Contents

[Project Name: 1](#_Toc67983248)

[Description: 1](#_Toc67983249)

[DevOps Tools Used 1](#_Toc67983250)

[Steps to Implement 1](#_Toc67983251)

[Jenkins Server 1](#_Toc67983252)

[Docker Server 2](#_Toc67983253)

[Output: 3](#_Toc67983254)

# Project Name:

Continuous Integration to deploy the applications in Docker Containers

# Description:

Create a Continuous Integration Job that Pulls the Code from GitHub repository, Builds the application with Maven, Copies the application package (in war/jar format) into Docker Server and creates the Docker Container

# DevOps Tools Used

* Git
* Maven
* Jenkins
* Docker

# Steps to Implement

## Jenkins Server

* Login to AWS Console
* Create EC2 Instance with Redhat Enterprise Linux 8 AMI
* Login to EC2 Instance from your local system using Git Bash or any other tools
* Install Git
* Install Maven
* Install Jenkins
  + Once Jenkins is installed, Open the http://<Public-IP of the Server>:8080 in a browser
  + Unlock the Jenkins by following the instructions
  + Install Suggested Plugins
  + Create Admin user
  + Then once you logged into Jenkins
    - Go to Manage Jenkins 🡪 Manage Plugins 🡪 Available Tab
    - Search for Publish Over SSH then install the Plugin
  + Mange jenkins 🡪 Configure System
    - Add SSH Server
    - Configure Docker Server Details
      * Hostname: Private IP of Docker server
      * Username
      * Password
  + Integrate Git, Some of the Git Projects to be considered
    - <https://github.com/daticahealth/java-tomcat-maven-example.git>
      * Software needed to run this in Docker Container is
        + Tomcat 8
        + Java8
    - <https://github.com/spring-projects/spring-petclinic.git>
      * Software needed to run this in Docker Container is
      * Java8
  + Integrate Maven
  + Integrate Docker

## Docker Server

1. Create EC2 Instance (ubuntu AMI) for Docker
2. Login to EC2 Instance from Git Bash

* Install Docker
* Create User named dockeradmin and set the Password
* sudo usermod -aG docker dockeradmin
* sudo service restart docker
* Enable Password Authentication (sudo -I or sudo su to become the root user and execute below tasks with root user)
  + vi /etc/ssh/sshd\_config
  + Uncomment PasswordAuthentication Yes
  + Comment PasswordAuthentication No
  + service sshd restart
* Create Dockerfile in /opt/docker
  + Create the docker folder under /opt if it does not exist
  + Dockefile should have entries
    - To install required software for the application package
    - Add Application Package into Container
    - Expose Port
    - Application Start Command

FYI -- Jenkins Workspace in Linux - /var/lib/jenkins/workspace/

# Output:

* Once the Build Job is executed successfully from Jenkins
  + The Docker Container should be running in Docker server
  + Copy Public IP of Docker Server and access it from Browser to see if you can access the application, we deployed in container