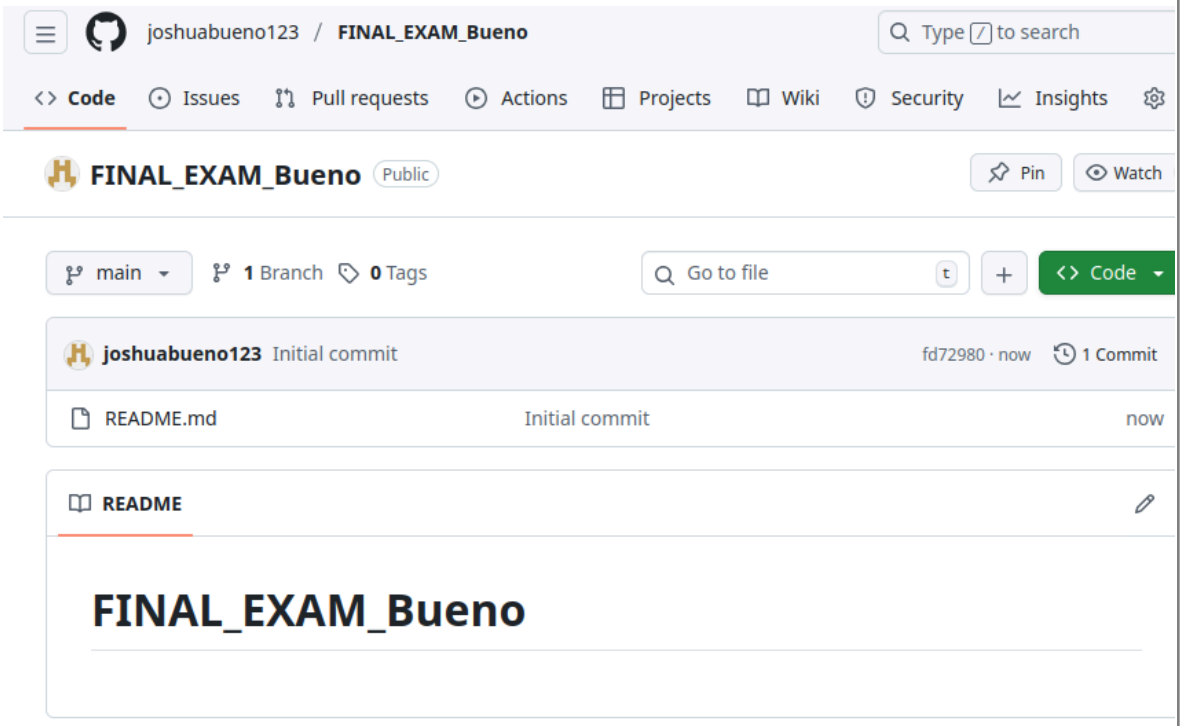
Bueno, Joshua C.	**11/14/25**
CPE31S2	**Engr. Robin Valenzuela**

Tools Needed:

1. VM with Ubuntu, CentOS and Ansible installed
2. Web browser

Procedure:

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



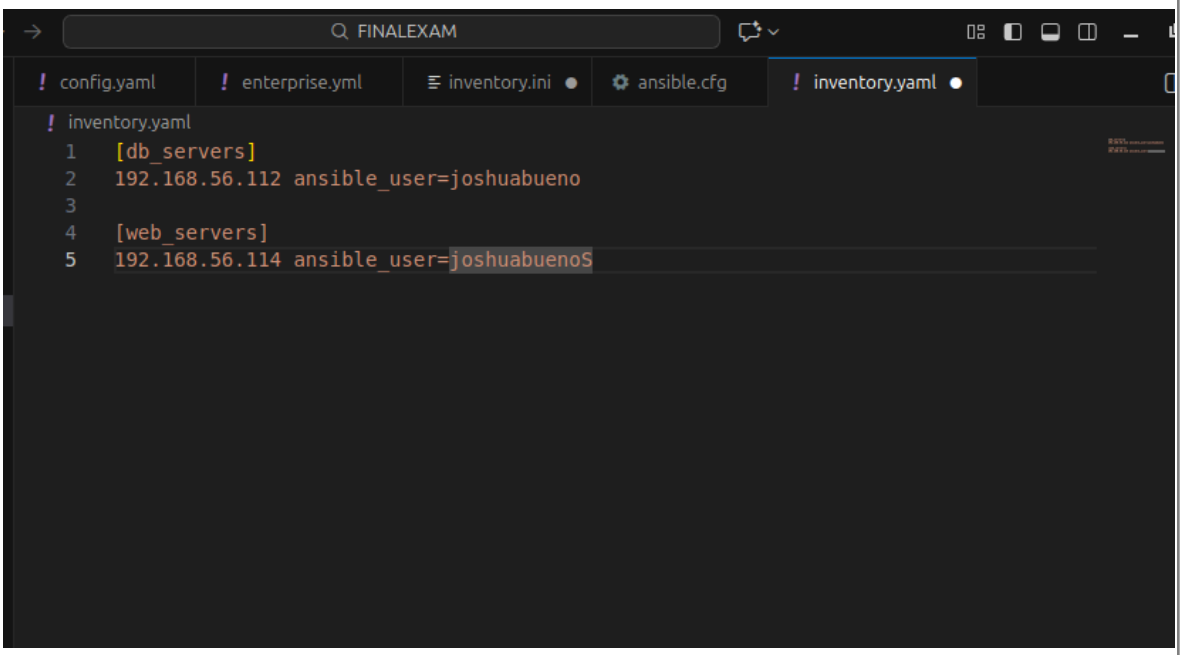
Analysis: I created a repository on my github account to put all the file that I will create in this exam

2. Clone your new repository in your VM

```
s://github.com/joshuabueno123/FINAL_EXAM_Bueno.git
Cloning into 'FINAL_EXAM_Bueno'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pac
reused 0 (from 0)
Receiving objects: 100% (3/3), done.
```

Analysis: I gitclone my repository that I created to have a backup file if I need and put it on my github account.

