

| | |
|--|---|
| Name: Aldwin Joseph B. Revilla | Date Performed: 11/21/2023 |
| Course/Section: CPE31S5 | Date Submitted: 11/23/2023 |
| Instructor: Engr. Roman Richard | Semester and SY: 1st sem 2023-2024 |

Activity 11: Containerization

1. Objectives

Create a Dockerfile and form a workflow using Ansible as Infrastructure as Code (IaC) to enable Continuous Delivery process

2. Discussion

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.

Source: <https://docs.docker.com/get-started/overview/>

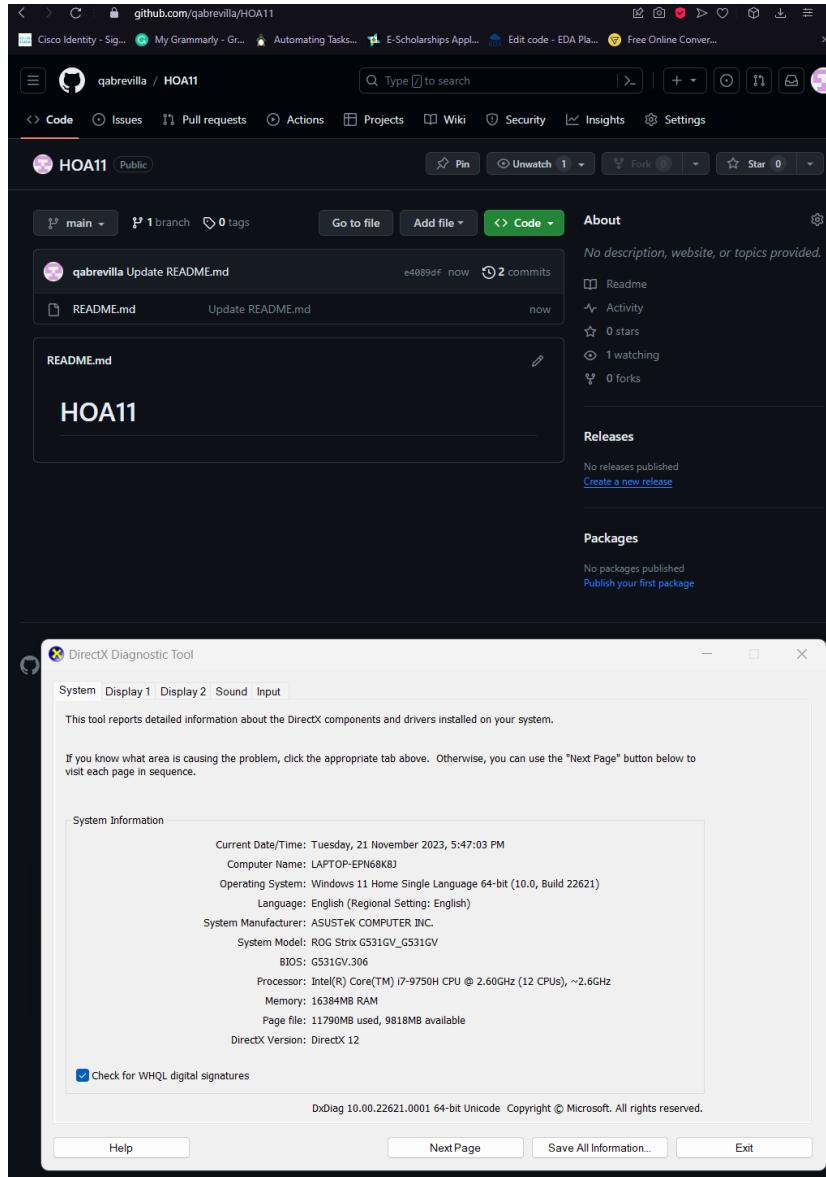
You may also check the difference between containers and virtual machines. Click the link given below.

Source: <https://docs.microsoft.com/en-us/virtualization/windowscontainers/about/containers-vs-vm>

