Département Mathématiques et Informatiques

Compte Rendu de TP
Ingénierie des Infrastructures et DevOps

Filière d'ingénieur :

Ingénierie Informatique, Big Data et Cloud Computing

TP: Jenkins, CI/CD

Réalisé par : Abdellatif Hassani

Professeur: Pr. Oumayma AGHERAI

Année Universitaire : 2024-2025

Sommaire

I) Partie 1: CI (Continuous Integration)

1) Installation de Jenkins

```
abdellatif@abdellatif-HP-250-G8-Notebook-PC:~/ENSET/S5/DevOps/TP2$ sudo wget -O
/usr/share/keyrings/jenkins-keyring.asc \
  https://pkg.jenkins.io/debian/jenkins.io-2023.key
--2024-10-10 09:18:28--  https://pkg.jenkins.io/debian/jenkins.io-2023.key
Resolving pkg.jenkins.io (pkg.jenkins.io)... 151.101.130.133, 151.101.2.133, 151
.101.66.133, ...
Connecting to pkg.jenkins.io (pkg.jenkins.io)|151.101.130.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3175 (3.1K) [application/pgp-keys]
Saving to: '/usr/share/keyrings/jenkins-keyring.asc'

/usr/share/keyrings 100%[====================>]   3.10K  --.-KB/s    in 0s

2024-10-10 09:18:29 (15.7 MB/s) - '/usr/share/keyrings/jenkins-keyring.asc' save
d [3175/3175]

abdellatif@abdellatif-HP-250-G8-Notebook-PC:~/ENSET/S5/DevOps/TP2$ echo "deb [si
gned-by=/usr/share/keyrings/jenkins-keyring.asc]" \
  https://pkg.jenkins.io/debian binary/ | sudo tee \
  /etc/apt/sources.list.d/jenkins.list > /dev/null
abdellatif@abdellatif-HP-250-G8-Notebook-PC:~/ENSET/S5/DevOps/TP2$ sudo apt-get
update
Hit:1 https://mirror.marwan.ma/ubuntu noble InRelease
Hit:2 https://dl.google.com/linux/chrome/deb stable InRelease
```

```
