Motivation

It was discussed during the lectures that for a final task can be taken any Java project ...

As I didn't have any experience with Java web application I decided to explore it in practice.

The final project includes:

- Java Project (webapp) – Trackizer & Tomcat

- Infrastructure automation tools: Terraform, Ansible

- CI/CD tools: Jenkins

- Databases: MySQL, MariaDB

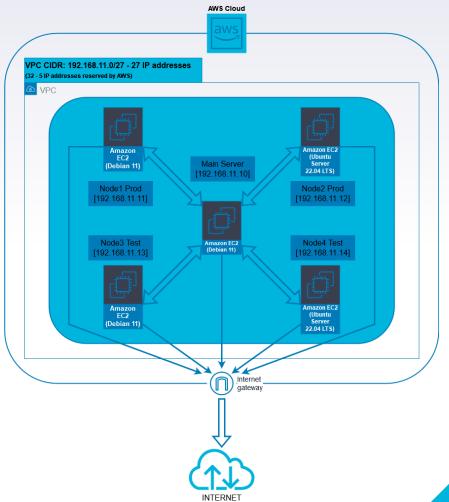- Cloud: AWS (Debian, Ubuntu)

- Dynamic DNS: No-IP

Terraform configuration for AWS

- Creates an Internet Gateway and attaches it to the VPC to allow traffic within the VPC to be reachable by the outside world

- Creates a public subnet

- Creates a route table for the public subnet and associates the table with the subnet

- Creates EC2 instances

5

A nsible

Main Server [192.168.11.10]
EC2 - Debian 11

my-tomcat-ansible

```
---
- name: Tomcat deployment playbook [Debian]
  hosts: debian                # Inventory hosts group / server to act on
  become: yes                  # If to escalate privilege
  become_method: sudo          # Set become method
  remote_user: admin           # Update username for remote server

  vars:
    tomcat_ver: 10.0.27        # Tomcat version to install
    ui_manager_user: manager   # User who can access the UI manager section only
    ui_manager_pass: _Str0ng   # UI manager user password
    ui_admin_username: admin   # User who can access both manager and admin UI sections
    ui_admin_pass: _Str0ngA    # UI admin password
  roles:
    - tomcat

- name: Tomcat deployment playbook [Ubuntu]
  hosts: ubuntu                # Inventory hosts group / server to act on
  become: yes                  # If to escalate privilege
  become_method: sudo          # Set become method
  remote_user: ubuntu          # Update username for remote

  vars:
    tomcat_ver: 10.0.27        # Tomcat version to install
    ui_manager_user: manager   # User who can access the UI manager section only
    ui_manager_pass: _Str0ngW  # UI manager user password
    ui_admin_username: admin   # User who can access both manager and admin UI sections
    ui_admin_pass: _Str0ng     # UI admin password
  roles:
    - tomcat
```

```
# ||| ansible.cfg |||
[defaults]
host_key_checking = false
inventory = ../hosts
```

my-ansible-insertDB

```
---
- name: Install DB into MySQL server
  hosts: db_servers            # Inventory hosts group / server to act on
  become: yes                  # If to escalate privilege
  become_user: root            # User on remote hosts
  become_method: sudo          # Set become method
  gather_facts: yes            # Collect useful variables about remote hosts

  tasks:

    - name: cd HOME
      shell: "cd $HOME"

    - name: mkdir trackizer_DB
      shell: "mkdir trackizer_DB"

    # Download DB source file to HOME/trackizer_DB
    - name: Download DB source file to HOME/trackizer_DB
      shell: wget https://raw.githubusercontent.com/Ivan2nas/L1_EPAN/main/1.N2@FinalN2@project/.db/test_database_v2
      args:
        chdir: ./trackizer_DB
        warn: false

    # - name: Copy DB from Git
    - name: Import DB
      shell: mysql -uroot -proot < ./trackizer_DB/test_database_v2

    - name: REMOVE trackizer_DB
      shell: "rm -r trackizer_DB"

    # ===== MAT :DB =====
    - name: Acquire temporary root password
      shell: "awk -F': ' '$0 ~ \"temporary password\"{print $2}' /var/log/mysqld.log"
      register: mysql_root_password_temp

    - name: Set new root password from default temporary password
      shell: "mysql -e \"SET PASSWORD = '{{ passwd_mysql_root }}';\""
      # --connect-expired-password -uroot -p'{{ mysql_root_password_temp.stdout }}'
      # && touch /root/.my.password_changed"
```

```
# ||| ansible.cfg |||
[defaults]
host_key_checking = false
inventory = ../hosts
```

my-mysql-ansible

```
# ||| ansible.cfg |||
[defaults]
host_key_checking = false
inventory = ../hosts
#ask_pass = false
roles_path=/home/admin/roles
```

```
---
- hosts: db_servers
  roles:
    - role: geerlingguy.mysql
      become: yes
  vars:
    mysql_root_password: root
    mysql_databases:
      - name: trackizer
    mysql_users:
      - name: root
#       host: "localhost"
        password: root
        priv: "*.*:ALL"
```