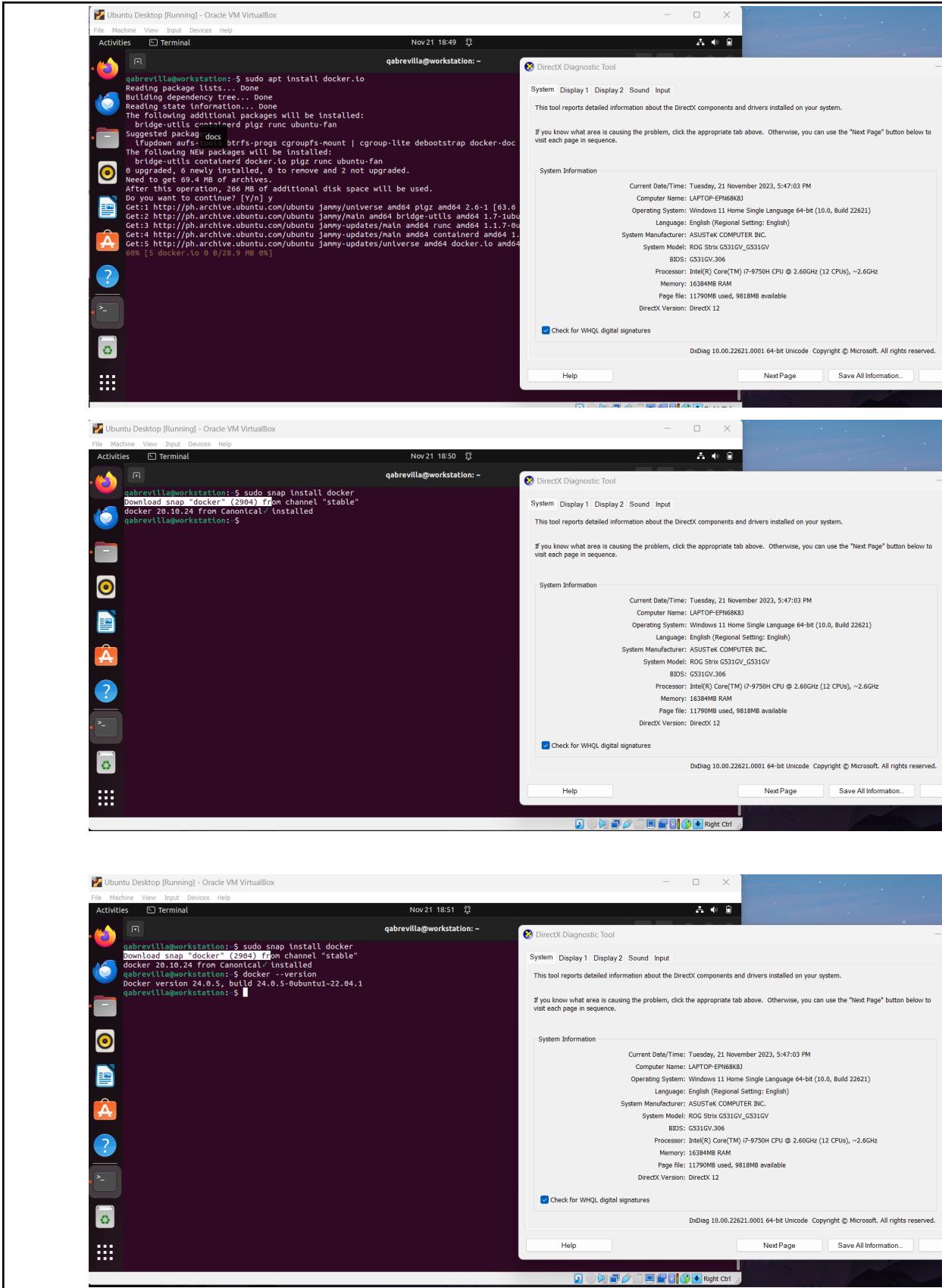
3. Tasks

1. Create a new repository for this activity.
2. Install Docker and enable the docker socket.
3. Add to Docker group to your current user.
4. Create a Dockerfile to install web and DB server.
5. Install and build the Dockerfile using Ansible.
6. Add, commit and push it to your repository.

4. Output (screenshots and explanations)



In github, I've created a new repository named HOA11. I also setup the ssh connection for me to be able to git clone the repository to the ubuntu desktop.



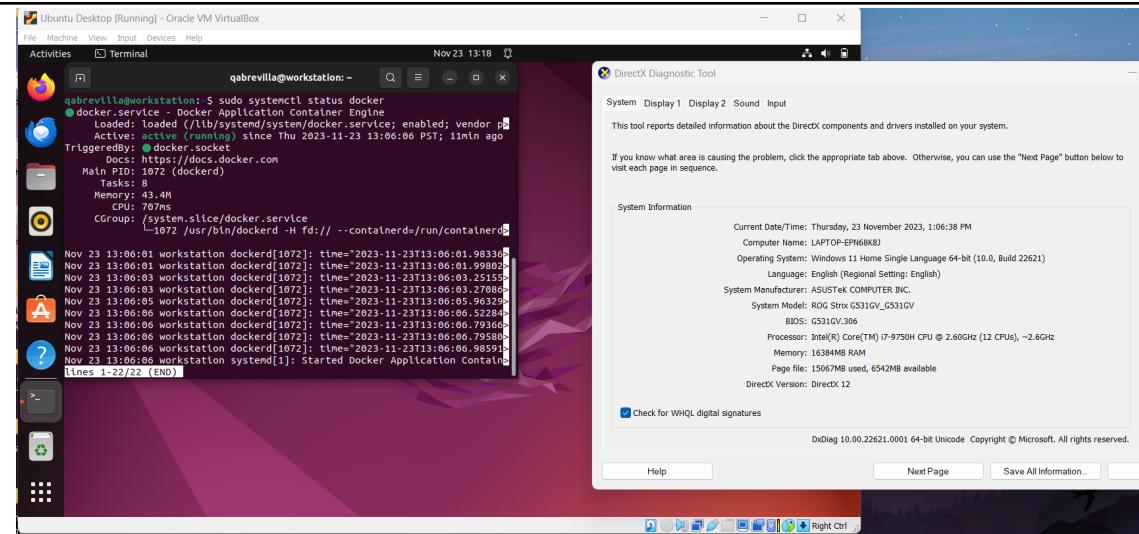


Figure 2 Install Docker and enable the docker socket.

In the official website of docker, we can follow the installation, requirements, and steps to be able to download docker in ubuntu.

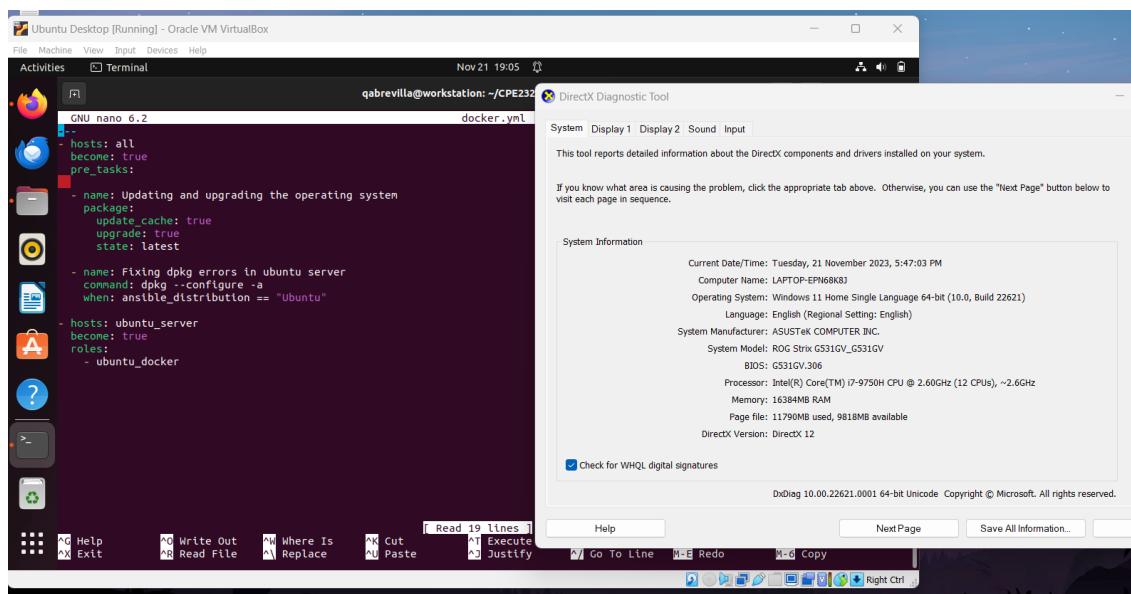


Figure 3 docker.yml

This is the main ansible playbook where it describes the flow of the tasks. It is connected to the directories and runs the contents of each files.

