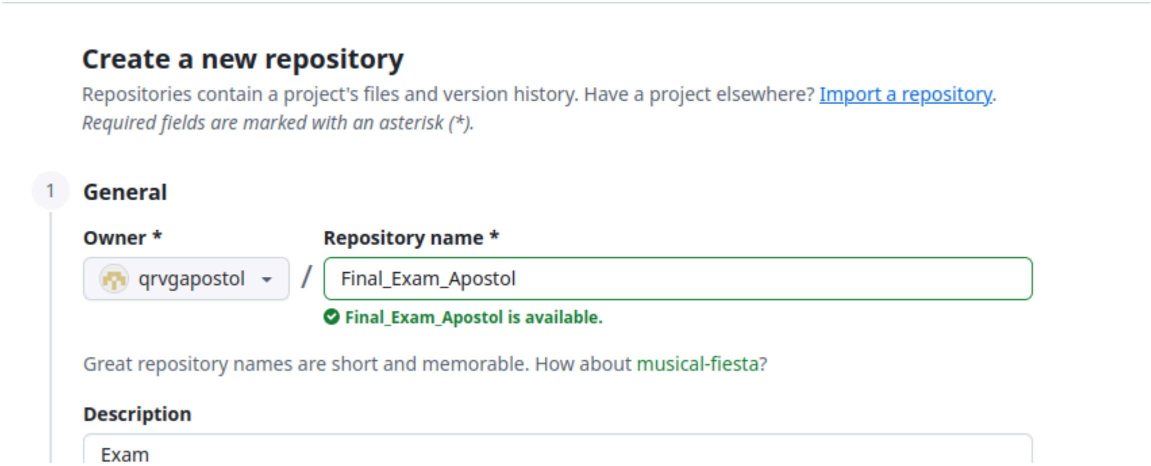
Name: Ruud Van G. Apostol	**Date:** 11/14/25
Course / Section: CPE212 / CPE31S4	**Instructor:** Engr. Robin Valenzuela

Procedure:

1. Create a repository and label it as "Final_Exam_Surname"



Create a new repository

Repositories contain a project's files and version history. Have a project elsewhere? Import a repository.
Required fields are marked with an asterisk (*).

1 **General**

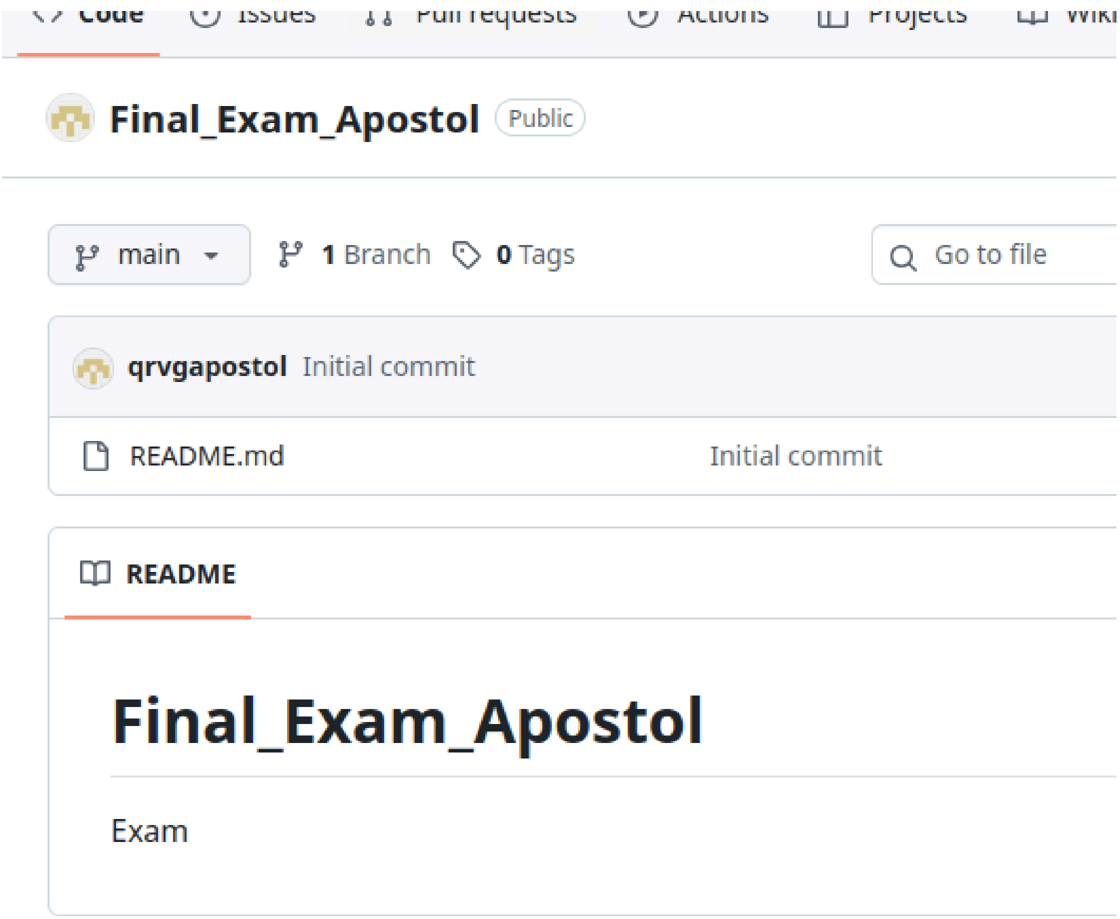
Owner * qrvgapostol / **Repository name *** Final_Exam_Apostol

Final_Exam_Apostol is available.