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



Analysis: I created all the config files and inventory that I will be using in installation like inventory, ansible, and config.yaml

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

Code:

enterprise.yml

```yaml
---
- name: Setup enterprise service and monitoring
  hosts: all
  become: true
  vars_files:
    - ./config.yaml     # make sure this file is in the same directory as playbook

  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

    - 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/v1.8.1
        dest: /tmp/node_exporter.tar.gz
        mode: '0644'
      when: monitoring_tool is defined and monitoring_tool == "node_exporter"

    - name: Extract Node Exporter
```

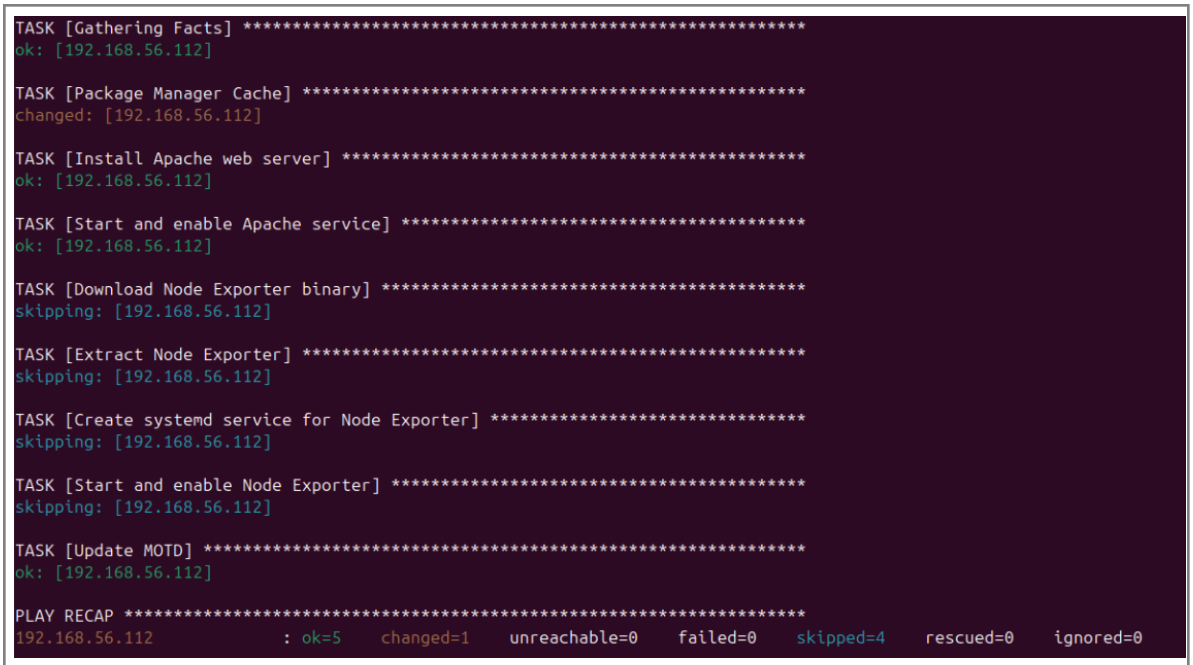
```
    ansible.builtin.unarchive:
      src: /tmp/node_exporter.tar.gz
      dest: /usr/local/bin/
      remote_src: yes
      extra_opts: [--strip-components=1]
    when: monitoring_tool is defined and monitoring_tool == "node_exporter"

  - 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
        Group=nogroup

        [Install]
        WantedBy=multi-user.target
    when: monitoring_tool is defined and monitoring_tool == "node_exporter"

  - 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('unknown_user') }}\n"
      owner: root
      group: root
      mode: '0644'
```

Analysis: This is the code that I created while installing the enterprise service in Debian and CentOS.

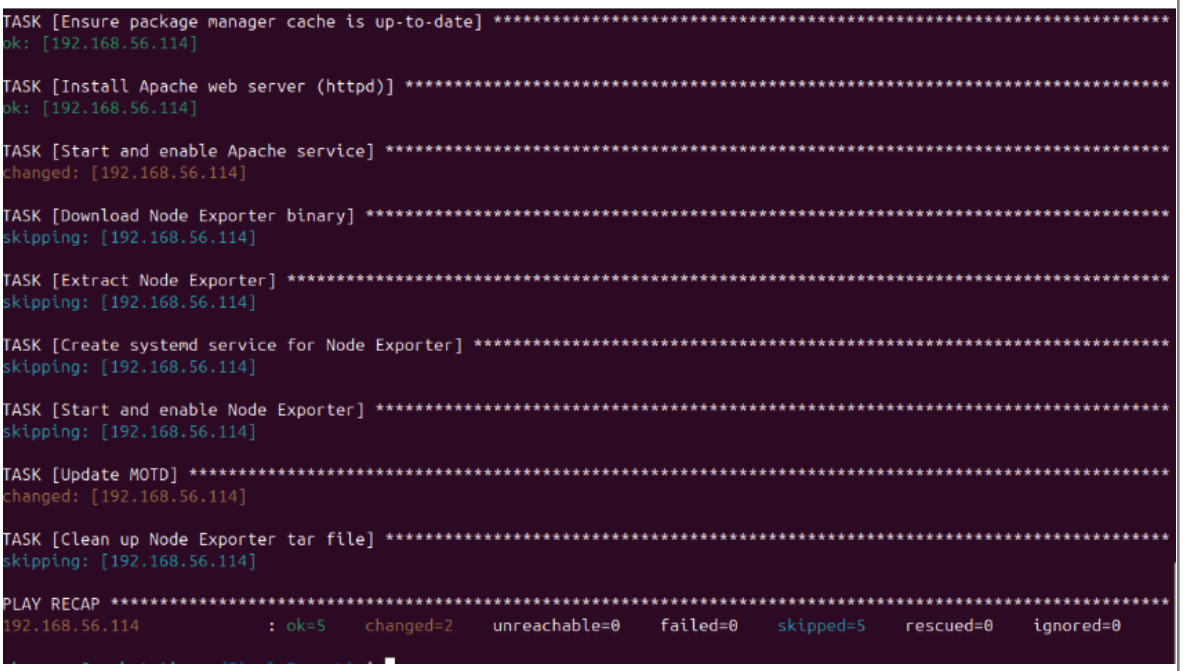
Output:

