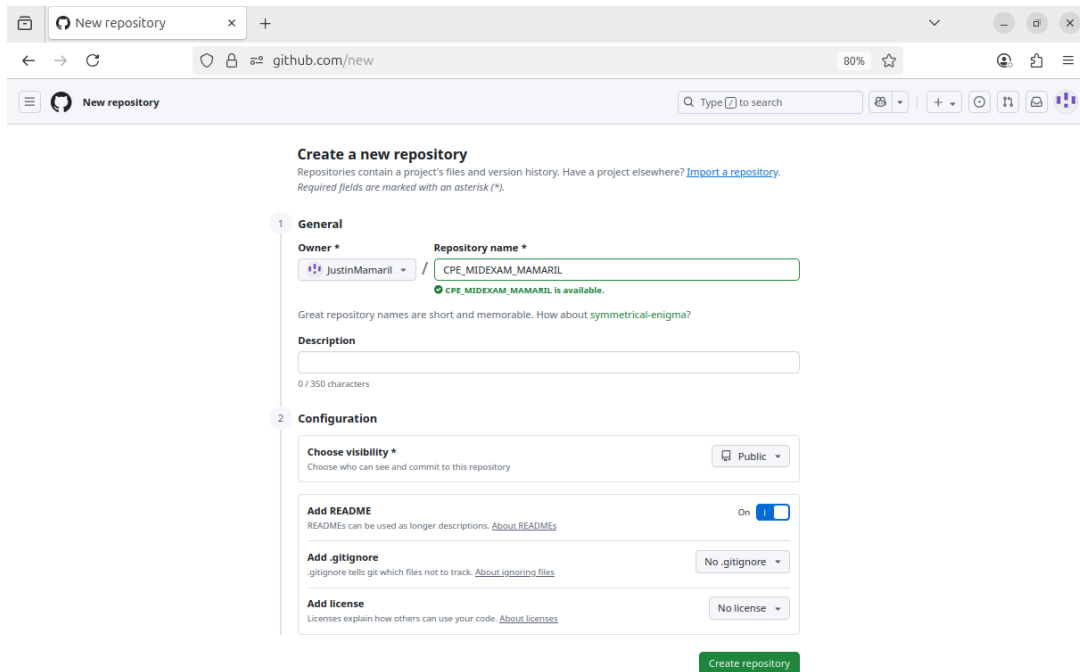| Name: Justin Kenneth I. Mamaril	Date Performed: 10/10/2025
Course/Section: CPE232-CPE31S	Date Submitted: 10/10/2025
Instructor:	Semester and SY: 1st Sem 25-26

Midterm Skills Exam: Install, Configure, and Manage Log Monitoring tools

1. Objectives

Create and design a workflow that installs, configure and manage enterprise availability, performance and log monitoring tools using Ansible as an Infrastructure as Code (IaC) tool.

2. Instructions

1. Create a repository in your GitHub account and label it CPE_MIDEXAM_SURNAME.
2. Clone the repository and do the following:
    2.1. Create an Ansible playbook that does the following with an input of a config.yaml file and arranged Inventory file:
    2.2. Install and configure Elastic Stack in separate hosts (Elastic Search, Kibana, Logstash) • Install Nagios in one host
    2.3. Install Grafana,Prometheus and Influxdb in seperate hosts (Influxdb,Grafana,Prometheus)
    2.4. Install Lamp Stack in separate hosts (Httpd + Php,Mariadb)
3. Document all your tasks using this document. Provide proofs of all the ansible playbooks codes and successful installations.
4. Document the push and commit from the local repository to GitHub.
5. Finally, paste also the link of your GitHub repository in the documentation.

3. **Output** (screenshots and explanations)

1.
- I created a github repository named CPE_MIDEXAM_MAMARIL and added a README file

```
pc1@workstation:~/CPE_MIDEXAM_MAMARIL$ git clone git@github.com:JustinMamaril/CP
E_MIDEXAM_MAMARIL.git
Cloning into 'CPE_MIDEXAM_MAMARIL'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.
```

2.
- i cloned the repository and the directory

```
pc1@workstation:~/CPE_MIDEXAM_MAMARIL$ sudo nano ansible.cfg
```

3.
- i created the ansible.cfg file which include this code

```
  GNU nano 7.2                              ansible.cfg
[defaults]
inventory=inventory.ini
private_key_file=~/.ssh/ansible
```

```
pc1@workstation:~/CPE_MIDEXAM_MAMARIL$ mkdir roles
pc1@workstation:~/CPE_MIDEXAM_MAMARIL$ cd roles
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles$ mkdir elasticstack
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles$ mkdir nagios
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles$ mkdir lampstack
```
4.
- Then i created the roles with their each respective directory

```
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles$ cd elasticstack
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles/elasticstack$ mkdir tasks
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles/elasticstack$ cd tasks
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles/elasticstack/tasks$ sudo nano main.y
aml
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles/elasticstack/tasks$ cd ..
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles/elasticstack$ cd ..
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles$ cd nagios
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles/nagios$ mkdir tasks
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles/nagios$ cd ..
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles$ cd lampstack
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles/lampstack$ mkdir tasks
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles/lampstack$ cd ..
```
5.