Great repository names are short and memorable. How about musical-fiesta?

Description

Exam



Final_Exam_Apostol Public

main · 1 Branch · 0 Tags · Go to file

qrvgapostol Initial commit

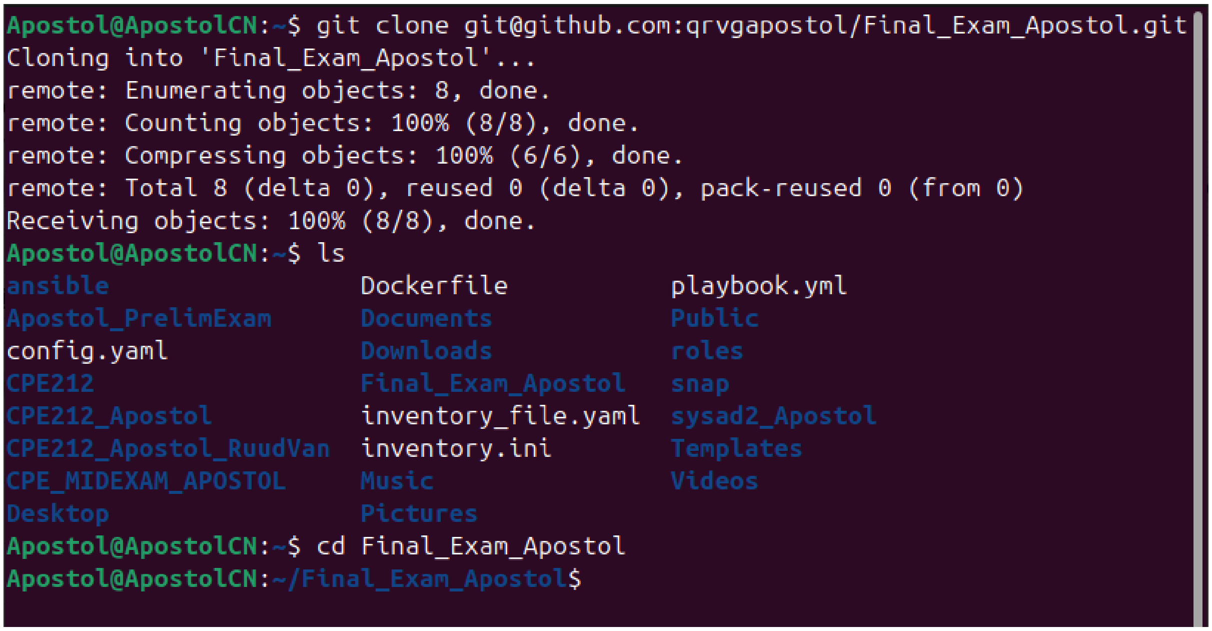
README.md — Initial commit

README

Final_Exam_Apostol

Exam

2. Clone your new repository in your VM

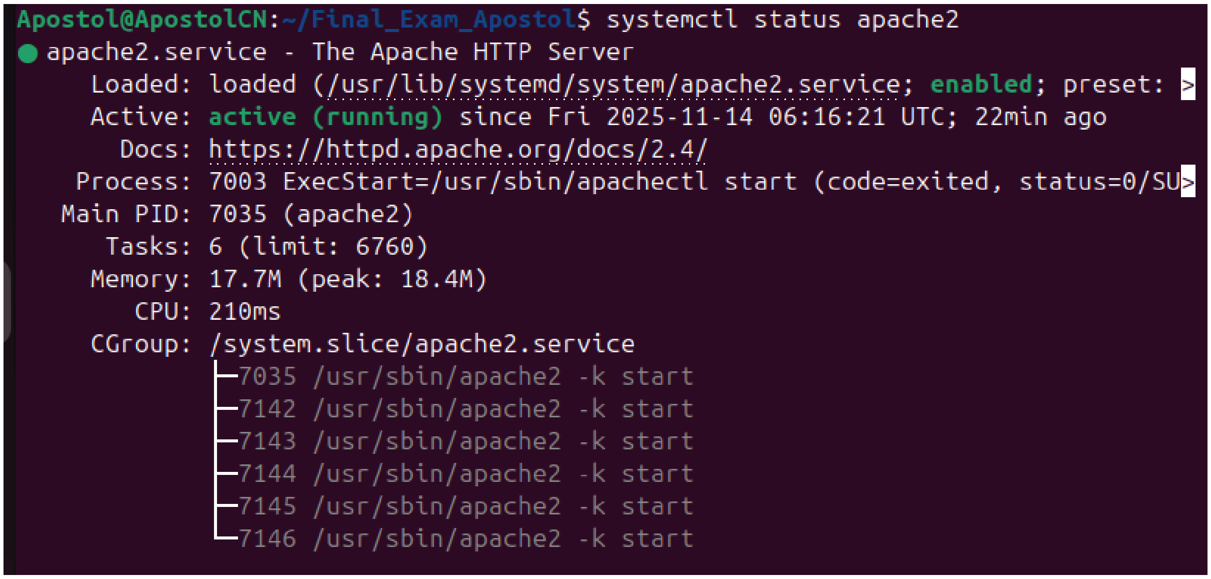


```
Apostol@ApostolCN:~$ git clone git@github.com:qrvgapostol/Final_Exam_Apostol.git
Cloning into 'Final_Exam_Apostol'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 8 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (8/8), done.
Apostol@ApostolCN:~$ ls
ansible                 Dockerfile           playbook.yml