hosts

Left side panel

Tomcat
Git Clone

MySQL / MariaDB
Import from Ansible-galaxy

Trackizer database forMySQL / MariaDB
Copy from Git

BASH
THE BOURNE-AGAIN SHELL

Right side nodes

Node 1 Prod [192.168.11.11]
EC2 - Debian 11
Tomcat 10.x
MariaDB 10.x ✓

Node2 Prod [192.168.11.12]
EC2 – Ubuntu 22.04 LTS
Tomcat 10.x
MySQL 8.x ✓
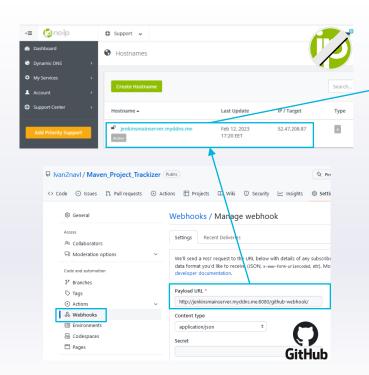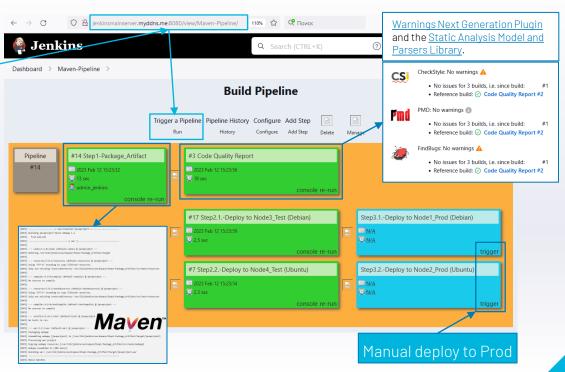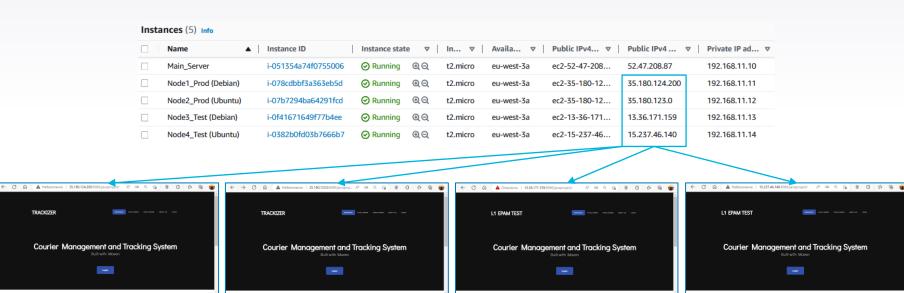
Node 3 Test [192.168.11.13]
EC2 - Debian 11
Tomcat 10.x
MariaDB 10.x ✓

Node 4 Test [192.168.11.14]
EC2 – Ubuntu 22.04 LTS
Tomcat 10.x
MySQL 8.x ✓

Jenkins

Demo (AWS)

Instances (5) Info

	Name ▲	Instance ID	Instance state ▽	In... ▽	Availa... ▽	Public IPv4... ▽	Public IPv4 ... ▽	Private IP ad... ▽
☐	Main_Server	i-051354a74f0755006	⊘ Running	t2.micro	eu-west-3a	ec2-52-47-208...	52.47.208.87	192.168.11.10
☐	Node1_Prod (Debian)	i-078cdbbf3a363eb5d	⊘ Running	t2.micro	eu-west-3a	ec2-35-180-12...	35.180.124.200	192.168.11.11
☐	Node2_Prod (Ubuntu)	i-07b7294ba64291fcd	⊘ Running	t2.micro	eu-west-3a	ec2-35-180-12...	35.180.123.0	192.168.11.12
☐	Node3_Test (Debian)	i-0f41671649f77b4ee	⊘ Running	t2.micro	eu-west-3a	ec2-13-36-171...	13.36.171.159	192.168.11.13
☐	Node4_Test (Ubuntu)	i-0382b0fd03b7666b7	⊘ Running	t2.micro	eu-west-3a	ec2-15-237-46...	15.237.46.140	192.168.11.14
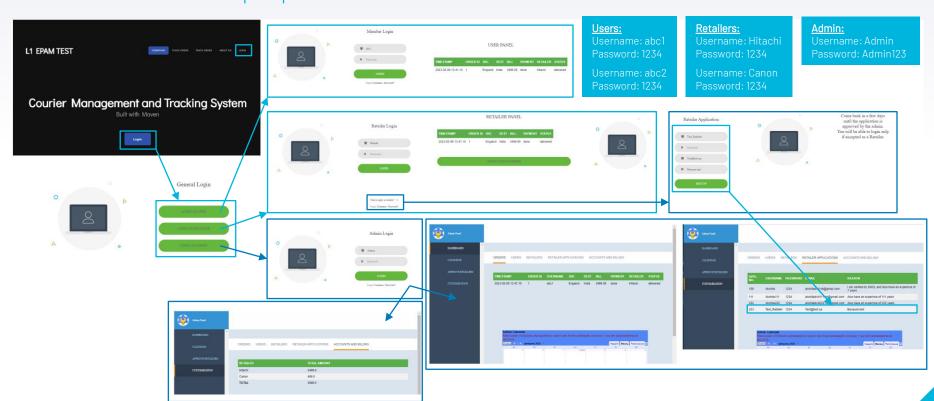
Web App Trackizer (Blocks)

Main Web Page
(with a description and credentials)

GitHub

Commit

GitHub Hook

Jenkins

Pipeline

Dockerfile

Files for Apache

Jenkinsfile

Artifact

Files for Apache
+
Dockerfile

Developer

docker

Main Web Page