Debian



Analysis: It shows in the image that all the tasks was successfully initiated and installed in Debian server.

CENTOS:



Analysis: Based on the output, it successfully downloaded the enterprise in both Debian and CentOS

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)

Debian:

```
PLAY [Install and configure Node Exporter on Debian servers] ***********************************************************

TASK [Gathering Facts] *************************************************************************************************
ok: [192.168.56.112]

TASK [Update APT cache] ************************************************************************************************
changed: [192.168.56.112]

TASK [Install required tools on Debian] ********************************************************************************
ok: [192.168.56.112]

TASK [Download Node Exporter binary for Debian] ************************************************************************
changed: [192.168.56.112]

TASK [Extract Node Exporter] *******************************************************************************************
changed: [192.168.56.112]

TASK [Create systemd service for Node Exporter] ************************************************************************
changed: [192.168.56.112]

TASK [Reload systemd] **************************************************************************************************
ok: [192.168.56.112]

TASK [Enable and start Node Exporter service] **************************************************************************
changed: [192.168.56.112]

PLAY RECAP *************************************************************************************************************
192.168.56.112             : ok=8    changed=5    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

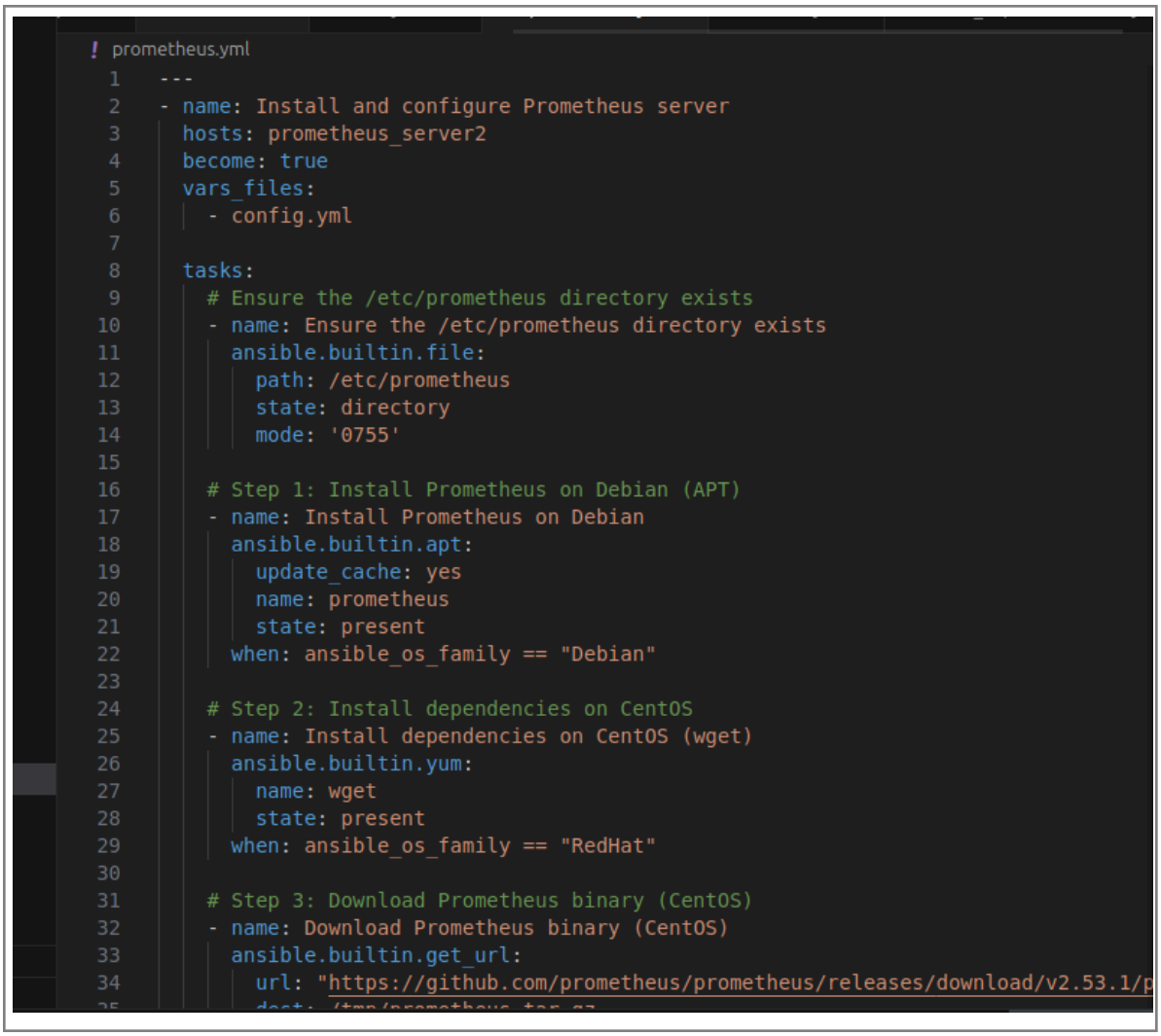
Analysis: The installation of monitoring tool was successful in Debian as the output shows all the tasks run smoothly

CENTOS:

```
BECOME password:

PLAY [Install and configure Node Exporter on CentOS servers] ***********************************************************

TASK [Gathering Facts] *************************************************************************************************
ok: [192.168.56.114]

TASK [Install required tools on CentOS] ********************************************************************************
ok: [192.168.56.114]

TASK [Download Node Exporter binary for CentOS] ************************************************************************
changed: [192.168.56.114]

TASK [Extract Node Exporter] *******************************************************************************************
changed: [192.168.56.114]

TASK [Create systemd service for Node Exporter] ************************************************************************
changed: [192.168.56.114]

TASK [Reload systemd] **************************************************************************************************
ok: [192.168.56.114]

TASK [Enable and start Node Exporter service] **************************************************************************
changed: [192.168.56.114]

PLAY RECAP *************************************************************************************************************
192.168.56.114             : ok=7    changed=4    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Analysis: The installation of monitoring tool was successful also in CentOS VM as the output shows all the tasks run smoothly

Prometheus Server
Code:



prometheus.yml

```yaml
---
- name: Install and configure Prometheus server
  hosts: prometheus_server2
  become: true
  vars_files:
    - config.yml

  tasks:
    # Ensure the /etc/prometheus directory exists
    - name: Ensure the /etc/prometheus directory exists
      ansible.builtin.file:
        path: /etc/prometheus
        state: directory
        mode: '0755'

    # Step 1: Install Prometheus on Debian (APT)
    - name: Install Prometheus on Debian
      ansible.builtin.apt:
        update_cache: yes
        name: prometheus
        state: present
      when: ansible_os_family == "Debian"

    # Step 2: Install dependencies on CentOS
    - name: Install dependencies on CentOS (wget)
      ansible.builtin.yum:
        name: wget
        state: present
      when: ansible_os_family == "RedHat"

    # Step 3: Download Prometheus binary (CentOS)
    - name: Download Prometheus binary (CentOS)
      ansible.builtin.get_url:
        url: "https://github.com/prometheus/prometheus/releases/download/v2.53.1/p
```

prometheus.yml

```
    dest: /tmp/prometheus.tar.gz
  when: ansible_os_family == "RedHat"

# Step 4: Extract Prometheus binary (CentOS)
- name: Extract Prometheus binary (CentOS)
  ansible.builtin.unarchive:
    src: /tmp/prometheus.tar.gz
    dest: /usr/local/bin/
    extra_opts: ["--strip-components=1"]
    remote_src: yes
  when: ansible_os_family == "RedHat"

# Step 5: Configure Prometheus to scrape all Node Exporters
- name: Configure Prometheus to scrape all Node Exporters
  ansible.builtin.copy:
    dest: /etc/prometheus/prometheus.yml
    content: |
      global:
        scrape_interval: 15s
      scrape_configs:
        - job_name: "node_exporter"
          static_configs:
            - targets:
              {% for host in groups['all_node_exporter'] %}
                - "{{ host }}:9100"
              {% endfor %}

# Step 6: Create Prometheus systemd service (for CentOS)
- name: Create Prometheus systemd service (CentOS)
  ansible.builtin.copy:
    dest: /etc/systemd/system/prometheus.service
    content: |
      [Unit]
      Description=Prometheus Monitoring
```

prometheus.yml

```
        Wants=network-online.target
        After=network-online.target

        [Service]
        User=nobody
        Group=nogroup
        Type=simple
        ExecStart=/usr/local/bin/prometheus \
          --config.file /etc/prometheus/prometheus.yml \
          --storage.tsdb.path /var/lib/prometheus \
          --web.console.templates=/usr/local/bin/consoles \
          --web.console.libraries=/usr/local/bin/console_libraries

        [Install]
        WantedBy=multi-user.target
      when: ansible_os_family == "RedHat"

    # Step 7: Reload systemd daemon
    - name: Reload systemd daemon
      ansible.builtin.systemd:
        daemon_reload: yes

    # Step 8: Enable and start Prometheus service
    - name: Enable and start Prometheus service
      ansible.builtin.systemd:
        name: prometheus
        state: started
        enabled: yes

    # Step 9: Update MOTD
    - name: Get current user for MOTD
      ansible.builtin.command:
        cmd: whoami
      register: current_user
```