Apostol_PrelimExam      Documents            Public
config.yaml             Downloads            roles
CPE212                  Final_Exam_Apostol   snap
CPE212_Apostol          inventory_file.yaml  sysad2_Apostol
CPE212_Apostol_RuudVan  inventory.ini        Templates
CPE_MIDEXAM_APOSTOL     Music                Videos
Desktop                 Pictures
Apostol@ApostolCN:~$ cd Final_Exam_Apostol
Apostol@ApostolCN:~/Final_Exam_Apostol$
```

3. Create an Ansible playbook that does the following with an input of a config.yaml file and structure inventory file.

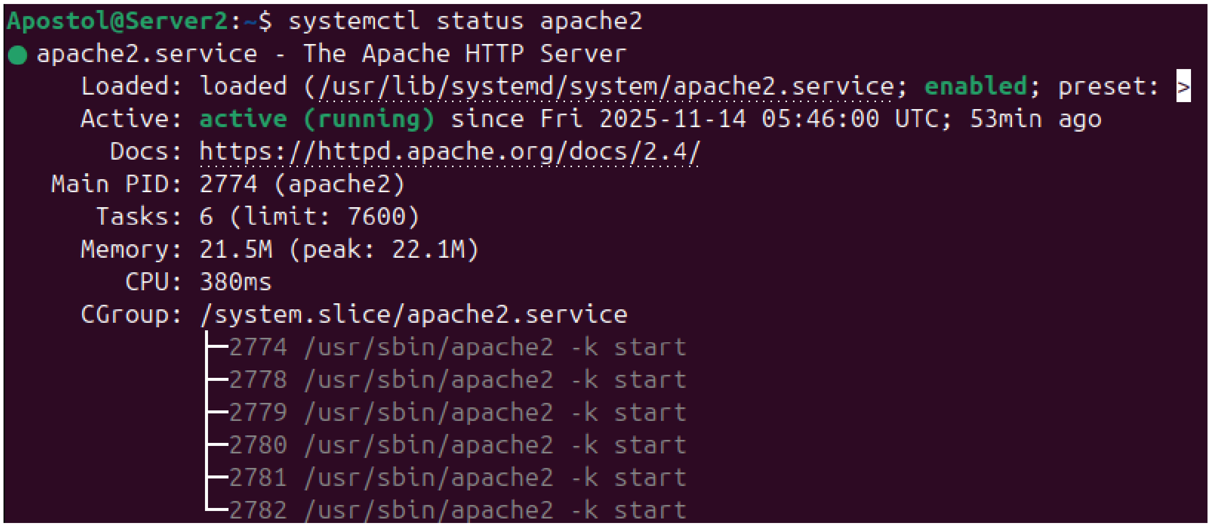
3.1 Install and configure one enterprise service that can be installed in Debian and Centos servers

Control Node:



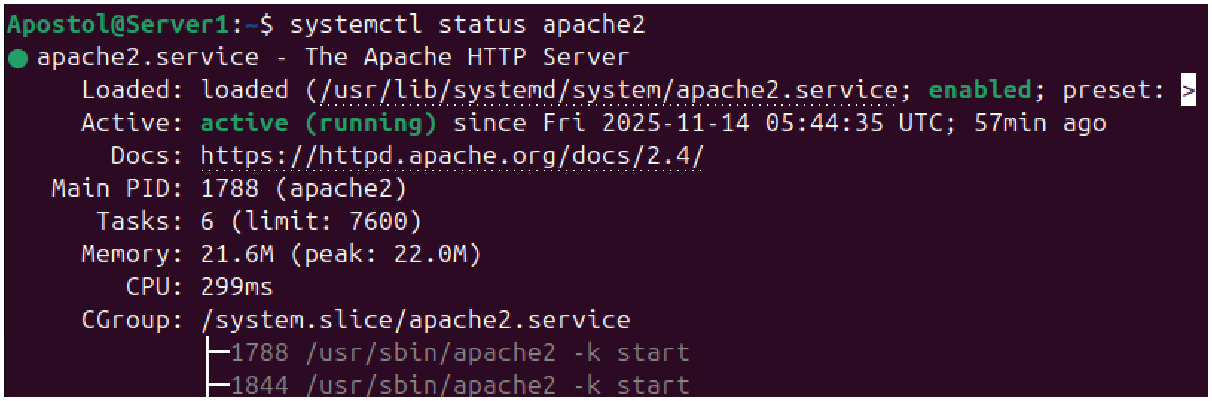
```
Apostol@ApostolCN:~/Final_Exam_Apostol$ systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: >
     Active: active (running) since Fri 2025-11-14 06:16:21 UTC; 22min ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 7003 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SU>
   Main PID: 7035 (apache2)
      Tasks: 6 (limit: 6760)
     Memory: 17.7M (peak: 18.4M)
        CPU: 210ms
     CGroup: /system.slice/apache2.service
             ├─7035 /usr/sbin/apache2 -k start
             ├─7142 /usr/sbin/apache2 -k start
             ├─7143 /usr/sbin/apache2 -k start
             ├─7144 /usr/sbin/apache2 -k start
             ├─7145 /usr/sbin/apache2 -k start
             └─7146 /usr/sbin/apache2 -k start
```

Server2:



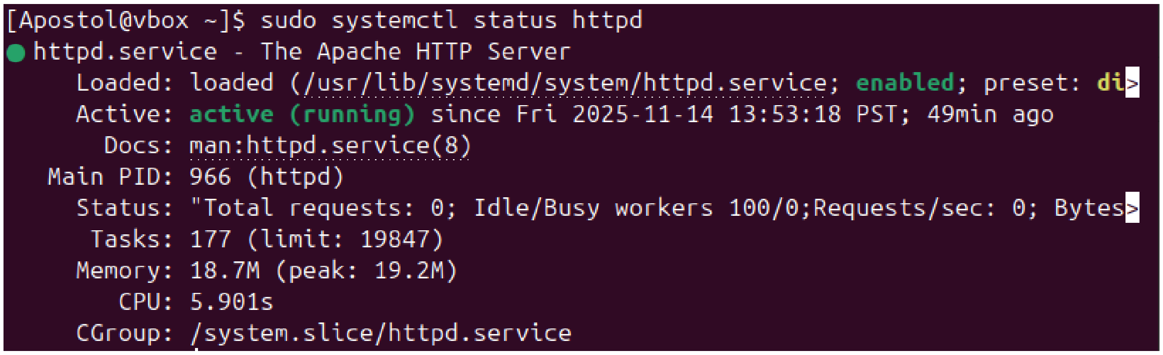
```
Apostol@Server2:~$ systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: >
     Active: active (running) since Fri 2025-11-14 05:46:00 UTC; 53min ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2774 (apache2)
      Tasks: 6 (limit: 7600)
     Memory: 21.5M (peak: 22.1M)
        CPU: 380ms
     CGroup: /system.slice/apache2.service
             ├─2774 /usr/sbin/apache2 -k start
             ├─2778 /usr/sbin/apache2 -k start
             ├─2779 /usr/sbin/apache2 -k start
             ├─2780 /usr/sbin/apache2 -k start
             ├─2781 /usr/sbin/apache2 -k start
             └─2782 /usr/sbin/apache2 -k start
```

Server1:



```
Apostol@Server1:~$ systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: >
     Active: active (running) since Fri 2025-11-14 05:44:35 UTC; 57min ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 1788 (apache2)
      Tasks: 6 (limit: 7600)
     Memory: 21.6M (peak: 22.0M)
        CPU: 299ms
     CGroup: /system.slice/apache2.service
             ├─1788 /usr/sbin/apache2 -k start
             ├─1844 /usr/sbin/apache2 -k start
```

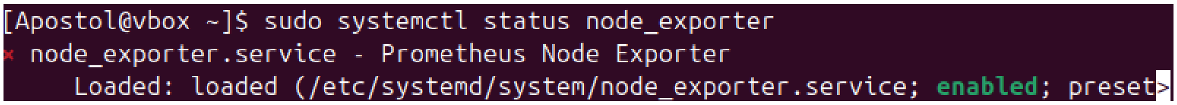
CentOS:



```
[Apostol@vbox ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: di>
     Active: active (running) since Fri 2025-11-14 13:53:18 PST; 49min ago
       Docs: man:httpd.service(8)
   Main PID: 966 (httpd)
     Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes>
      Tasks: 177 (limit: 19847)
     Memory: 18.7M (peak: 19.2M)
        CPU: 5.901s
     CGroup: /system.slice/httpd.service
```

