utils
          - device-mapper-persistent-data
          - lvm2
        state: present

    - name: Add Docker repository
      yum_repository:
```

```
name: docker-ce
description: Docker CE Stable - $basearch
baseurl: https://download.docker.com/linux/centos/7/$basearch/stable
gpgkey: https://download.docker.com/linux/centos/gpg
enabled: yes

- name: Install Docker
  yum:
    name: docker-ce
    state: present
```

```
- name: Start and enable Docker service
  systemd:
    name: docker
    state: started
    enabled: yes
```

#### **\*Add to Docker group to your current user.**

```
- name: Adding group to Docker
  shell:
    usermod -aG docker leonard
```

#### **\*Create a Dockerfile to install the web and DB server.**

```
- hosts: web_servers
  become: true
  pre_tasks:

    - name: dpkg for Ubuntu
      shell:
        dpkg --configure -a
      when: ansible_distribution == "Ubuntu"

    - name: Install Docker (Ubuntu)
      apt:
        name: docker
        state: latest
        when: ansible_distribution == "Ubuntu"

    - name: Install SDK (Ubuntu)
      shell:
        pip install docker-py

    - name: Adding group to Docker
      shell:
        usermod -aG docker leonard
```

```
- name: Enable/Restart Docker (Ubuntu)
  service:
    name: docker
    state: started
    enabled: true

- name: Creating Directory for Dockerfile
  file:
    path: ./root/demo-dockerfile
    state: directory
    owner: root
    group: root
```

```
    group: root
    mode: '0755'

- hosts: db_servers
  become: true
  pre_tasks:

    - name: Install required packages
      yum:
        name:
          - yum-utils
          - device-mapper-persistent-data
          - lvm2
        state: present

    - name: Add Docker repository
      yum_repository:
```

```
        name: docker-ce
        description: Docker CE Stable - $basearch
        baseurl: https://download.docker.com/linux/centos/7/$basearch/stable
        gpgkey: https://download.docker.com/linux/centos/gpg
        enabled: yes

    - name: Install Docker
      yum:
        name: docker-ce
        state: present
```

```
    - name: Start and enable Docker service
      systemd:
        name: docker
        state: started
        enabled: yes
```

## Install and build the Dockerfile using Ansible.

```
leonard@SERVER1:~$ systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset:
   Active: active (running) since Thu 2023-11-16 17:55:47 PST; 28min ago
     Docs: https://docs.docker.com
   Main PID: 4969 (dockerd)
    Tasks: 8
   CGroup: /system.slice/docker.service
           └─4969 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/contai

Nov 16 17:55:45 SERVER1 dockerd[4969]: time="2023-11-16T17:55:45.614197437+08:0
Nov 16 17:55:45 SERVER1 dockerd[4969]: time="2023-11-16T17:55:45.614202856+08:0
Nov 16 17:55:45 SERVER1 dockerd[4969]: time="2023-11-16T17:55:45.614205143+08:0
Nov 16 17:55:45 SERVER1 dockerd[4969]: time="2023-11-16T17:55:45.615117942+08:0
Nov 16 17:55:46 SERVER1 dockerd[4969]: time="2023-11-16T17:55:46.296705417+08:0
Nov 16 17:55:46 SERVER1 dockerd[4969]: time="2023-11-16T17:55:46.632119950+08:0
Nov 16 17:55:47 SERVER1 dockerd[4969]: time="2023-11-16T17:55:47.113253289+08:0
Nov 16 17:55:47 SERVER1 dockerd[4969]: time="2023-11-16T17:55:47.115802519+08:0
Nov 16 17:55:47 SERVER1 systemd[1]: Started Docker Application Container Engine
Nov 16 17:55:47 SERVER1 dockerd[4969]: time="2023-11-16T17:55:47.574770009+08:0
lines 1-19/19 (END)
```

```
[leonard@localhost ~]$ systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; vendor preset: disa
   Active: active (running) since Thu 2023-11-16 05:16:51 EST; 3min 1s ago
     Docs: http://docs.docker.com
   Main PID: 6502 (dockerd-current)
    Tasks: 21
   CGroup: /system.slice/docker.service
           └─6502 /usr/bin/dockerd-current --add-runtime docker-runc=/usr/libexec/do...
           └─6510 /usr/bin/docker-containerd-current -l unix:///var/run/docker/libco...