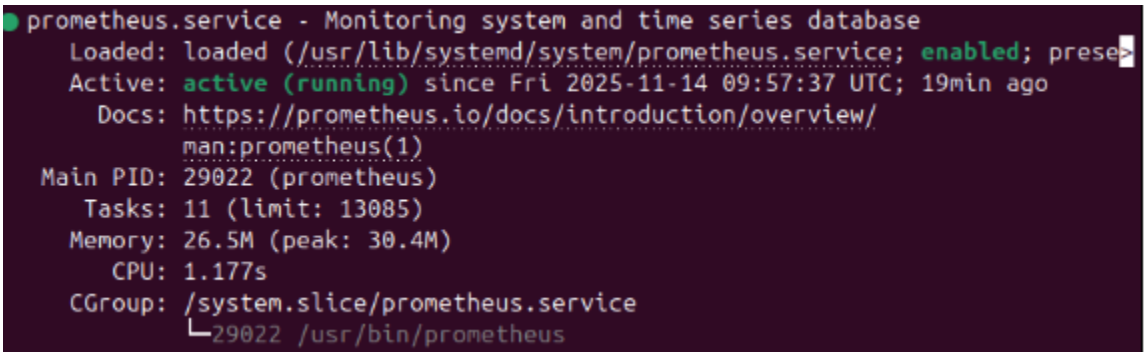
prometheus.yml

```
69            Wants=network-online.target
70            After=network-online.target
71
72            [Service]
73            User=nobody
74            Group=nogroup
75            Type=simple
76            ExecStart=/usr/local/bin/prometheus \
77              --config.file /etc/prometheus/prometheus.yml \
78              --storage.tsdb.path /var/lib/prometheus \
79              --web.console.templates=/usr/local/bin/consoles \
80              --web.console.libraries=/usr/local/bin/console_libraries
81
82            [Install]
83            WantedBy=multi-user.target
84        when: ansible_os_family == "RedHat"
85
86    # Step 7: Reload systemd daemon
87    - name: Reload systemd daemon
88      ansible.builtin.systemd:
89        daemon_reload: yes
90
91    # Step 8: Enable and start Prometheus service
92    - name: Enable and start Prometheus service
93      ansible.builtin.systemd:
94        name: prometheus
95        state: started
96        enabled: yes
97
98    # Step 9: Update MOTD
99    - name: Get current user for MOTD
100     ansible.builtin.command:
101       cmd: whoami
102     register: current_user
103
104   - name: Update MOTD with user info
105     ansible.builtin.copy:
106       dest: /etc/motd
107       content: |
108         Ansible Managed by {{ current_user.stdout }}
109       owner: root
110       group: root
111       mode: '0644'S
```

Analysis: this is the code for the prometheus server

Debian:

```
'all'

PLAY [Install and configure Prometheus server] *********************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.112]

TASK [Install Prometheus on Debian] ********************************************
ok: [192.168.56.112]

TASK [Install dependencies on CentOS] ******************************************
skipping: [192.168.56.112]

TASK [Download Prometheus binary (CentOS)] *************************************
skipping: [192.168.56.112]

TASK [Extract Prometheus (CentOS)] *********************************************
skipping: [192.168.56.112]

TASK [Configure Prometheus to scrape all Node Exporters] ***********************
changed: [192.168.56.112]

TASK [Enable and start Prometheus] *********************************************
ok: [192.168.56.112]

PLAY RECAP *********************************************************************
192.168.56.112             : ok=4    changed=1    unreachable=0    failed=0    skipped=3    rescued=0    ignored=0
```



Analysis: This is the Debian VM, and it is also successful and it said “active”

CENTOS:

```
TASK [Install dependencies on CentOS (wget)] ***********************************************************************
ok: [192.168.56.114]

TASK [Download Prometheus binary (CentOS)] *************************************************************************
ok: [192.168.56.114]

TASK [Extract Prometheus binary (CentOS)] **************************************************************************
ok: [192.168.56.114]

TASK [Configure Prometheus to scrape all Node Exporters] ***********************************************************
changed: [192.168.56.114]

TASK [Create Prometheus systemd service (CentOS)] ******************************************************************
changed: [192.168.56.114]

TASK [Reload systemd daemon] ***************************************************************************************
ok: [192.168.56.114]

TASK [Enable and start Prometheus service] *************************************************************************
changed: [192.168.56.114]

TASK [Get current user for MOTD] ***********************************************************************************
changed: [192.168.56.114]

TASK [Update MOTD with user info] **********************************************************************************
changed: [192.168.56.114]

PLAY RECAP *********************************************************************************************************
192.168.56.114             : ok=11   changed=5    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```

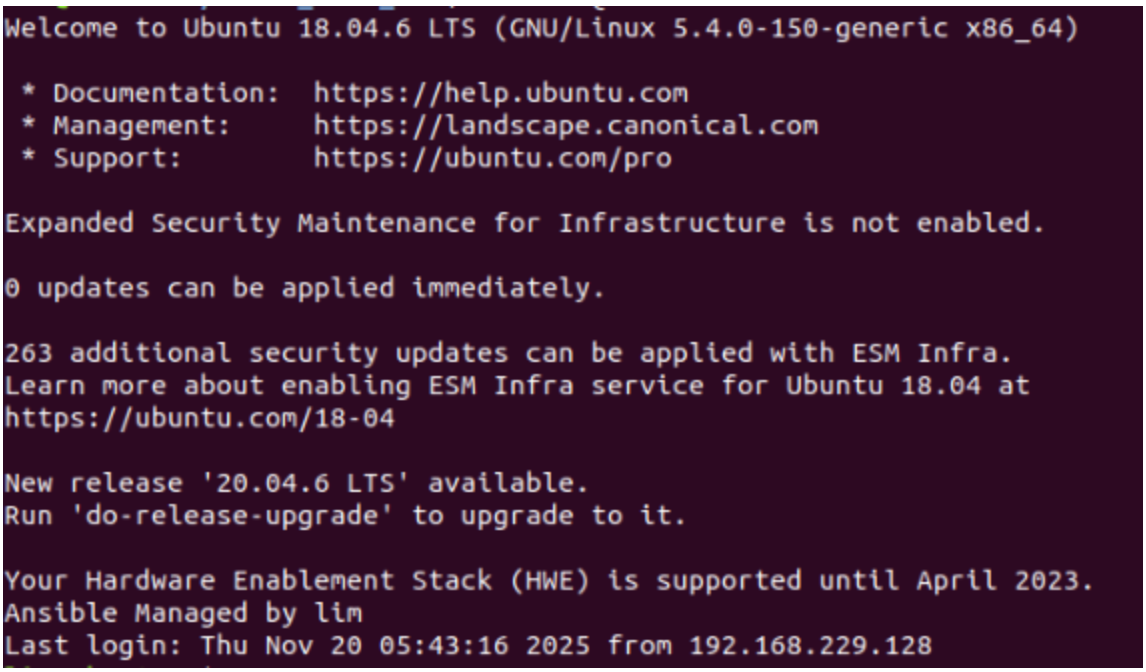
```
● node_exporter.service - Prometheus Node Exporter
     Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; preset
     Active: active (running) since Fri 2025-11-14 17:48:06 PST; 38min ago
   Main PID: 6531 (node_exporter)
      Tasks: 6 (limit: 23007)
     Memory: 5.0M
        CPU: 132ms
     CGroup: /system.slice/node_exporter.service
             └─6531 /usr/local/bin/node_exporter

