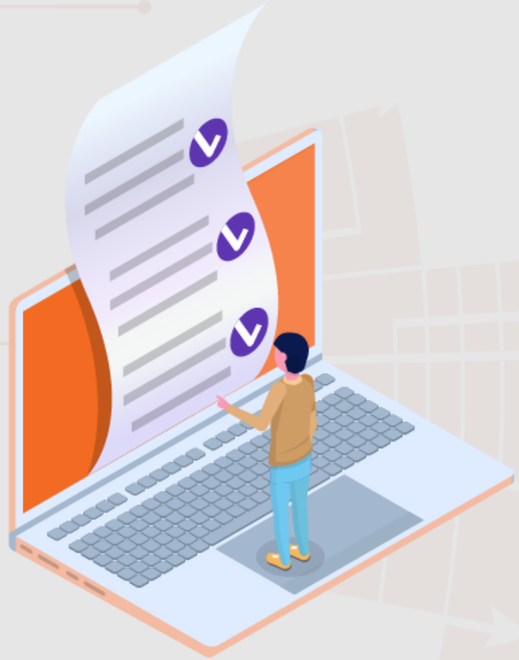
Integration and Deployment

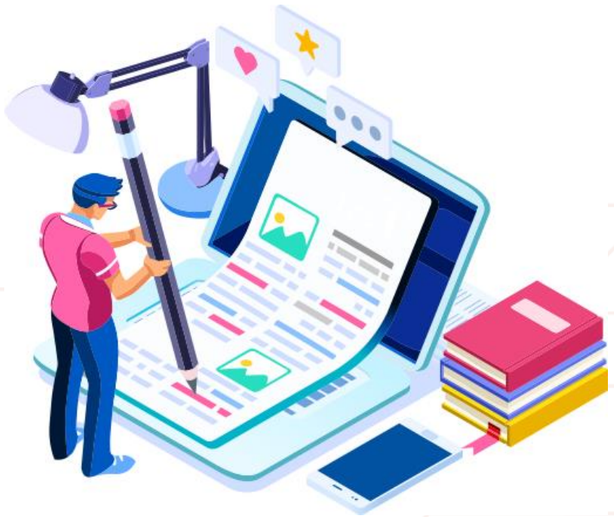
Course-End Project

Objectives

To build an infrastructure to host a software on AWS EC2 instance for a clinic to have online access to the pets' data and their owners visit along with the consultation given.

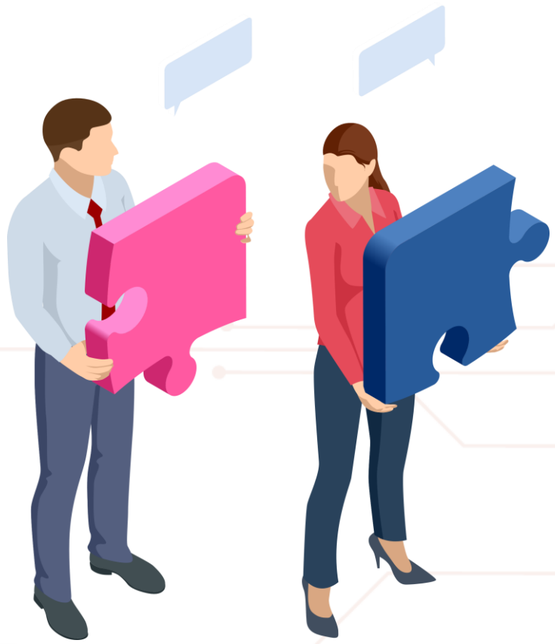


Prerequisites



- SpringBoot
- Jenkins
- Docker
- AWS

Problem Statement and Motivation



Problem Statement:

This assignment is designed to help understand how to plan and develop the back end for a given problem. Further, to gain hands-on experience building the CI/CD Pipeline using Jenkins and then containerizing the application on the AWS cloud platform.

Real-World Scenario:

Dr. Shawn runs a pet clinic. He needs to record the visits and other details associated with the pets and their owners visiting his clinic. He has software developed by Bella Solutions, a software company, to manage the same.

Bella Solutions aims to host the software solution for Dr. Shawn on AWS EC2 instance to have online access from anywhere by building CI CD Pipeline and containerizing the solution using Docker on AWS EC2.

Industry Relevance

Skills used in the project and their usage in the industry are given below:

Jenkins:

Popular CI/CD pipeline tool that helps in automating and integrating multiple technologies and stages of software development.