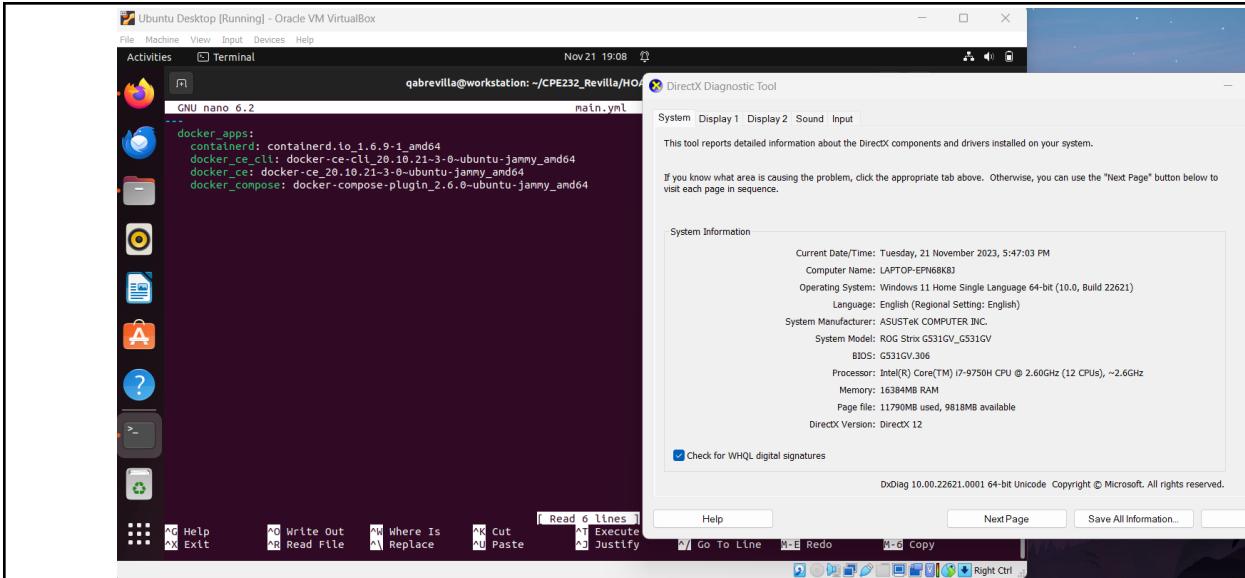


Figure 4 ubuntu_docker/defaults/main.yml

For creating a container for docker, we need to have docker_apps because it is a prerequisite.

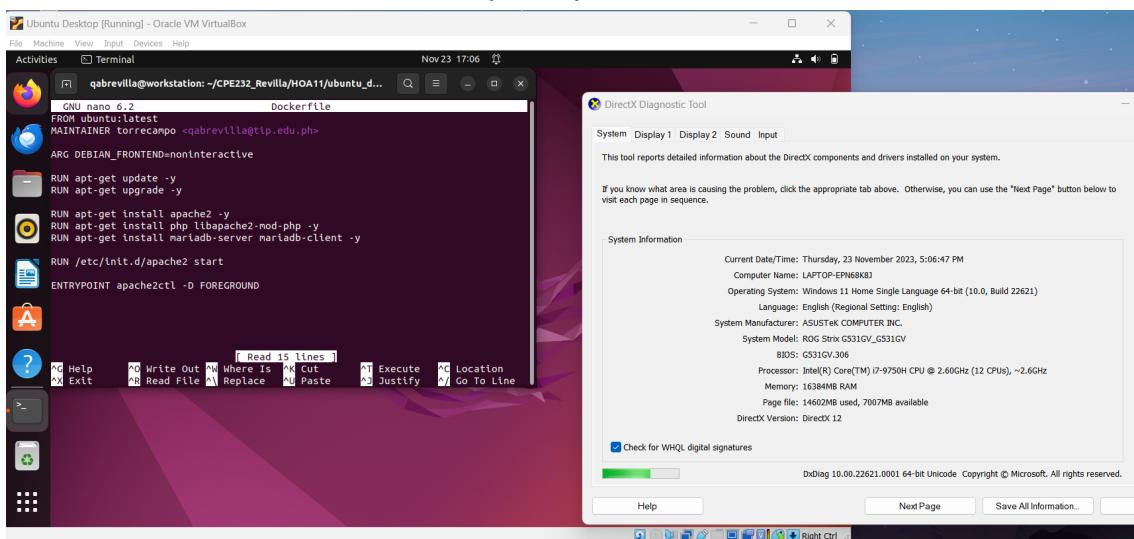


Figure 5 ubuntu_docker/files/Dockerfile

This file will be useful when the installation of docker and configuration occur. The tasks will search for this file as they do their role.