- Then I made each role inside have their own tasks directory.

```
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles/elasticstack/tasks$ cp main.yaml ~/C
PE_MIDEXAM_MAMARIL/roles/nagios
pc1@workstation:~/CPE_MIDEXAM_MAMARIL/roles/elasticstack/tasks$ cp main.yaml ~/C
PE_MIDEXAM_MAMARIL/roles/lampstack
```

- Then I just copied the main.yaml that i made in the elasticsearch to make it
  easier.

```
GNU nano 7.2                          inventory.ini *
[all]
192.168.56.106
192.168.56.107
192.168.56.109 ansible_user=mamarilcentos
```

6.

- I made the inventory.ini and then input all of my nodes

pc1@workstation:~/CPE_MIDEXAM_MAMARIL$ sudo nano site.yaml

```
GNU nano 7.2                          site.yaml
---
hosts: all
  become: true
  pre_tasks:

  - name: update repository index (CentOS)
    tags: always
    dnf:
      update_cache: yes
    changed_when: false
    when: ansible_distribution == "CentOS"

  - name: install updates (Ubuntu)
    tags: always
    apt:
      update_cache: yes
    changed_when: false
    when: ansible_distribution == "Ubuntu"

hosts: all
  become: true
  roles:
    - elasticstack

hosts: all
  become: true
  roles:
    - nagios
```

7.

```
hosts: all
  become: true
  roles:
    - lampstack
```

- Then I created the site.yaml or config yaml to call each role.

8. Proceeding to the input of codes in main.yaml in respective roles, ill just provide screenshots.

elasticstack

nagios

```yaml
GNU nano 7.2                          main.yaml
---
name: Install Nagios (Ubuntu)
  apt:
    name:
      - nagios4
    state: present
    update_cache: yes
  when: ansible_distribution == "Ubuntu"

name: Enable EPEL repo (CentOS/RHEL)
  yum:
    name: epel-release
    state: present
  when: ansible_distribution == "CentOS"

name: Install Nagios core and essential plugins (CentOS/RHEL)
  yum:
    name:
      - nagios
      - nagios-plugins
      - nagios-plugins-disk
      - nagios-plugins-http
      - nagios-plugins-load
      - nagios-plugins-ping
      - nagios-plugins-procs
      - nagios-plugins-users
    state: present
```

```yaml
    state: present
  when: ansible_distribution == "CentOS"

name: Ensure Nagios service is started and enabled (Ubuntu)
  service:
    name: nagios4
    state: started
    enabled: true
  when: ansible_distribution == "Ubuntu"

name: Ensure Nagios service is started and enabled (CentOS/RHEL)
  service:
    name: nagios
    state: started
    enabled: true
  when: ansible_distribution == "CentOS"
```

lampstack

```
GNU nano 7.2                          main.yaml
---
- name: Install Apache and PHP
  apt:
    name:
      - apache2
      - php
    state: present
    update_cache: yes

- name: Install MariaDB Server
  apt:
    name: mariadb-server
    state: present
    update_cache: yes
```

```
pc1@workstation:~/CPE_MIDEXAM_MAMARIL$ ansible-playbook site.yaml -K
BECOME password:

PLAY [all] *********************************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.109]
ok: [192.168.56.106]
ok: [192.168.56.107]

TASK [update repository index (CentOS)] ****************************************
skipping: [192.168.56.106]
skipping: [192.168.56.107]
ok: [192.168.56.109]

TASK [install updates (Ubuntu)] ************************************************
skipping: [192.168.56.109]
ok: [192.168.56.106]
ok: [192.168.56.107]

PLAY [all] *********************************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.109]
ok: [192.168.56.106]
ok: [192.168.56.107]

TASK [elasticstack : add elasticstack prerequisites (Ubuntu)] ******************
skipping: [192.168.56.109]
ok: [192.168.56.107]
ok: [192.168.56.106]
```