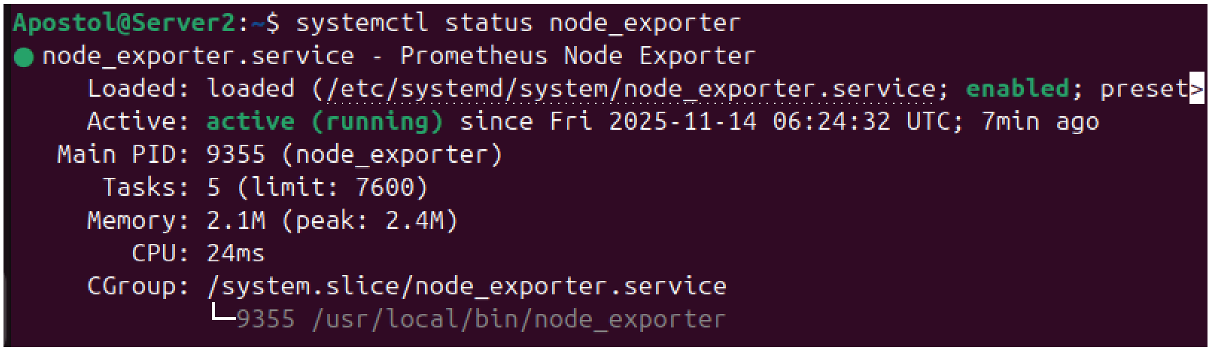
3.2 Install and configure one monitoring tool that can be installed in Debian and Centos servers (if it is a stack there should be option of different host)

CentOS:



```
[Apostol@vbox ~]$ sudo systemctl status node_exporter
× node_exporter.service - Prometheus Node Exporter
     Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; preset>
```

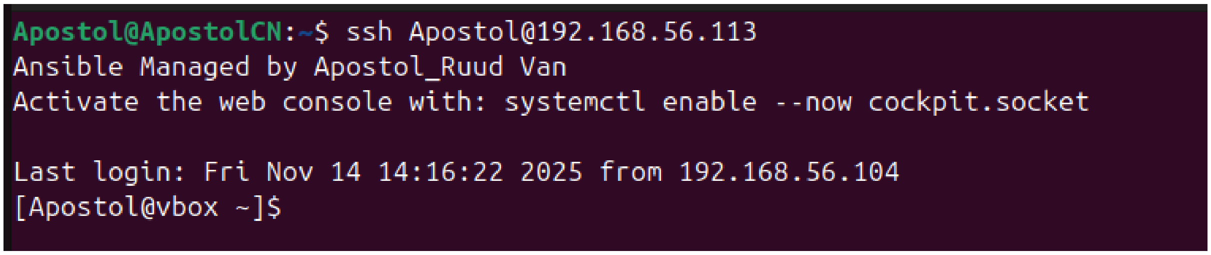
Server 2:



```
Apostol@Server2:~$ systemctl status node_exporter
● node_exporter.service - Prometheus Node Exporter
     Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; preset>
     Active: active (running) since Fri 2025-11-14 06:24:32 UTC; 7min ago
   Main PID: 9355 (node_exporter)
      Tasks: 5 (limit: 7600)
     Memory: 2.1M (peak: 2.4M)
        CPU: 24ms
     CGroup: /system.slice/node_exporter.service
             └─9355 /usr/local/bin/node_exporter
```

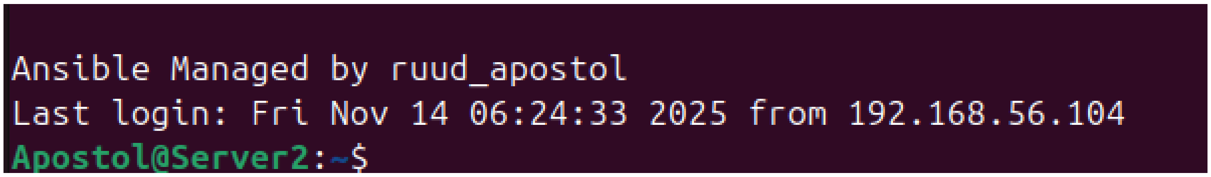
4.4 Change Motd as "Ansible Managed by <username>"

Centos:



```
Apostol@ApostolCN:~$ ssh Apostol@192.168.56.113
Ansible Managed by Apostol_Ruud Van
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Fri Nov 14 14:16:22 2025 from 192.168.56.104
[Apostol@vbox ~]$
```

Server2:

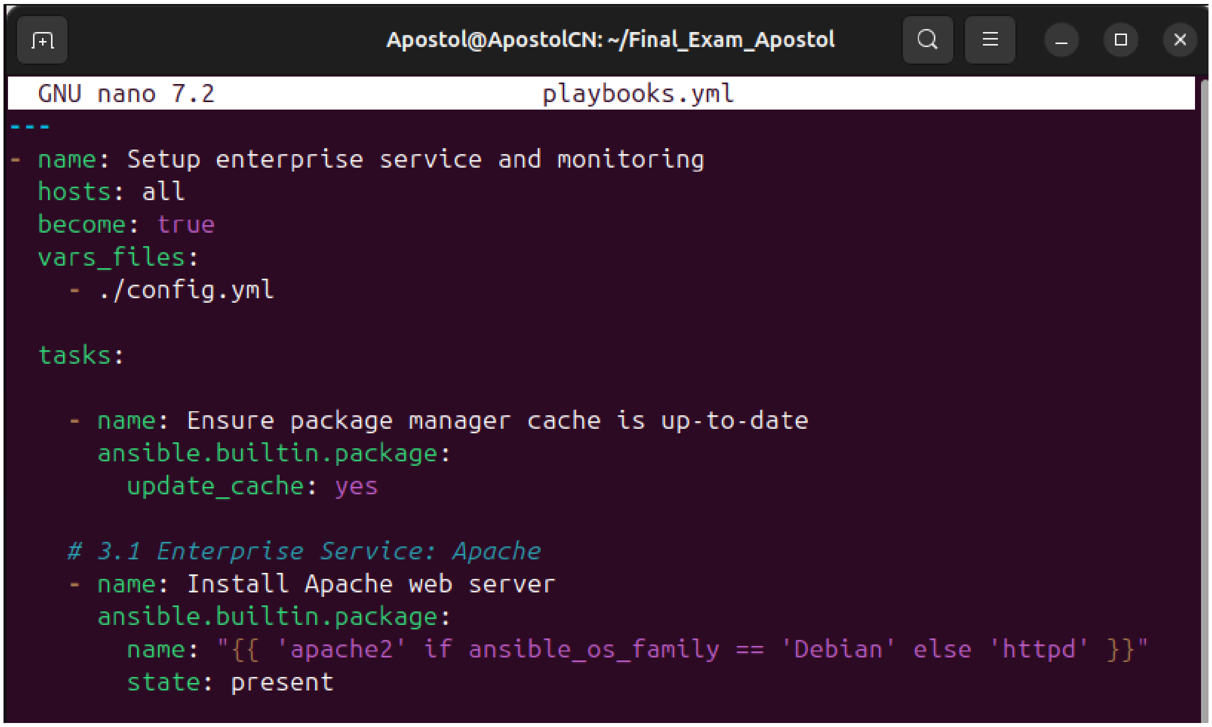


```
Ansible Managed by ruud_apostol
Last login: Fri Nov 14 06:24:33 2025 from 192.168.56.104
Apostol@Server2:~$
```

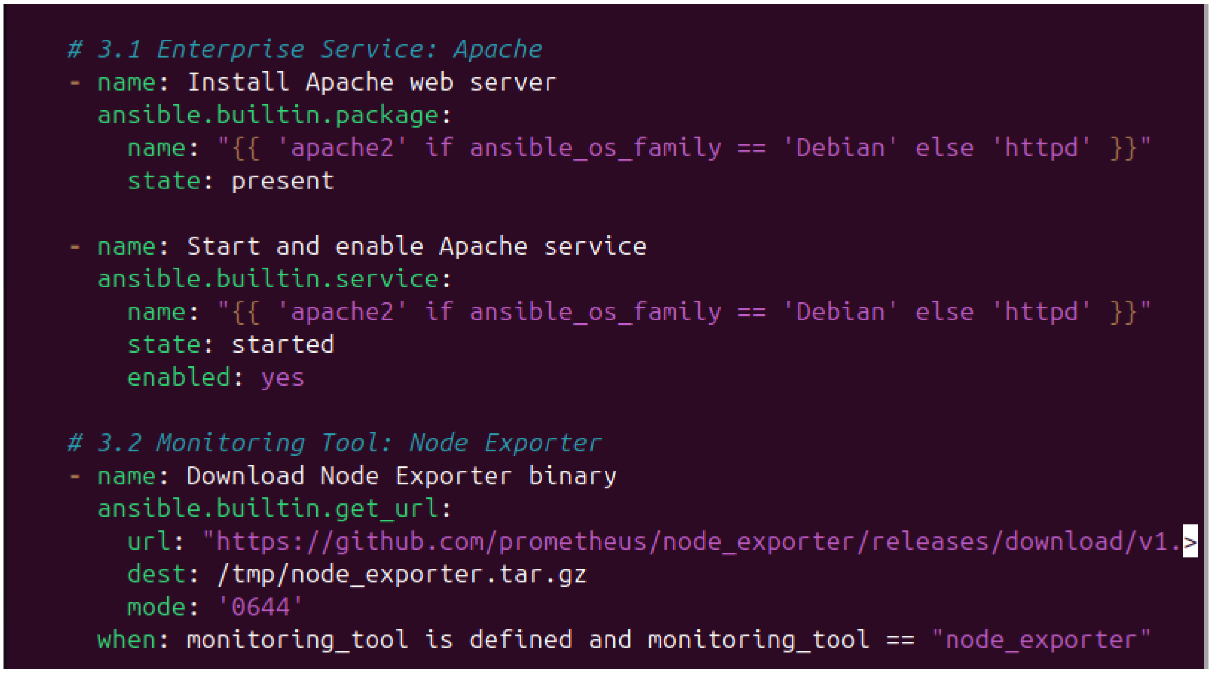
4. Push and commit your files in GitHub

5. Make sure to show evidence of input (codes) process (codes successfully running) and output (evidence of installation)