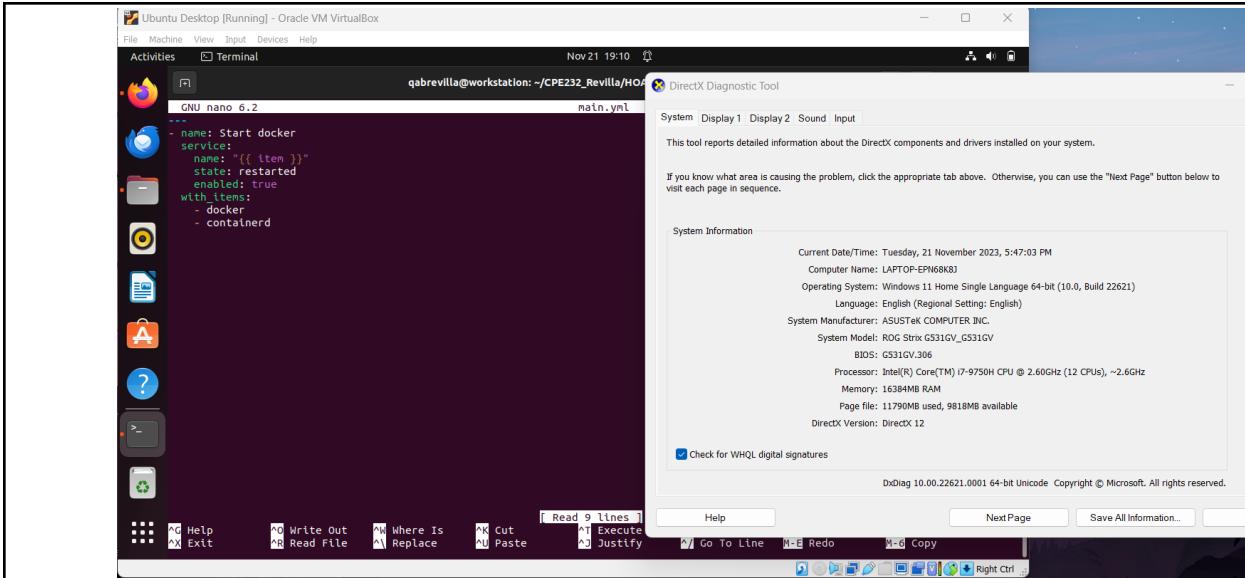
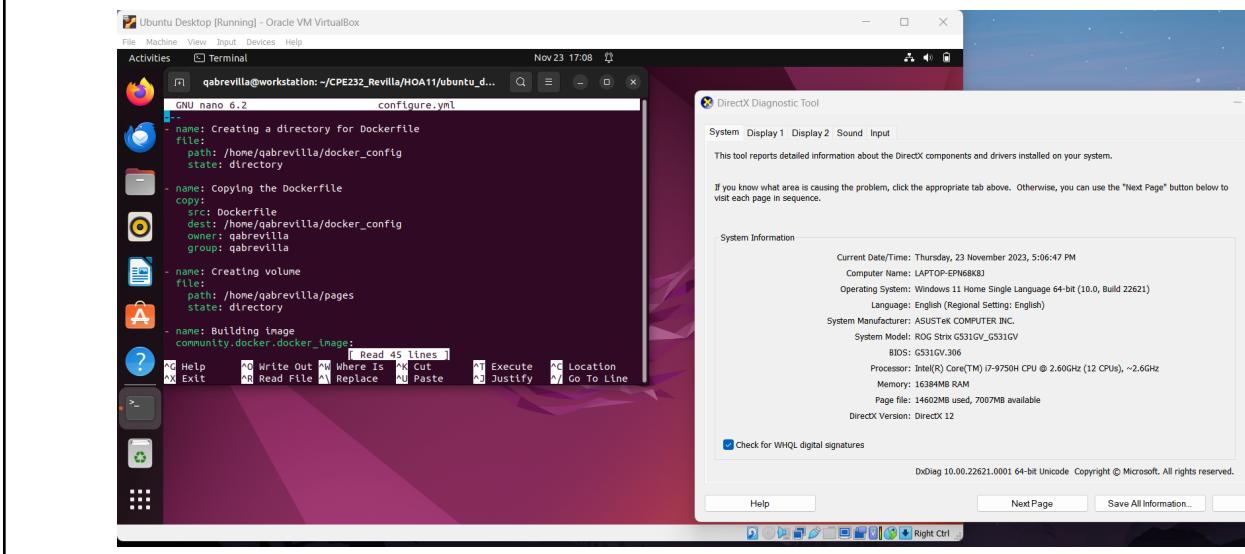
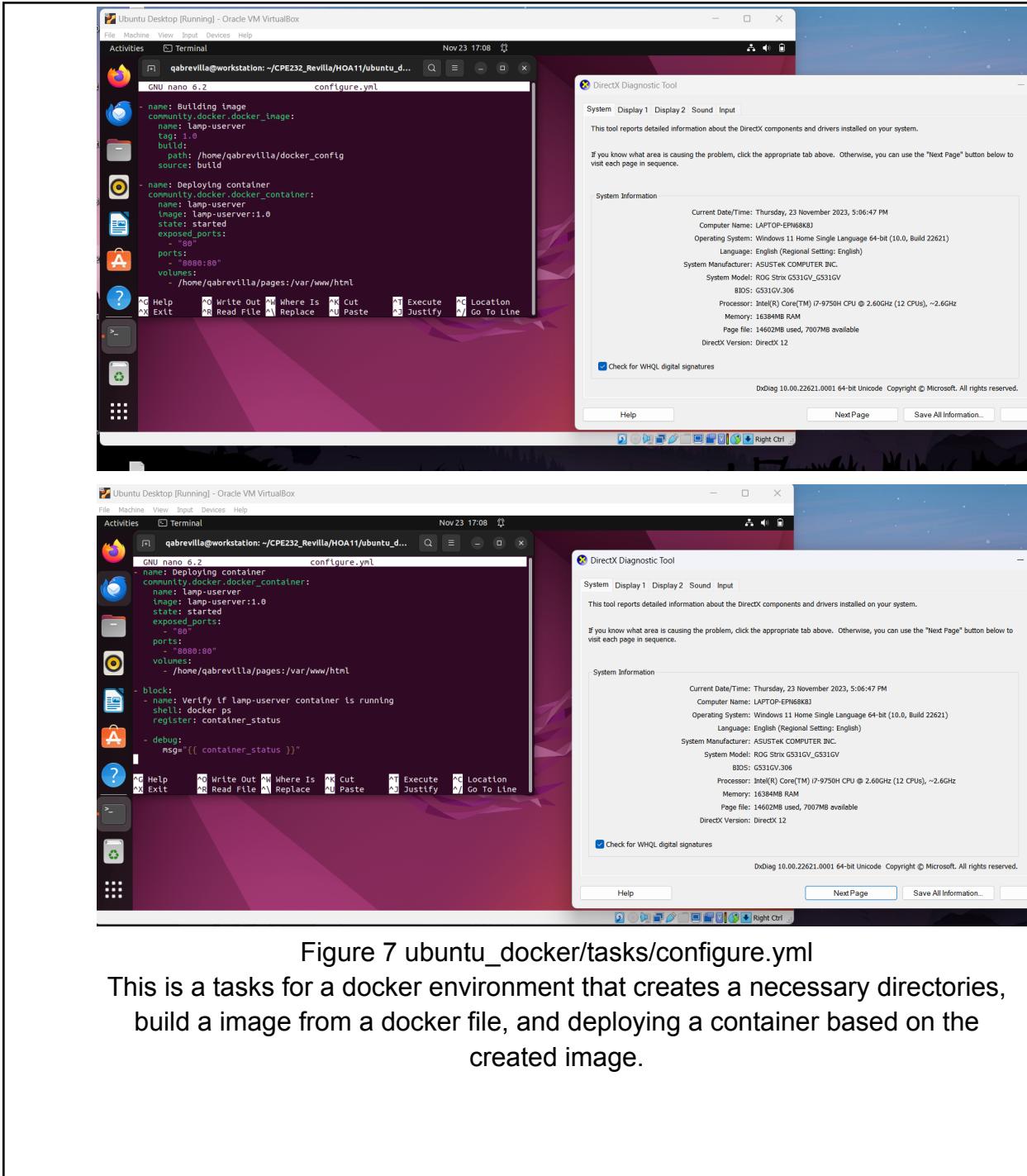


Figure 6 ubuntu_docker/handlers/main.yml

This is the handlers file that contains the tasks that start docker and create containerd.