Nov 14 17:48:06 vbox node_exporter[6531]: ts=2025-11-14T09:48:06.155Z caller=no
```

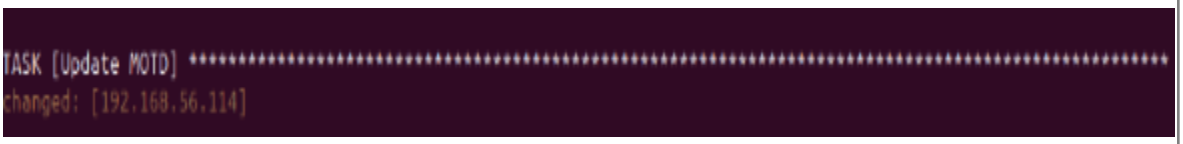
Analysis: It says active so the tasks and installation was successful in CentOS VM

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

Code:

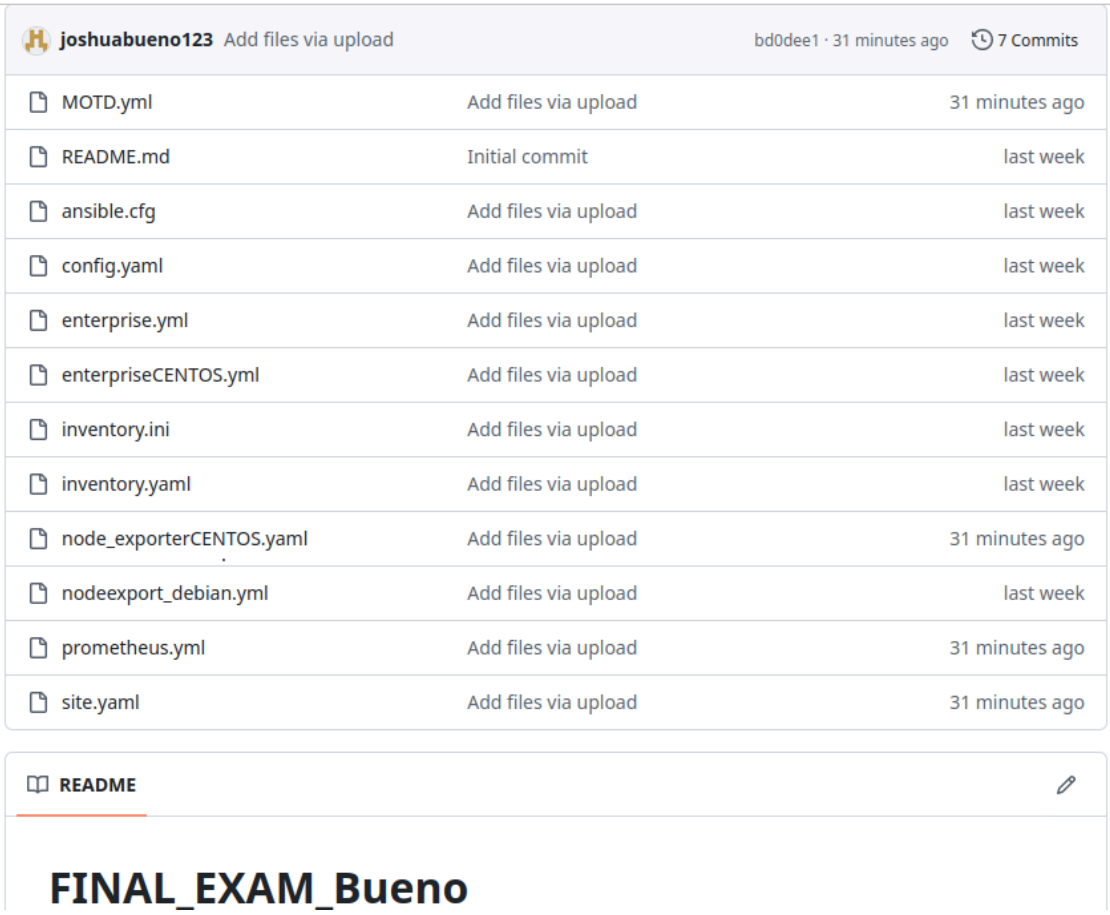
```
65
66      # 4.4 MOTD Message
67      - name: Update MOTD
68        ansible.builtin.copy:
69          dest: /etc/motd
70          content: "Ansible Managed by {{ motd_user | default('unknown_user') }}\n"
71          owner: root
72          group: root
73          mode: '0644'
74
```





Analysis: I successfully change the Motd as the “ansible managed” by my username. Based on the output the task updated the MOTD successfully.

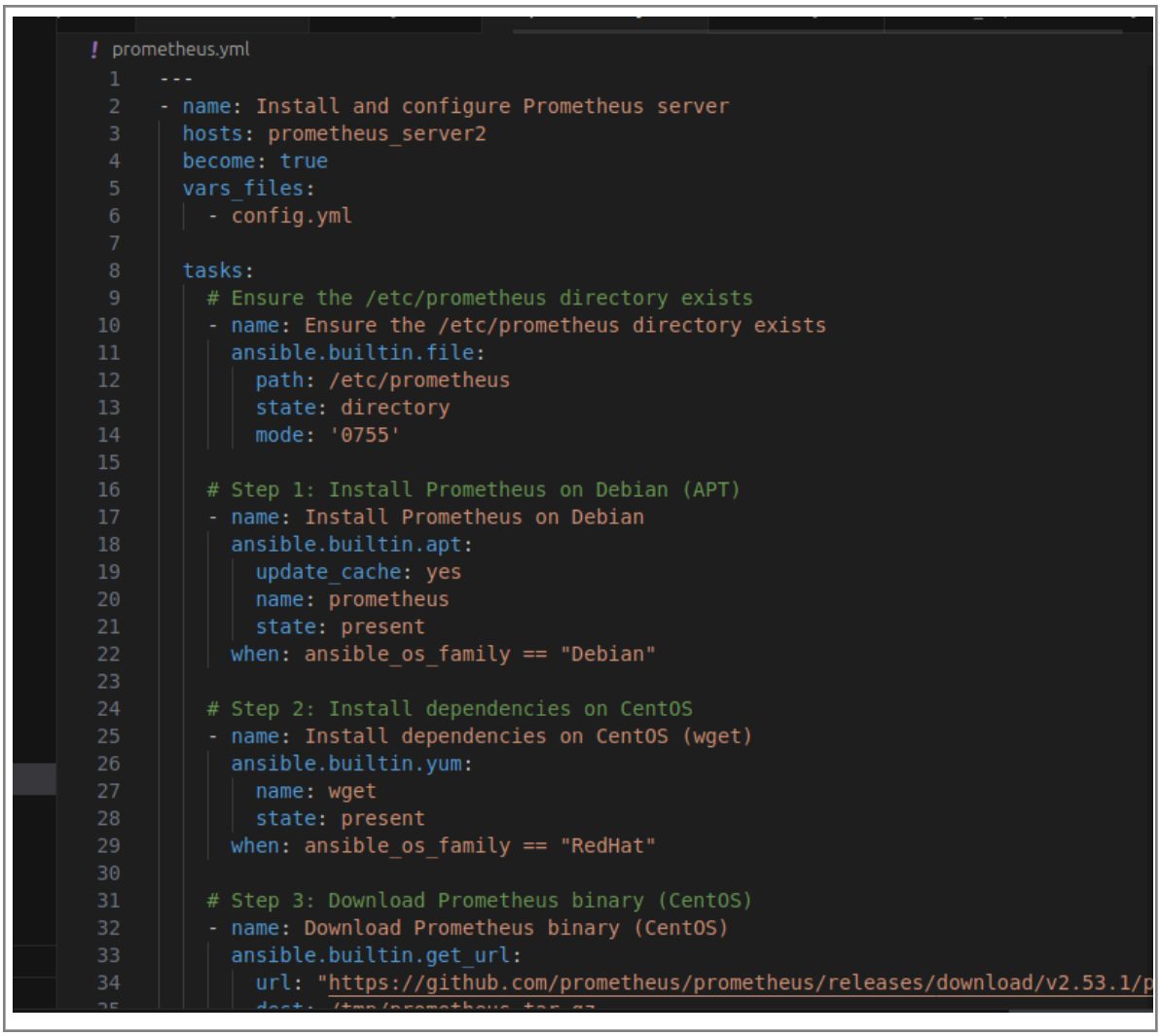
4. Push and commit your files in GitHub



Analysis: I successfully push and commit all the files that I used in the exam in my github account

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

Code:



prometheus.yml