abdellatif@abdellatif-HP-250-G8-Notebook-PC:~/ENSET/S5/DevOps/TP2$ sudo apt-get
install jenkins
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  acl apg colord-data cups-pk-helper fonts-opensymbol gir1.2-notify-0.7
  gnome-control-center-faces gnome-online-accounts heif-gdk-pixbuf
```

```
abdellatif@abdellatif-HP-250-G8-Notebook-PC:~/ENSET/S5/DevOps/TP2$ sudo systemct
l status jenkins
● jenkins.service - Jenkins Continuous Integration Server
     Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: >
     Active: active (running) since Thu 2024-10-10 09:23:28 +01; 54s ago
   Main PID: 12153 (java)
      Tasks: 51 (limit: 14011)
     Memory: 852.8M (peak: 867.8M)
        CPU: 29.013s
     CGroup: /system.slice/jenkins.service
```
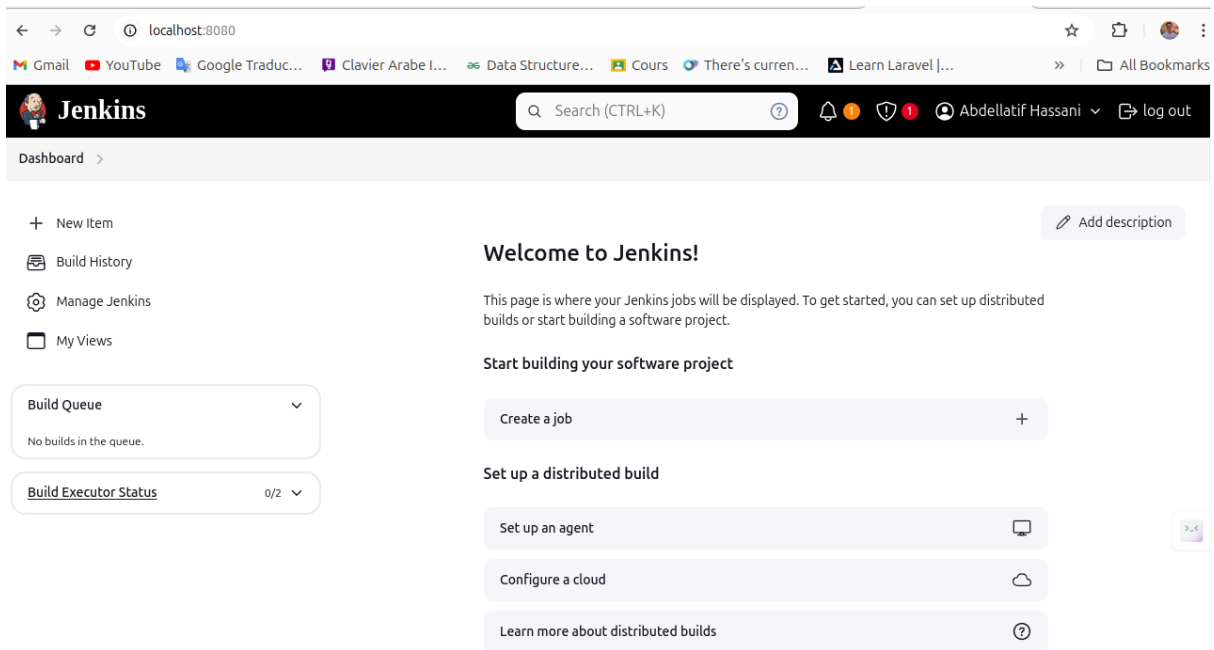
2) Création et configuration d'une simple page web


```
abdellatif@abdellatif-HP-250-G8-Notebook-PC:~/ENSET/S5/DevOps/TP2$ touch index.html
abdellatif@abdellatif-HP-250-G8-Notebook-PC:~/ENSET/S5/DevOps/TP2$ nano index.html
abdellatif@abdellatif-HP-250-G8-Notebook-PC:~/ENSET/S5/DevOps/TP2$ touch Dockerfile
abdellatif@abdellatif-HP-250-G8-Notebook-PC:~/ENSET/S5/DevOps/TP2$ nano Dockerfile
abdellatif@abdellatif-HP-250-G8-Notebook-PC:~/ENSET/S5/DevOps/TP2$ cat Dockerfile
# Use the official Nginx image from Docker Hub
FROM nginx:alpine

# Copy the HTML file into the Nginx web directory
COPY index.html /usr/share/nginx/html/

# Expose port 80
EXPOSE 80

# Start Nginx when the container runs
CMD ["nginx", "-g", "daemon off;"]
```
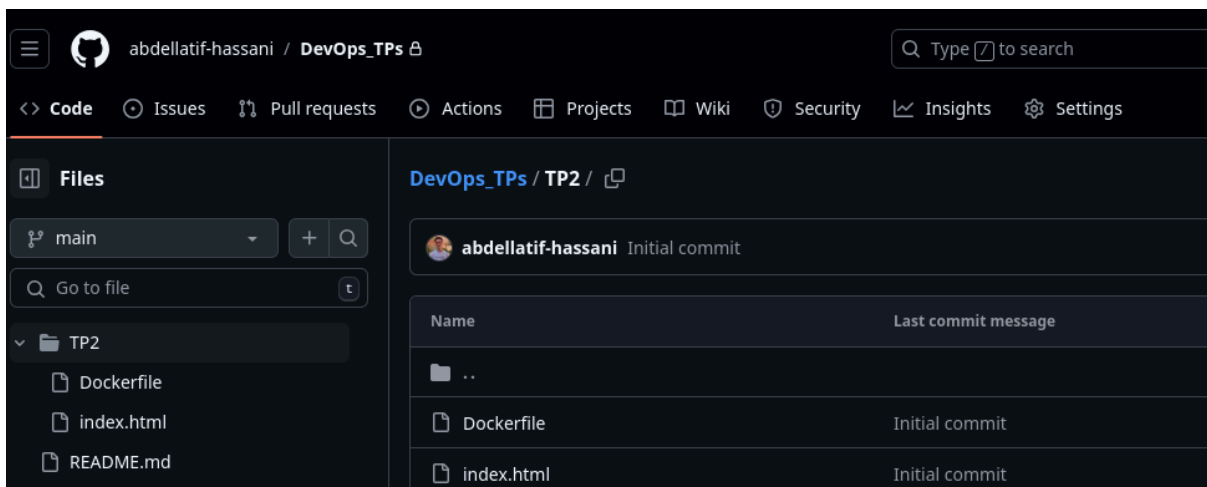
3)   Création d'un repository github