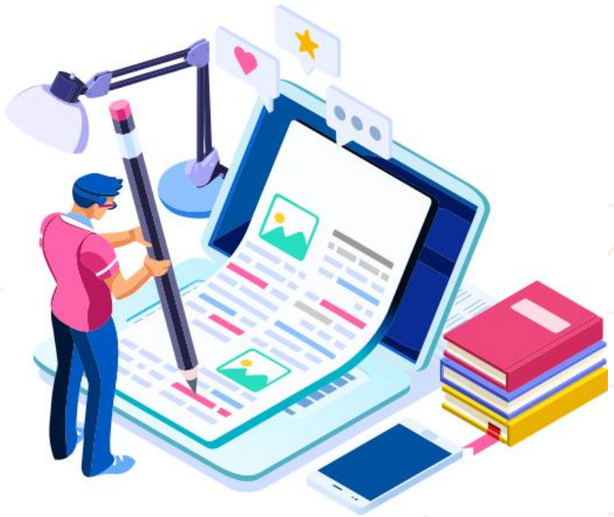
Docker:

Apart from various use cases for Docker, it is majorly used for containing an entire application as a light weight image which then can be deployed on multiple server.

AWS and EC2:

AWS is a one of the most used cloud platform for managing different applications and projects of any scale. EC2 is one of simplest way to get an application deployed.

Task (Activities)



1. Import the given Spring Boot project with the generated code in Eclipse
2. Configure the project with Dockerfile and Jenkinsfile
3. Build the project using the maven package
4. Create and Launch AWS EC2 Instance
5. Configure EC2 Instance with JDK 11, Docker and Jenkins
6. Sync the given code to git repo
7. Create Jenkins Pipeline on EC2 with Git and GitHub
8. Build the pipeline to dockerize the application

Project Reference

Task 1:

Springboot – Lesson 1

Task 2:

Jenkins – Lesson 3

Docker – Lesson 3

Task 3:

Springboot – Lesson 1

Task 4 and 5:

Jenkins – Lesson 3

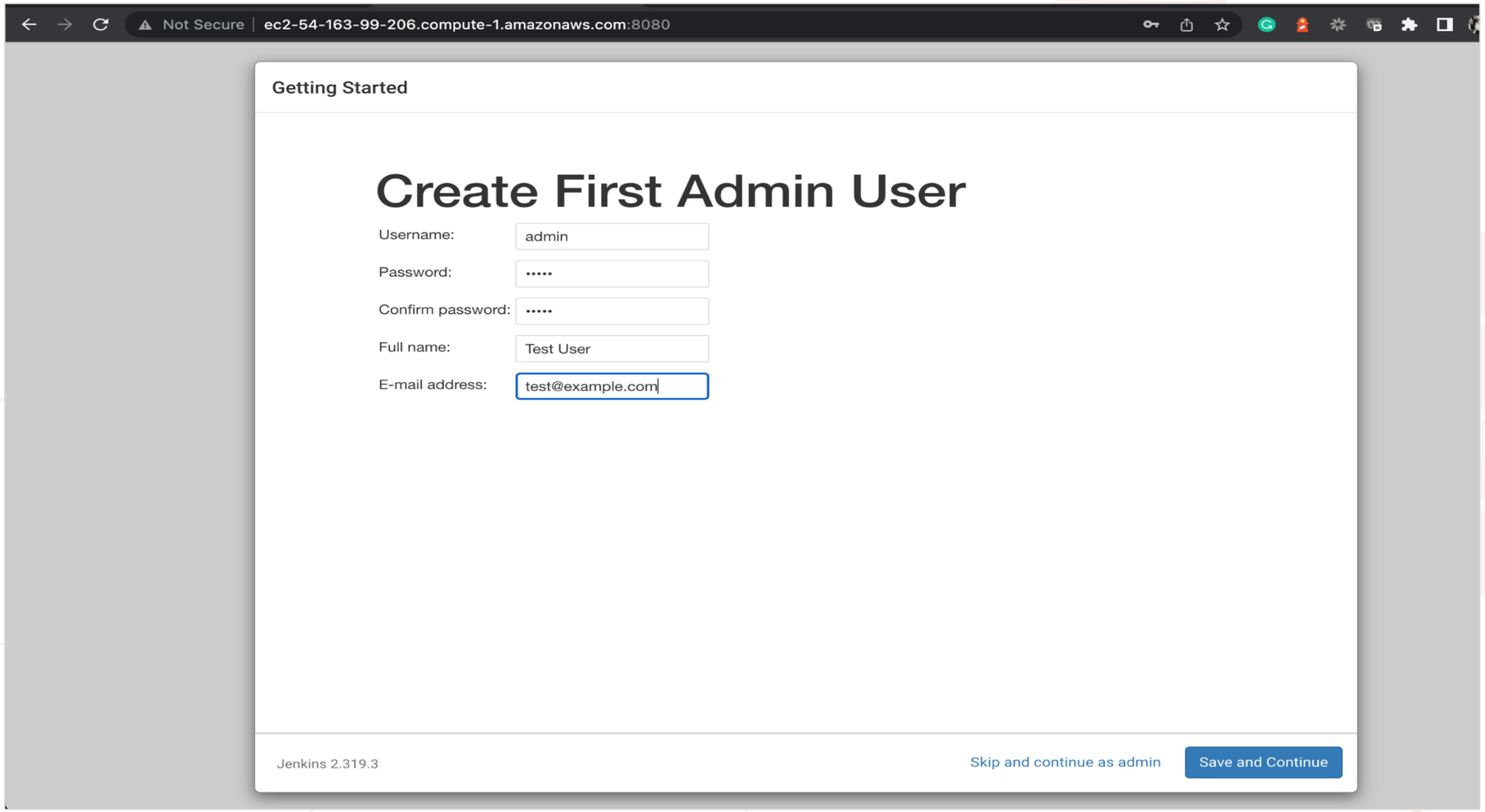
Task 6 and 7:

Jenkins – Lesson 1

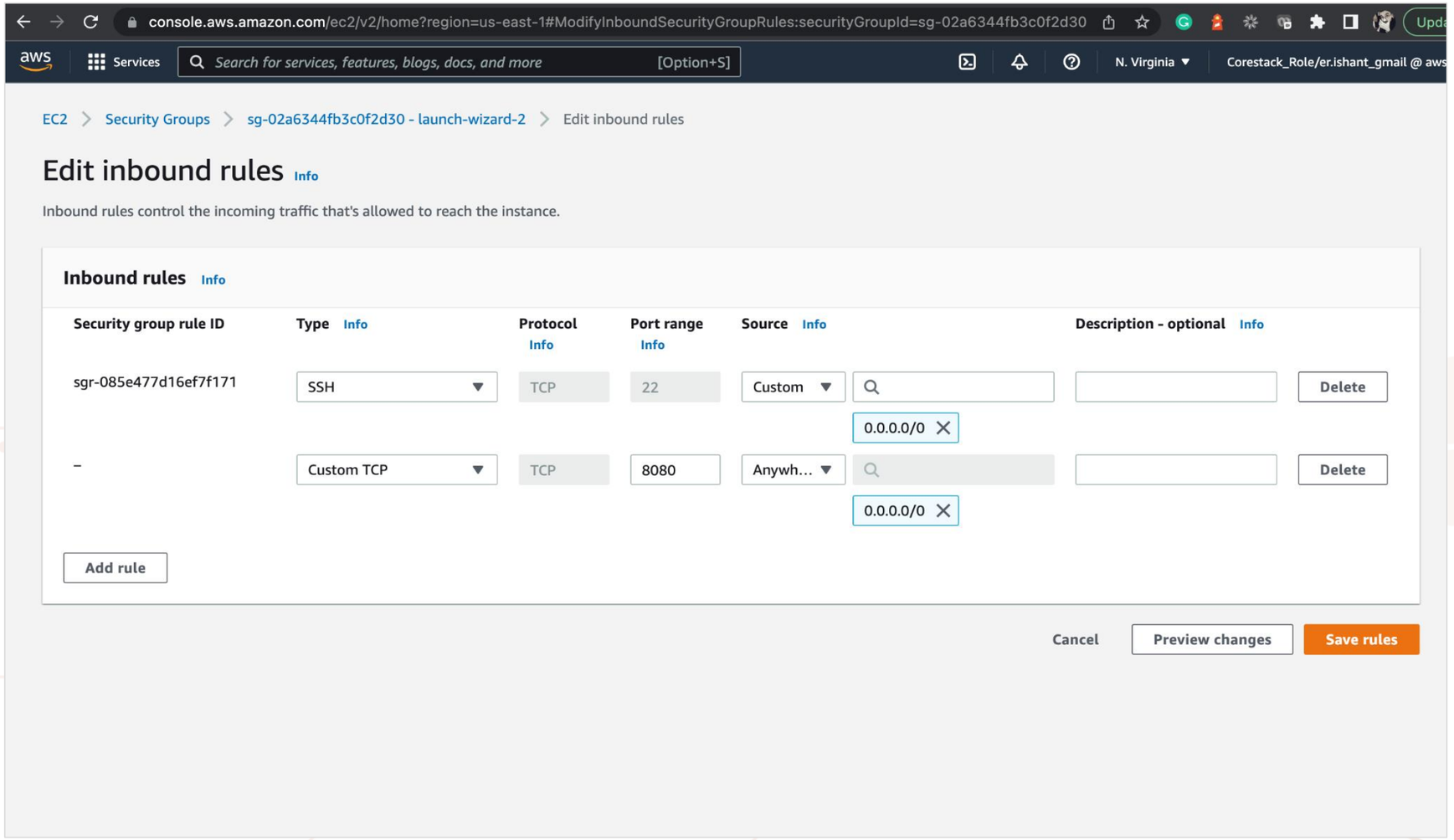
Task 8:

Docker – Lesson 2 and 3

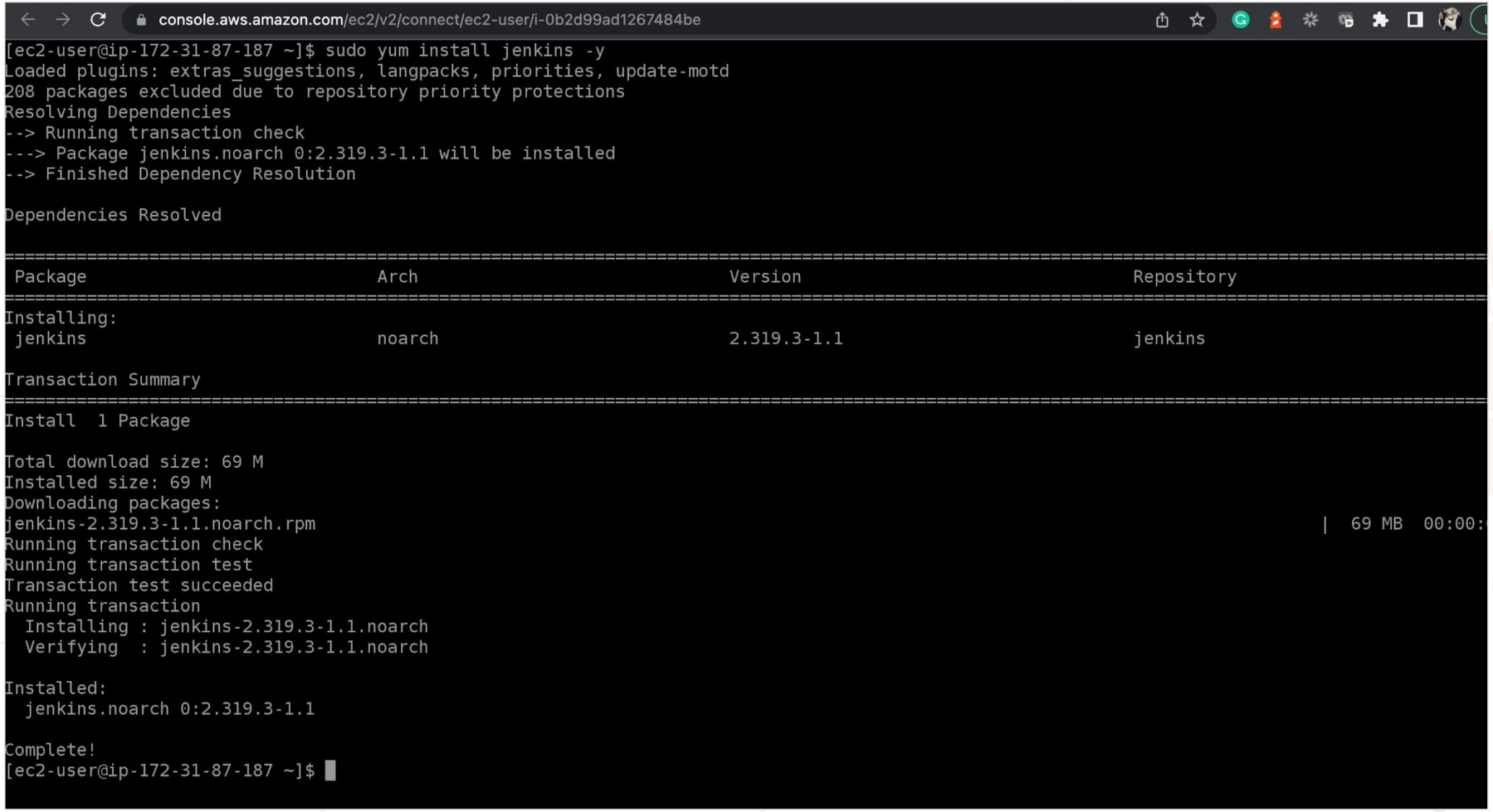
Reference Outputs



Reference Outputs



Reference Outputs



Thank you