```yaml
---
- name: Install and configure Prometheus server
  hosts: prometheus_server2
  become: true
  vars_files:
    - config.yml

  tasks:
    # Ensure the /etc/prometheus directory exists
    - name: Ensure the /etc/prometheus directory exists
      ansible.builtin.file:
        path: /etc/prometheus
        state: directory
        mode: '0755'

    # Step 1: Install Prometheus on Debian (APT)
    - name: Install Prometheus on Debian
      ansible.builtin.apt:
        update_cache: yes
        name: prometheus
        state: present
      when: ansible_os_family == "Debian"

    # Step 2: Install dependencies on CentOS
    - name: Install dependencies on CentOS (wget)
      ansible.builtin.yum:
        name: wget
        state: present
      when: ansible_os_family == "RedHat"

    # Step 3: Download Prometheus binary (CentOS)
    - name: Download Prometheus binary (CentOS)
      ansible.builtin.get_url:
        url: "https://github.com/prometheus/prometheus/releases/download/v2.53.1/p
```

prometheus.yml

```
    dest: /tmp/prometheus.tar.gz
  when: ansible_os_family == "RedHat"

# Step 4: Extract Prometheus binary (CentOS)
- name: Extract Prometheus binary (CentOS)
  ansible.builtin.unarchive:
    src: /tmp/prometheus.tar.gz
    dest: /usr/local/bin/
    extra_opts: ["--strip-components=1"]
    remote_src: yes
  when: ansible_os_family == "RedHat"

# Step 5: Configure Prometheus to scrape all Node Exporters
- name: Configure Prometheus to scrape all Node Exporters
  ansible.builtin.copy:
    dest: /etc/prometheus/prometheus.yml
    content: |
      global:
        scrape_interval: 15s
      scrape_configs:
        - job_name: "node_exporter"
          static_configs:
            - targets:
              {% for host in groups['all_node_exporter'] %}
                - "{{ host }}:9100"
              {% endfor %}

# Step 6: Create Prometheus systemd service (for CentOS)
- name: Create Prometheus systemd service (CentOS)
  ansible.builtin.copy:
    dest: /etc/systemd/system/prometheus.service
    content: |
      [Unit]
      Description=Prometheus Monitoring
```

prometheus.yml