4) Création et configuration d'un Job Jenkins

Dashboard > New Item

Enter an item name

Job1 TP2

Select an item type

Freestyle project
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders

OK

5) Liste des Services Actifs dans le Cluster

Docker plugin 1.6.2
This plugin integrates Jenkins with Docker
Report an issue with this plugin

docker-build-step 2.12
This plugin allows to add various docker commands to your job as build steps.
Report an issue with this plugin

Warning: The currently installed plugin version may not be safe to use. Please review the following security notices:

- CSRF vulnerability and missing permission check

6) Configuration de job1tp2 pour générer et publier une image Docker du projet sur Docker Hub (Tag latest)

Dashboard > Job1 TP2 > Configuration

Configure

- ⚙️ General
- ⅄ **Source Code Management**
- 🕑 Build Triggers
- 🌐 Build Environment
- ▤ Build Steps
- 📦 Post-build Actions

Configure

- ⚙️ General
- ⅄ Source Code Management
- 🕑 Build Triggers
- 🌐 Build Environment
- ▤ **Build Steps**
- 📦 Post-build Actions

● Git ?

Repositories ?

Repository URL ?

https://github.com/abdellatif-hassani/DevOps_TPs

Credentials ?

abdellatifhassani1205@gmail.com/******

+ Add

- none -

+ Add

Docker registry URL ?

Registry credentials

abdellatif2002/******

+ Add

☑ Environment variables

List of variables ?

Name

DOCKER_HOST

Value

tcp://172.17.0.1:2375

- Le résultat : la création de l'image dans le repository DockerHub

```
#8 writing image sha256:3c1394f921fd014ec6ecac2f8c8d8b23852b1fcec79715711033b955e364d494 done
#8 naming to docker.io/abdellatif2002/web-app1:latest done
#8 DONE 0.0s
+ echo ****
+ docker login -u abdellatif2002 --password-stdin
WARNING! Your password will be stored unencrypted in /var/lib/jenkins/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credential-stores

Login Succeeded
+ docker push abdellatif2002/web-app1:latest
The push refers to repository [docker.io/abdellatif2002/web-app1]
```

```
latest: digest: sha256:99fa6699eebd1cf73ce7a9e562616aaa811643125c687ed1cc257d84cd992ed3 size: 2196
Finished: SUCCESS
```

abdellatif2002/web-app1

Last pushed 4 minutes ago

This repository does not have a description ✏ ⓘ INCOMPLETE

This repository does not have a category ✏ ⓘ INCOMPLETE
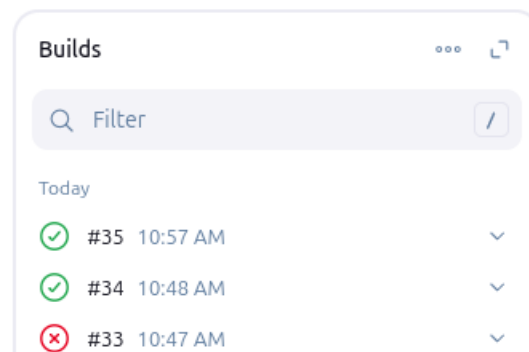
Tags

This repository contains 1 tag(s).

Tag	OS	Type	Pulled	Pushed
● latest	🐧	Image	9 minutes ago	4 minutes ago

7) Faire un changement dans index.html, découvrir les changements sur le job1tp2.