9.

```
ok: [192.168.56.109]

TASK [elasticstack : add elastickstack GPG key (Ubuntu)] ********************************************
skipping: [192.168.56.109]
ok: [192.168.56.107]
ok: [192.168.56.106]

TASK [elasticstack : import elastickstack repository (Ubuntu)] **************************************
skipping: [192.168.56.109]
fatal: [192.168.56.106]: FAILED! => {"changed": false, "msg": "Failed to update apt cache: W:Updating from such a reposi
tory can't be done securely, and is therefore disabled by default., W:See apt-secure(8) manpage for repository creation
and user configuration details., E:The repository 'https://artifacts.elastic.co/packages/8.14.1/apt stable Release' does
 not have a Release file."}
fatal: [192.168.56.107]: FAILED! => {"changed": false, "msg": "Failed to update apt cache: W:Updating from such a reposi
tory can't be done securely, and is therefore disabled by default., W:See apt-secure(8) manpage for repository creation
and user configuration details., E:The repository 'https://artifacts.elastic.co/packages/8.14.1/apt stable Release' does
 not have a Release file."}

TASK [elasticstack : Add Elasticsearch Yum Repository] *********************************************
ok: [192.168.56.109]

TASK [elasticstack : install elasticstack for Ubuntu (Elasticsearch, Kibana, Logstash)] ************
skipping: [192.168.56.109]

TASK [elasticstack : install elasticstack for CentOS (Elasticsearch, Kibana, Logstash)] ************
skipping: [192.168.56.109]

PLAY [all] ****************************************************************************************

TASK [Gathering Facts] ***************************************************************************
ok: [192.168.56.109]
```

```
TASK [Gathering Facts] ***************************************************************************
ok: [192.168.56.109]

TASK [nagios : Install Nagios (Ubuntu)] ********************************************************
skipping: [192.168.56.109]

TASK [nagios : Enable EPEL repo (CentOS/RHEL)] ************************************************
ok: [192.168.56.109]

TASK [nagios : Install Nagios core and essential plugins (CentOS/RHEL)] ***********************
ok: [192.168.56.109]

TASK [nagios : Ensure Nagios service is started and enabled (Ubuntu)] *************************
skipping: [192.168.56.109]

TASK [nagios : Ensure Nagios service is started and enabled (CentOS/RHEL)] ********************
ok: [192.168.56.109]

PLAY [all] ***********************************************************************************

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.109]

TASK [lampstack : Install apache and php for Ubuntu Servers] ******************************
skipping: [192.168.56.109]

TASK [lampstack : Install apache and php for CentOS servers] ******************************
ok: [192.168.56.109]

TASK [lampstack : Install MariaDB package for CentOS] ************************************
```

```
PLAY [all] **************************************************************************************

TASK [Gathering Facts] ***********************************************************************
ok: [192.168.56.109]

TASK [lampstack : Install apache and php for Ubuntu Servers] ****************************
skipping: [192.168.56.109]

TASK [lampstack : Install apache and php for CentOS servers] ****************************
ok: [192.168.56.109]

TASK [lampstack : Install MariaDB package for CentOS] **********************************
```

```
PLAY [all] ***********************************************************************************

TASK [Gathering Facts] **********************************************************************
ok: [192.168.56.109]

TASK [lampstack : Install apache and php for Ubuntu Servers] ********************************
skipping: [192.168.56.109]

TASK [lampstack : Install apache and php for CentOS servers] ********************************
ok: [192.168.56.109]

TASK [lampstack : Install MariaDB package for CentOS] **************************************
changed: [192.168.56.109]

TASK [lampstack : Start enable mariadb] ***************************************************
skipping: [192.168.56.109]

TASK [lampstack : Install MariaDB package for CentOS] **************************************
skipping: [192.168.56.109]

TASK [lampstack : Install MariaDB package for CentOS] **************************************
ok: [192.168.56.109]

PLAY RECAP **********************************************************************************
192.168.56.106             : ok=5    changed=0    unreachable=0    failed=1    skipped=2    rescued=0    ignored=0
192.168.56.107             : ok=5    changed=0    unreachable=0    failed=1    skipped=2    rescued=0    ignored=0
192.168.56.109             : ok=13   changed=1    unreachable=0    failed=0    skipped=11   rescued=0    ignored=0

bc1@workstation: ~/CPE_MIDEXAM_MAMARIL $
```

- I runned the code and it successfully showed that it runned one error

GitHub link:

https://github.com/JustinMamaril/CPE_MIDEXAM_MAMARIL.git

Conclusions: (link your conclusion from the objective)
- **I concluded that I successfully created a workflow that configure and download certain software using ansible and properly separate them with the use of roles and ansible playbook. I also successfully deploy everything automatically and make sure the configurations are exactly the same across all the servers with the use of roles.**