Activities Terminal Nov 23 17:17

```
qabrevilla@workstation:~/CPE232_Revilla/HOA11/ubuntu_d... GNU nano 6.2 install.yml
- name: Uninstall old Docker versions
  apt:
    name:
      - docker
      - docker-engine
      - docker.io
      - containerd
      - runc
    state: absent

- name: Creating a directory for packages
  file:
    path: /home/userserver/docker-deb
    state: directory

- name: Downloading docker components
  get_url:
    url: "https://download.docker.com/linux/ubuntu/dists/jammy/pool/stable/amd64/{{ item }}"
    dest: /home/userserver/docker-deb
  with_items:
    - "{{ docker_apps.containerd }}.deb"
    - "{{ docker_apps.docker_ce_cli }}.deb"
    - "{{ docker_apps.docker_ce }}.deb"
    - "{{ docker_apps.docker_compose }}.deb"

- name: Installing docker components
  shell:
    - cd /home/userserver/docker-deb
    - dpkg -i "{{ item }}"
  with_items:
```

[Read 92 lines]

Help Write Out Where Is Cut Execute Paste Justify Go To Line

dest: /home/userserver/docker-deb

DirectX Diagnostic Tool

System Display 1 Display 2 Sound Input

This tool reports detailed information about the DirectX components and drivers installed on your system.

If you know what area is causing the problem, click the appropriate tab above. Otherwise, you can use the "Next Page" button below to visit each page in sequence.

System Information

Current Date/Time: Thursday, 23 November 2023, 5:06:47 PM
Computer Name: LAPTOP-EPN68K8J
Operating System: Windows 11 Home Single Language 64-bit (10.0, Build 22621)
Language: English (Regional Setting: English)
System Manufacturer: ASUSTek COMPUTER INC.
System Model: ROG Strix G531GV_G531GV
BIOS: G531GV_306
Processor: Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz (12 CPUs), ~2.6GHz
Memory: 16384MB RAM
Page file: 14602MB used, 7007MB available
DirectX Version: DirectX 12

Check for WHQL digital signatures

DxDiag 10.00.22621.0001 64-bit Unicode Copyright © Microsoft. All rights reserved.

Help Next Page Save All Information...

Activities Terminal Nov 23 17:17

```
qabrevilla@workstation:~/CPE232_Revilla/HOA11/ubuntu_d... GNU nano 6.2 install.yml
- name: Installing docker components
  shell:
    - cd /home/userserver/docker-deb
    - dpkg -i "{{ item }}"
  with_items:
    - "{{ docker_apps.containerd }}.deb"
    - "{{ docker_apps.docker_ce_cli }}.deb"
    - "{{ docker_apps.docker_ce }}.deb"
    - "{{ docker_apps.docker_compose }}.deb"

- name: Fixing /var/run/docker.sock error
  shell: chmod 666 /var/run/docker.sock

- name: Ensure group docker exists
  group:
    name: docker
    state: present

- name: Adding docker to the group of the current user
  user:
    name: userserver
    groups: docker
    append: yes

- name: Start docker services
  service:
    name: "{{ item }}"
    state: started
  with_items:
    - docker
    - containerd
```

Help Write Out Where Is Cut Execute Paste Justify Go To Line

dest: /home/userserver/docker-deb

DirectX Diagnostic Tool

System Display 1 Display 2 Sound Input

This tool reports detailed information about the DirectX components and drivers installed on your system.

If you know what area is causing the problem, click the appropriate tab above. Otherwise, you can use the "Next Page" button below to visit each page in sequence.

System Information

Current Date/Time: Thursday, 23 November 2023, 5:06:47 PM
Computer Name: LAPTOP-EPN68K8J
Operating System: Windows 11 Home Single Language 64-bit (10.0, Build 22621)
Language: English (Regional Setting: English)
System Manufacturer: ASUSTek COMPUTER INC.
System Model: ROG Strix G531GV_G531GV
BIOS: G531GV_306
Processor: Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz (12 CPUs), ~2.6GHz
Memory: 16384MB RAM
Page file: 14602MB used, 7007MB available
DirectX Version: DirectX 12

Check for WHQL digital signatures

DxDiag 10.00.22621.0001 64-bit Unicode Copyright © Microsoft. All rights reserved.

Help Next Page Save All Information...

Activities Terminal Nov 23 17:18

```
qabrevilla@workstation:~/CPE232_Revilla/HOA11/ubuntu_d... GNU nano 6.2 install.yml
cd /home/userserver/docker-deb
dpkg -i {{ item }}
with_items:
  - "{{ docker_apps.containerd }}.deb"
  - "{{ docker_apps.docker_ce_cli }}.deb"
  - "{{ docker_apps.docker_ce }}.deb"
  - "{{ docker_apps.docker_compose }}.deb"

- name: Fixing /var/run/docker.sock error
  shell: chmod 666 /var/run/docker.sock

- name: Ensure group docker exists
  group:
    name: docker
    state: present

- name: Adding docker to the group of the current user
  user:
    name: userserver
    groups: docker
    append: yes

- name: Start docker services
  service:
    name: "{{ item }}"
    state: started
  with_items:
    - docker
    - containerd

- name: Install python
```

Help Write Out Where Is Cut Execute Paste Justify Go To Line

dest: /home/userserver/docker-deb

DirectX Diagnostic Tool

System Display 1 Display 2 Sound Input

This tool reports detailed information about the DirectX components and drivers installed on your system.

If you know what area is causing the problem, click the appropriate tab above. Otherwise, you can use the "Next Page" button below to visit each page in sequence.

System Information

Current Date/Time: Thursday, 23 November 2023, 5:06:47 PM
Computer Name: LAPTOP-EPN68K8J
Operating System: Windows 11 Home Single Language 64-bit (10.0, Build 22621)
Language: English (Regional Setting: English)
System Manufacturer: ASUSTek COMPUTER INC.
System Model: ROG Strix G531GV_G531GV
BIOS: G531GV_306
Processor: Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz (12 CPUs), ~2.6GHz
Memory: 16384MB RAM
Page file: 14602MB used, 7007MB available
DirectX Version: DirectX 12

Check for WHQL digital signatures

DxDiag 10.00.22621.0001 64-bit Unicode Copyright © Microsoft. All rights reserved.

Help Next Page Save All Information...