```
        Wants=network-online.target
        After=network-online.target

        [Service]
        User=nobody
        Group=nogroup
        Type=simple
        ExecStart=/usr/local/bin/prometheus \
          --config.file /etc/prometheus/prometheus.yml \
          --storage.tsdb.path /var/lib/prometheus \
          --web.console.templates=/usr/local/bin/consoles \
          --web.console.libraries=/usr/local/bin/console_libraries

        [Install]
        WantedBy=multi-user.target
      when: ansible_os_family == "RedHat"

    # Step 7: Reload systemd daemon
    - name: Reload systemd daemon
      ansible.builtin.systemd:
        daemon_reload: yes

    # Step 8: Enable and start Prometheus service
    - name: Enable and start Prometheus service
      ansible.builtin.systemd:
        name: prometheus
        state: started
        enabled: yes

    # Step 9: Update MOTD
    - name: Get current user for MOTD
      ansible.builtin.command:
        cmd: whoami
      register: current_user
```

prometheus.yml

```
69            Wants=network-online.target
70            After=network-online.target
71
72            [Service]
73            User=nobody
74            Group=nogroup
75            Type=simple
76            ExecStart=/usr/local/bin/prometheus \
77              --config.file /etc/prometheus/prometheus.yml \
78              --storage.tsdb.path /var/lib/prometheus \
79              --web.console.templates=/usr/local/bin/consoles \
80              --web.console.libraries=/usr/local/bin/console_libraries
81
82            [Install]
83            WantedBy=multi-user.target
84        when: ansible_os_family == "RedHat"
85
86    # Step 7: Reload systemd daemon
87    - name: Reload systemd daemon
88      ansible.builtin.systemd:
89        daemon_reload: yes
90
91    # Step 8: Enable and start Prometheus service
92    - name: Enable and start Prometheus service
93      ansible.builtin.systemd:
94        name: prometheus
95        state: started
96        enabled: yes
97
98    # Step 9: Update MOTD
99    - name: Get current user for MOTD
100     ansible.builtin.command:
101       cmd: whoami
102     register: current_user
```

```
103
104   - name: Update MOTD with user info
105     ansible.builtin.copy:
106       dest: /etc/motd
107       content: |
108         Ansible Managed by {{ current_user.stdout }}
109       owner: root
110       group: root
111       mode: '0644'S
```

Analysis: This is the code in the installation of the tasks, the prometheus, node exporter, motd.

Process:

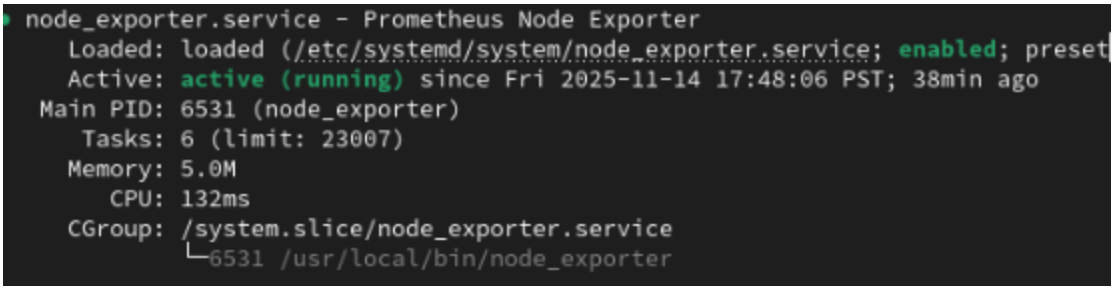
```
'all'

PLAY [Install and configure Prometheus server] *********************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [192.168.56.112]

TASK [Install Prometheus on Debian] ********************************************************************************
ok: [192.168.56.112]

TASK [Install dependencies on CentOS] ******************************************************************************
skipping: [192.168.56.112]

TASK [Download Prometheus binary (CentOS)] *************************************************************************
skipping: [192.168.56.112]

TASK [Extract Prometheus (CentOS)] *********************************************************************************
skipping: [192.168.56.112]

TASK [Configure Prometheus to scrape all Node Exporters] ***********************************************************
changed: [192.168.56.112]

TASK [Enable and start Prometheus] *********************************************************************************
ok: [192.168.56.112]

PLAY RECAP *********************************************************************************************************
192.168.56.112             : ok=4    changed=1    unreachable=0    failed=0    skipped=3    rescued=0    ignored=0
```

Analysis: This is the process of installing all the required tasks in Debian

Output:

```
TASK [Install dependencies on CentOS (wget)] ***********************************************************************
ok: [192.168.56.114]

TASK [Download Prometheus binary (CentOS)] *************************************************************************
ok: [192.168.56.114]

TASK [Extract Prometheus binary (CentOS)] **************************************************************************
ok: [192.168.56.114]

TASK [Configure Prometheus to scrape all Node Exporters] ***********************************************************
changed: [192.168.56.114]

TASK [Create Prometheus systemd service (CentOS)] ******************************************************************
changed: [192.168.56.114]

TASK [Reload systemd daemon] ***************************************************************************************
ok: [192.168.56.114]

TASK [Enable and start Prometheus service] *************************************************************************
changed: [192.168.56.114]

TASK [Get current user for MOTD] ***********************************************************************************
changed: [192.168.56.114]

TASK [Update MOTD with user info] **********************************************************************************
changed: [192.168.56.114]

PLAY RECAP *********************************************************************************************************
192.168.56.114             : ok=11   changed=5    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```

Analysis: As you can see in the output, all the installation was successful



Analysis: It says active also, the installation was successful also in CentOS

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

Github Link:
https://github.com/joshuabueno123/FINAL_EXAM_Bueno.git

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