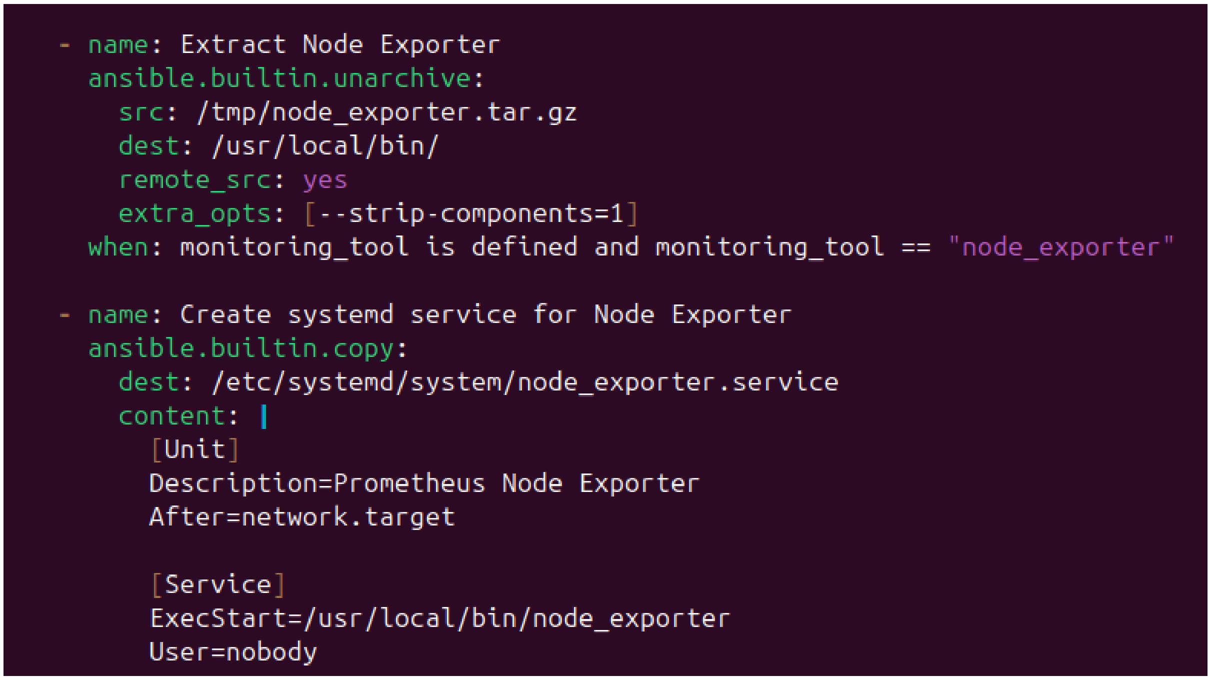
Playbooks.yml:



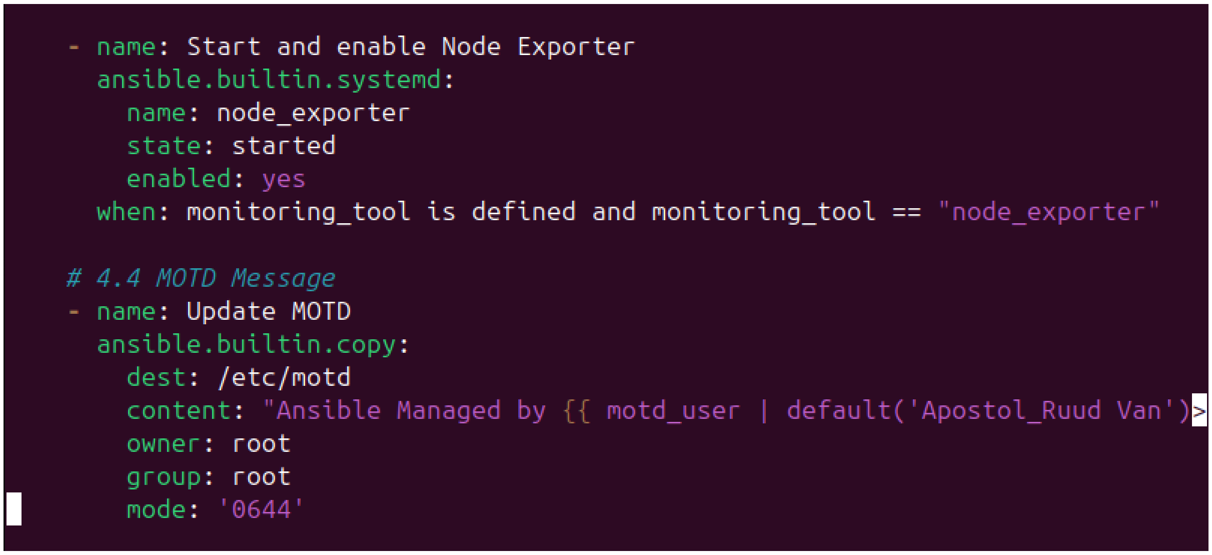
```
Apostol@ApostolCN: ~/Final_Exam_Apostol
 GNU nano 7.2                         playbooks.yml
---
- name: Setup enterprise service and monitoring
  hosts: all
  become: true
  vars_files:
    - ./config.yml

  tasks:

    - name: Ensure package manager cache is up-to-date
      ansible.builtin.package:
        update_cache: yes

    # 3.1 Enterprise Service: Apache
    - name: Install Apache web server
      ansible.builtin.package:
        name: "{{ 'apache2' if ansible_os_family == 'Debian' else 'httpd' }}"
        state: present
```



```
    # 3.1 Enterprise Service: Apache
    - name: Install Apache web server
      ansible.builtin.package:
        name: "{{ 'apache2' if ansible_os_family == 'Debian' else 'httpd' }}"
        state: present

    - name: Start and enable Apache service
      ansible.builtin.service:
        name: "{{ 'apache2' if ansible_os_family == 'Debian' else 'httpd' }}"
        state: started
        enabled: yes

    # 3.2 Monitoring Tool: Node Exporter
    - name: Download Node Exporter binary
      ansible.builtin.get_url:
        url: "https://github.com/prometheus/node_exporter/releases/download/v1.>
        dest: /tmp/node_exporter.tar.gz
        mode: '0644'
      when: monitoring_tool is defined and monitoring_tool == "node_exporter"
```