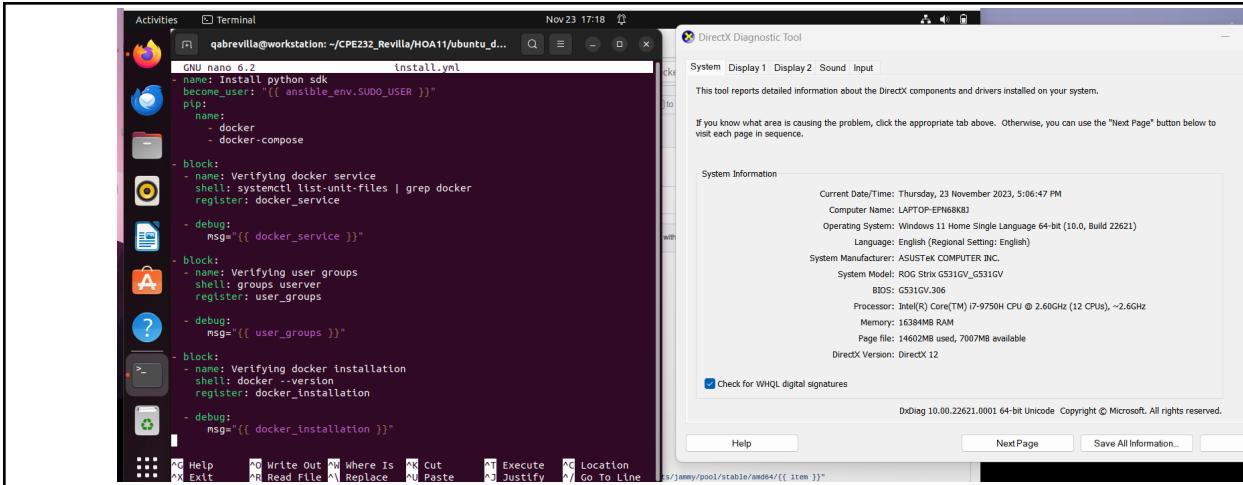


Figure 8 ubuntu_docker/tasks/install.yml

This is a tasks that installs docker on the remote server (ubuntu server). It contains a installation of docker, docker engine, docker.io, and containerd. It also has the requirements needed for them to work on Ubuntu distribution.

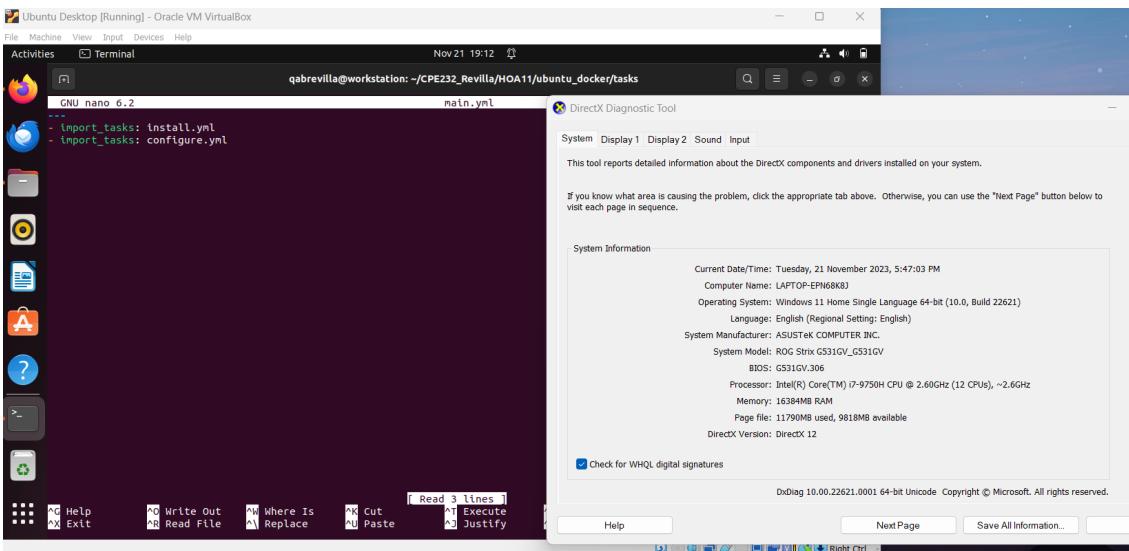


Figure 9 ubuntu_docker/tasks/main.yml

It calls the 2 main tasks, the install.yml and the configure.yml both needed to perform the tasks.