Un nouveau job est déclenché :

II) Partie 2: CI/CD (continuous delivery/continuous deployment)

1) Création d'un CI/CD pipeline qui permet de : build, test et deployer une application web d'une une instance EC2 AWS en utilisant Jenkins pipelines:

- Configuration de Pipeline

- Création d'une machine virtuel en AWS (EC2)

- Jenkins file :

```
pipeline {
    agent any
    environment {
        DOCKERHUB_CREDENTIALS = credentials('dockerhub-cred')
        DOCKER_IMAGE = "abdellatif2002/web-app1"
        DOCKER_TAG = "v${BUILD_NUMBER}"
        EC2_SERVER = "13.38.117.4"
    }
    stages {
        stage('Checkout') {
            steps {
                checkout scm
            }
        }
        stage('Build Docker Image') {
            steps {
                sh "docker build -t ${DOCKER_IMAGE}:${DOCKER_TAG} ."
                sh "docker tag ${DOCKER_IMAGE}:${DOCKER_TAG} ${DOCKER_IMAGE}:latest"
            }
        }
        stage('Login to DockerHub') {
            steps {
                sh 'echo $DOCKERHUB_CREDENTIALS_PSW | docker login -u $DOCKERHUB_CREDENTIALS_USR --
password-stdin'
            }
        }
        stage('Push to DockerHub') {
            steps {
                sh "docker push ${DOCKER_IMAGE}:${DOCKER_TAG}"
                sh "docker push ${DOCKER_IMAGE}:latest"
            }
        }
        stage('Deploy to EC2') {
            steps {
                script {
                    sshagent(credentials: ['ec2-ssh-key']) {
                        sh """
                            ssh -o StrictHostKeyChecking=no ubuntu@${EC2_SERVER} '
                                # Verify Docker is running
                                if ! docker info > /dev/null 2>&1; then
                                    echo "Docker is not running or not installed"
                                    exit 1
                                fi

                                # Pull the new image
                                docker pull ${DOCKER_IMAGE}:latest

                                # Stop and remove existing container
                                if docker ps -a | grep -q webapp1; then
                                    docker stop webapp1 || true
                                    docker rm webapp1 || true
                                fi

                                # Run the new container
                                docker run -d -p 80:80 --name webapp1 ${DOCKER_IMAGE}:latest

                                # Clean up old images
                                docker system prune -f
                            '
                        """
                    }
                }
            }
        }
    }
    post {
        always {
            sh 'docker logout'
        }
        success {
            echo 'Pipeline completed successfully!'
        }
        failure {
            echo 'Pipeline failed! Check the logs for details.'
        }
    }
}
```
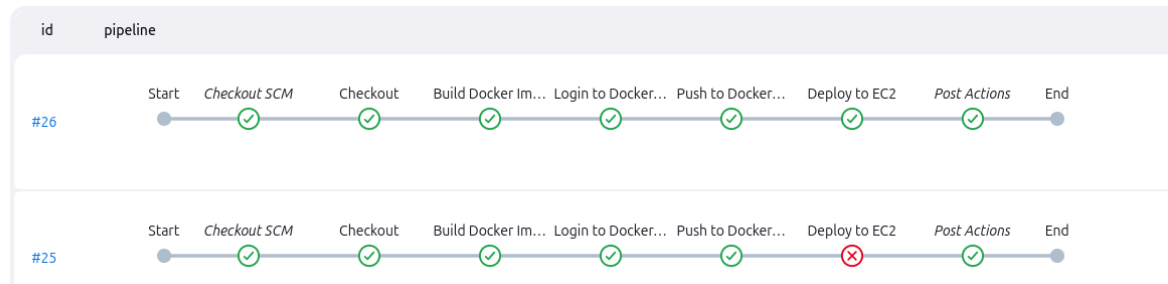
- Stages de pipeline:

```
ubuntu@ip-172-31-45-178:~$ curl localhost
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>TP 2</title>
</head>
<body>
    <h1>ENSET BDCC-3 V1</h1>
</body>
</html>
ubuntu@ip-172-31-45-178:~$ curl localhost
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>TP 2</title>
</head>
<body>
    <h1>ENSET BDCC-3 V2</h1>
</body>
</html>
ubuntu@ip-172-31-45-178:~$
```