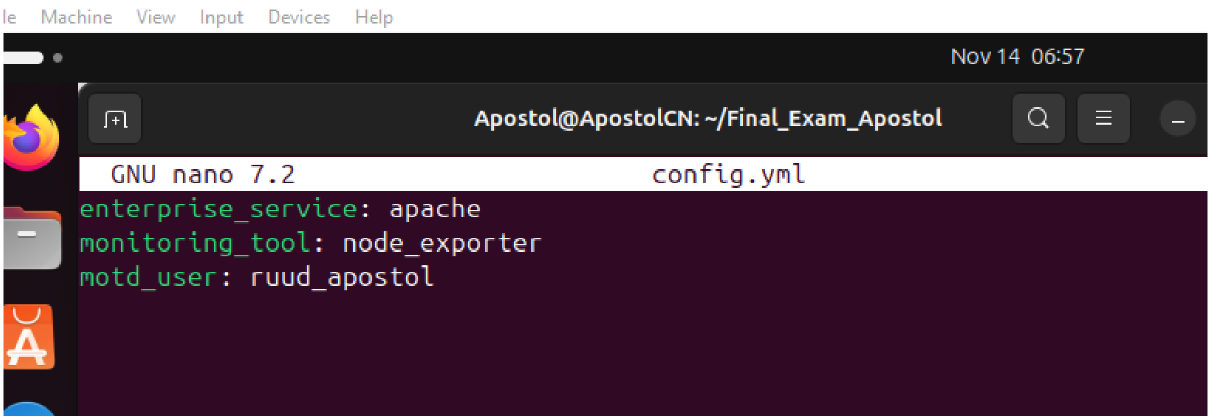


```
  - name: Extract Node Exporter
    ansible.builtin.unarchive:
      src: /tmp/node_exporter.tar.gz
      dest: /usr/local/bin/
      remote_src: yes
      extra_opts: [--strip-components=1]
    when: monitoring_tool is defined and monitoring_tool == "node_exporter"

  - name: Create systemd service for Node Exporter
    ansible.builtin.copy:
      dest: /etc/systemd/system/node_exporter.service
      content: |
        [Unit]
        Description=Prometheus Node Exporter
        After=network.target

        [Service]
        ExecStart=/usr/local/bin/node_exporter
        User=nobody
```



```
  - name: Start and enable Node Exporter
    ansible.builtin.systemd:
      name: node_exporter
      state: started
      enabled: yes
    when: monitoring_tool is defined and monitoring_tool == "node_exporter"

  # 4.4 MOTD Message
  - name: Update MOTD
    ansible.builtin.copy:
      dest: /etc/motd
      content: "Ansible Managed by {{ motd_user | default('Apostol_Ruud Van')>
      owner: root
      group: root
      mode: '0644'
```

Config.yml:


le Machine View Input Devices Help
Nov 14 06:57
Apostol@ApostolCN: ~/Final_Exam_Apostol
GNU nano 7.2
config.yml

```
enterprise_service: apache
monitoring_tool: node_exporter
motd_user: ruud_apostol
```

5. For your final exam to be counted, please paste your repository link as an answer in this exam.

Link: https://github.com/qrvgapostol/Final_Exam_Apostol

Note: Extra points if you will implement the said services via containerization.