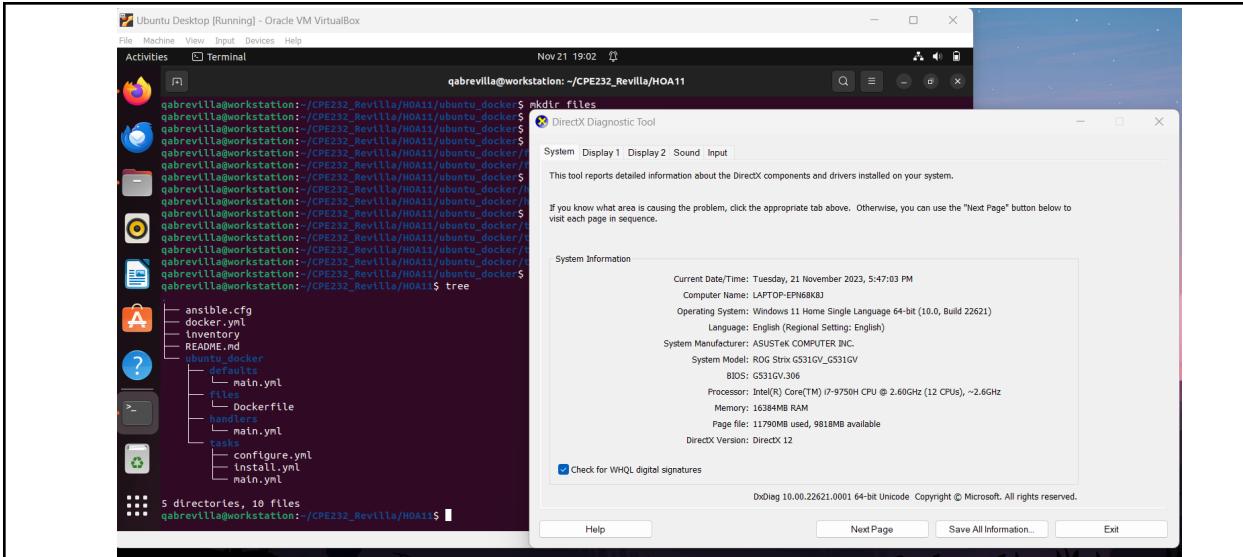
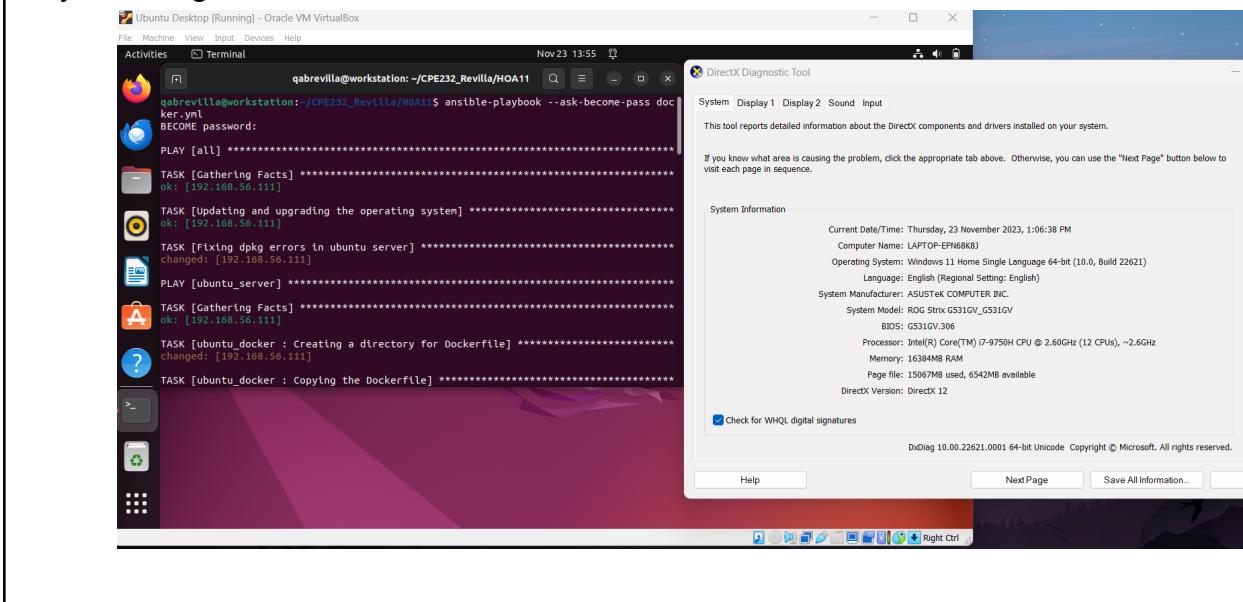


Figure 10 overview of the directory

By installing the tree command, it can be used to show the directories' architecture.



Running the ansible playbook

```

qabrevilla@workstation: ~/CPE232_Revilla/HOA11
TASK [ubuntu_docker : Verifying user groups] *****
ok: [192.168.56.111] => {
    "changed": true,
    "cmd": "groups userver",
    "delta": "0:00:00.006522",
    "end": "2023-11-23 17:23:51.013778",
    "failed": false,
    "msg": "",
    "rc": 0,
    "start": "2023-11-23 17:23:51.007256",
    "stderr": "",
    "stderr_lines": [],
    "stdout": "userver : userver docker",
    "stdout_lines": [
        "userver : userver docker"
    ]
}

TASK [ubuntu_docker : Verifying docker installation] *****
ok: [192.168.56.111] => {

TASK [ubuntu_docker : debug] *****
ok: [192.168.56.111] => {
    "changed": true,
    "cmd": "docker --version",
    "delta": "0:00:00.120661",
    "end": "2023-11-23 17:23:51.576525",
    "failed": false,
    "msg": ""
}

```



```

qabrevilla@workstation: ~/CPE232_Revilla/HOA11
TASK [ubuntu_docker : Verifying docker installation] *****
ok: [192.168.56.111] => {

TASK [ubuntu_docker : debug] *****
ok: [192.168.56.111] => {
    "changed": true,
    "cmd": "docker --version",
    "delta": "0:00:00.120661",
    "end": "2023-11-23 17:23:51.576525",
    "failed": false,
    "msg": ""
}

```



```

qabrevilla@workstation: ~/CPE232_Revilla/HOA11
TASK [ubuntu_docker : Creating a directory for Dockerfile] *****
ok: [192.168.56.111] => {
    "changed": true
}

```



```

qabrevilla@workstation: ~/CPE232_Revilla/HOA11
TASK [ubuntu_docker : Copying the Dockerfile] *****
ok: [192.168.56.111] => {
    "changed": true
}

```



```

qabrevilla@workstation: ~/CPE232_Revilla/HOA11
TASK [ubuntu_docker : Creating volume] *****
ok: [192.168.56.111] => {
    "changed": true
}

```



```

qabrevilla@workstation: ~/CPE232_Revilla/HOA11
TASK [ubuntu_docker : Building image] *****
changed: [192.168.56.111]

```



```

qabrevilla@workstation: ~/CPE232_Revilla/HOA11
TASK [ubuntu_docker : Deploying container] *****
changed: [192.168.56.111]

```



```

qabrevilla@workstation: ~/CPE232_Revilla/HOA11
TASK [ubuntu_docker : Verify if lamp-server container is running] *****
changed: [192.168.56.111]