Nov 16 05:16:49 localhost.localdomain dockerd-current[6502]: time="2023-11-16T05:16:...
Nov 16 05:16:51 localhost.localdomain dockerd-current[6502]: time="2023-11-16T05:16:...
Nov 16 05:16:51 localhost.localdomain dockerd-current[6502]: time="2023-11-16T05:16:...
Nov 16 05:16:51 localhost.localdomain dockerd-current[6502]: time="2023-11-16T05:16:...
Nov 16 05:16:51 localhost.localdomain dockerd-current[6502]: time="2023-11-16T05:16:...
Nov 16 05:16:51 localhost.localdomain dockerd-current[6502]: time="2023-11-16T05:16:...
Nov 16 05:16:51 localhost.localdomain dockerd-current[6502]: time="2023-11-16T05:16:...
Nov 16 05:16:51 localhost.localdomain dockerd-current[6502]: time="2023-11-16T05:16:...
Nov 16 05:16:51 localhost.localdomain dockerd-current[6502]: time="2023-11-16T05:16:...
Nov 16 05:16:51 localhost.localdomain systemd[1]: Started Docker Application Contai...
Hint: Some lines were ellipsized, use -l to show in full.
[leonard@localhost ~]$
```



Add, commit and push it to your repository.

```
leonard@workstation:~/gabiano_mod11$ git add *
leonard@workstation:~/gabiano_mod11$ git commit -m "ugh"
[main 576c52e] ugh
 4 files changed, 96 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 dockerfile
 create mode 100644 ins_docker.yml
 create mode 100644 inventory
leonard@workstation:~/gabiano_mod11$ git push origin
Counting objects: 6, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 1.24 KiB | 633.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0)
To github.com:CLGabiano/gabiano_mod11.git
 58a3fc4..576c52e  main -> main
leonard@workstation:~/gabiano_mod11$
```

The screenshot shows the GitHub interface for a repository named 'gabiano\_mod11', which is public. At the top, there are buttons for 'Pin' and 'Unwatch'. Below this, the repository is currently on the 'main' branch, with 1 branch and 0 tags. Action buttons include 'Go to file', 'Add file', and 'Code'. The commit history shows a single commit by 'CLGabiano' with the message 'ugh' at hash '576c52e', made 1 minute ago. This commit includes five files: 'README.md' (initial commit, 1 hour ago), 'ansible.cfg', 'dockerfile', 'ins\_docker.yml', and 'inventory' (all committed 1 minute ago).

File	Commit Message	Commit Hash	Time
README.md	Initial commit	576c52e	1 hour ago
ansible.cfg	ugh	576c52e	1 minute ago
dockerfile	ugh	576c52e	1 minute ago
ins_docker.yml	ugh	576c52e	1 minute ago
inventory	ugh	576c52e	1 minute ago

github link:

[https://github.com/CLGabiano/gabiano\\_mod11.git](https://github.com/CLGabiano/gabiano_mod11.git)

**Reflections:**

Answer the following:

1. What are the benefits of implementing containerizations?

Containerization provides a way to package and isolate applications along with their dependencies, making it easier to deploy and run them consistently across different computing environments. This leads to benefits such as improved efficiency, scalability, and portability, allowing developers to focus on building and deploying applications without worrying about the underlying infrastructure.

**Conclusions:**

In everyday language, containerization is like neatly organizing your apps in self-contained boxes. This simplifies moving them around, saves time, and guarantees they work consistently, no matter where they're placed. It's like having a handy, versatile toolkit for software that streamlines the process of building and running things.