```



```

qabrevilla@workstation: ~/CPE232_Revilla/HOA11
TASK [ubuntu_docker : debug] *****
ok: [192.168.56.111] => {
    "changed": true,
    "cmd": "docker ps",
    "delta": "0:00:00.029976",
    "end": "2023-11-23 17:26:26.263462",
    "failed": false,
    "msg": "",
    "rc": 0,
    "start": "2023-11-23 17:26:26.233486",
    "stderr": "",
    "stderr_lines": [],
    "stdout": "CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
EATED  STATUS PORTS NAMES
0e90b9d3223a lamp-server:1.0 '/bin/sh -c 'apache2.\'' 1 second ago Up Less than a se
cond ago 0.0.0.0:8000->80/tcp lamp-server",
    "stdout_lines": [
        "CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
0e90b9d3223a lamp-server:1.0 '/bin/sh -c 'apache2.\'' 1 seco
nd ago Up Less than a second 0.0.0.0:8000->80/tcp lamp-server"
    ]
}

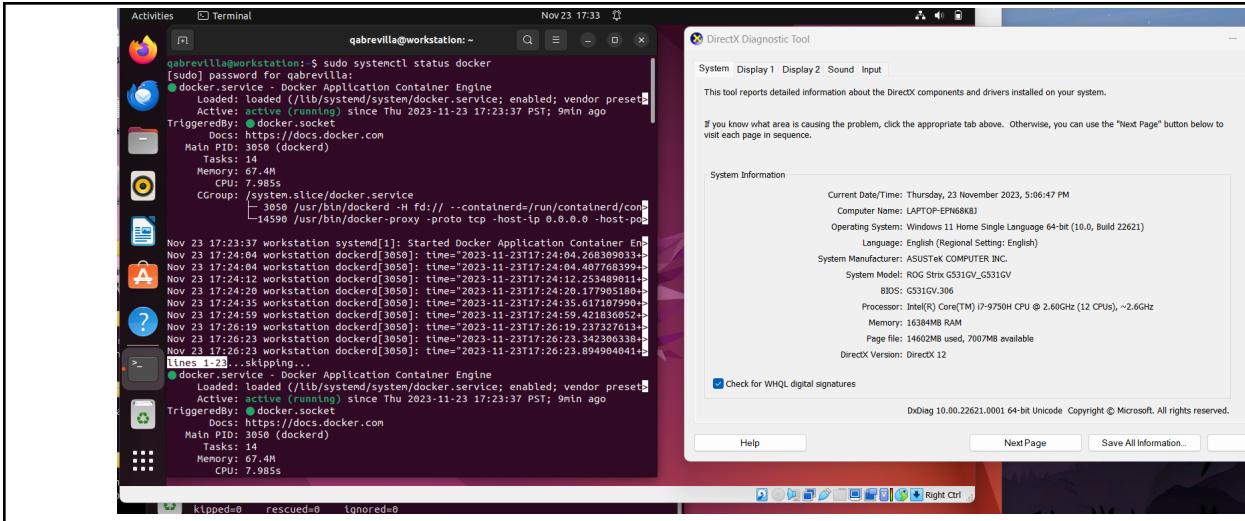
```



```

PLAY RECAP *****
192.168.56.111 : ok=27 changed=13 unreachable=0 failed=0 s

```



Proof that it was successfully installed in the remote server

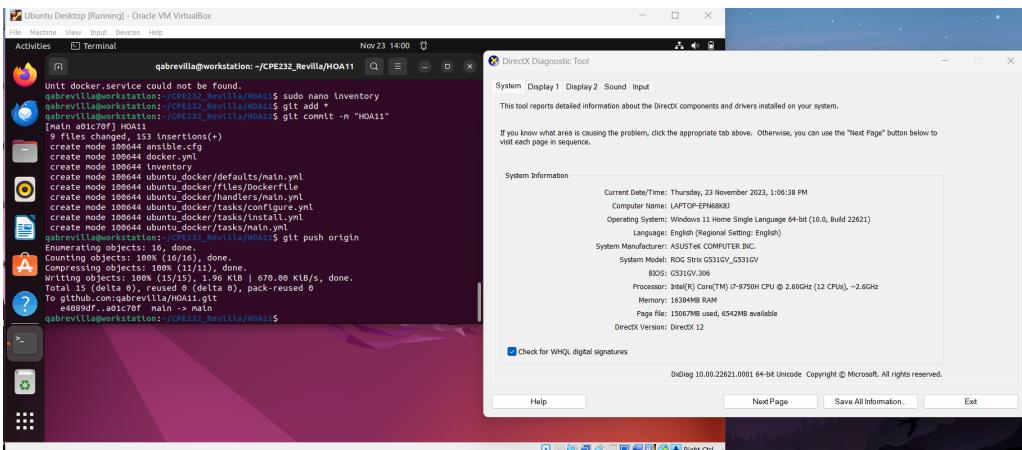


Figure 11 Git push

After the hands-on-activity 11 is done, we need to save it to the repository.

The screenshot shows a GitHub repository named 'HOA11' owned by 'qabrevilla'. The repository is public and contains one branch ('main') and three commits. The latest commit was pushed a minute ago. The repository includes files like 'ubuntu_docker', 'README.md', 'ansible.cfg', 'docker.yml', and 'inventory'. To the right of the repository details, there's an 'About' section with a note: 'No description, website, or topics provided.' It also lists 'Readme', 'Activity', '0 stars', '1 watching', and '0 forks'. Below the repository details is a 'Releases' section stating 'No releases published'. Overlaid on the bottom half of the screen is the 'DirectX Diagnostic Tool' window. The tool has tabs for 'System', 'Display 1', 'Display 2', 'Sound', and 'Input'. The 'System' tab is selected, showing system information: Current Date/Time (Thursday, 23 November 2023, 1:06:38 PM), Computer Name (LAPTOP-EPN68K8), Operating System (Windows 11 Home Single Language 64-bit (10.0, Build 22621)), Language (English (Regional Setting: English)), System Manufacturer (ASUSTeK COMPUTER INC.), System Model (ROG Strix G531GV_G531GV), BIOS (G531GV.306), Processor (Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz (12 CPUs), ~2.6GHz), Memory (16384MB RAM), Page file (15067MB used, 6542MB available), and DirectX Version (DirectX 12). A checkbox for 'Check for WHQL digital signatures' is checked. At the bottom of the tool window, it says 'DxDiag 10.00.22621.0001 64-bit Unicode Copyright © Microsoft. All rights reserved.' Below the tool window are buttons for 'Help', 'Next Page', 'Save All Information...', and 'Exit'. The footer of the GitHub page includes links for Terms, Privacy, Security, Status, Docs, Contact GitHub, Pricing, API, Training, Blog, and About.

Proof that it was successfully saved and the github was updated.

Reflections:

Answer the following:

1. What are the benefits of implementing containerizations?

Using containerization, it will help create an efficient environment for the application. It isolates the application using Linux Kernels features (namespaces and cgroups) to run them without interfering with other processes or other containers under the same hosts. Containers are lightweight because they use the same hosts' kernel meaning fewer resources are needed compared to the partitioning of traditional virtual machines. Using docker containers is a software deployment technique that provides consistency in the system, promotes portability, and isolation development.

Conclusions:

In this activity, we are able to introduce another system administration requirement, using docker and containers. Docker is a program that provides a platform for developers to create packages, distribute, and run applications in a much more efficient way. I also learned about the containers and how we can use the basic idea of using it. These system tools are helpful in managing large contents and help provide an isolated environment for applications to remove the risk of interference and noises when running applications. Installing docker in the traditional way is easy but it requires a lot of time when doing it repeatedly. The challenge for us is to create a Ansible playbook that installs and configures an initial setup for the docker. I found many problems in my first attempt, especially in the file paths. To solve this I need to check and correct the path of the created file/directory because I was not consistent in the file path. Overall the experience is good and It was a relief when the ansible playbook works and did the tasks successfully.