**SWE-645 HW2**

In order to complete Homework2 we need to do the following steps. This document provides detail explanation of the steps along with screenshots. It has URLs of each completed tasks for reference. The tasks given in assignment is given below:

1. Building an image for the survey.html application and pushing it to Docker Hub using Docker Desktop.
2. Use Rancher to set up AWS EC2 instances in order to deploy the application on a Kubernetes cluster.
3. Install and Set Up Rancher
4. Deploy the Kubernetes Cluster Using Rancher UI
5. Set Up AWS EC2 Instance for Jenkins
6. Set Up GitHub Repository for project
7. Create a CI/CD Pipeline with Jenkins

**Step 1: Building an image for the survey.html application and pushing it to Docker Hub using Docker Desktop.**

1. First, create an account through Docker hub (<https://hub.docker.com/>). Then download and install docker desktop through docker website (<https://www.docker.com/>). Verifying the installation:

A screenshot of a computer

Description automatically generated

1. Login to the docker desktop through the app that was downloaded before and login to docker hub through web browser.

A screenshot of a computer

Description automatically generated

1. Preparing the folder for Survey.html. In order to push into the Docker hub, First Create StudentSurvey folder and push all the relevant html files into the folder.

A screenshot of a computer

Description automatically generated­­

1. ­­Go to parent directory of above html files and use the command to Generate StudentSurvey.war as “jar -cvf studentsurvey.war -C StudentSurvey . “

A screenshot of a computer

Description automatically generated

1. After generating war file make sure to keep it in the same folder as above. Now create docker file in the same directory as of StudentSurvey.war file. To create docker file we use **nano Dockerfile** then add content as shown below

A screenshot of a computer

Description automatically generated

1. Create a docker image by opening the terminal at the directory created above. The command used to create docker image is as follows:

**docker build -t studentsurvey .**

**A screenshot of a computer screen

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

1. As shown in above picture, image is successfully created. Now we need to run the above image at a port using the following command.

**docker run -it -p 8082:8080 studentsurvey**

**A screenshot of a computer

Description automatically generated**

1. It is visible that application will be operational on localhost:8082/warfile name

URL: <http://localhost:8082/studentsurvey/survey.html>

A computer screen shot of a survey form

Description automatically generated

A screenshot of a computer

Description automatically generated

1. If I click on cancel button on survey.html it will redirect to index.html page

A computer screen shot of a computer

Description automatically generated

1. After that we can push our image to docker hub using following commands.

🡪 docker login -u prasanna0307

A screenshot of a computer

Description automatically generated

🡪 docker tag studentsurvey prasanna0307/studentsurvey1

🡪 docker push prasanna0307/studentsurvey1

A screen shot of a computer

Description automatically generated

🡪 It can be visible in the following image

A screenshot of a computer

Description automatically generated

**Step 2: Set up AWS EC2 instances in order to deploy the application on a Kubernetes cluster.**

1. We need to install an AWS EC2 instance to do this step. Log in to the AWS console <https://aws.amazon.com/> and create an account.
2. After logging into AWS Academy, clicking on Launch AWS Academy Learner Lab, we need to click on start lab then click on AWS green dot we can launch AWS console.
3. In AWS console, Click on Launch Instance option to create an instance with the following specifications.

🡪 AMI used: Ubuntu 24.04 HVM SSD

🡪 t3.large

🡪 30 GiB Storage

1. We need two of those instances for this task. And we need to create an Elastic IP address and need to allocate them to the instances we created which will be useful for Rancher and Jenkins.
2. Now we need to assign inbound and outbound rules by clicking security tab under the instances.

🡪 Type: Custome TCP, Port : 8080, Source: 0.0.0.0/0

1. After setting up all the configurations we need to launch EC2 instance and needs to connect to it. We need to enter username as “root” and then click on connect we will the following in the new window.

A screenshot of a computer

Description automatically generated

1. We need to run following Commands in EC2 instance.

🡪 sudo apt update

🡪 sudo apt install docker.io -y

After this docker should be successfully installed

A screenshot of a computer

Description automatically generated

1. It indicates that docker is successfully installed. And the instance will look as follows:

A screenshot of a computer

Description automatically generated

**Step3: Install and Set Up Rancher**

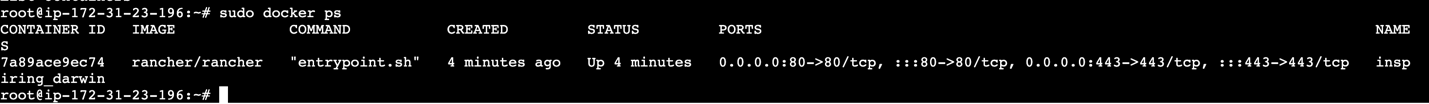
1. Click <https://www.rancher.com/quick-start> and navigate to the link. Under Rancher will find Deploy Rancher we have some instructions as shown below.

A screenshot of a computer

Description automatically generated

Copy and paste the command that is there in Start the server section.

1. After running the command docker ps, we will get the following output



1. We need to go to the “Public IPV4 DNS” in the instance page. There it redirects to Rancher. It initially shows the warning but we can continue with that. It looks something like this:

Screens screenshot of a computer

Description automatically generated

1. We need to copy the command and replace the container id with (7a89ace9ec74) this is the one we got above after running docker ps command so that it will displays the password.
2. Copy the password and paste it in the rancher login screen that is displayed above.

After that we will be able to create a new password. After successfully login it will display the following screen.

A screenshot of a computer

Description automatically generated

**Step 4: Deploy the Kubernetes Cluster Using Rancher UI**

1. As the Rancher is ready, with this UI we can create a cluster and deploy the survey.html file on it.
2. In the dashboard shown above we can click on the create button. It will displays the following

A screenshot of a computer

Description automatically generated

1. Click on Custom, It will displays a form to fill in which we need to give all the details about the cluster like name as “swehw2” leaving remaining default values as it and then proceed clicking on create.
2. In step1, we need to check all the boxes. Now copy the command which is there in the step2 and paste it in the EC2 instance and run it. The cluster is in updating state for few minutes then it will become active.

A screenshot of a computer

Description automatically generated

1. Once the cluster is active, click on explore mode

A screenshot of a computer

Description automatically generated

1. After active state, click on the created cluster and On the left side display we have several options. Choose workloads and then deployment.Click create then we will get the form. Give details as below:

🡪 Name : swehw2deploy, Replicas(number of pods): 3, Container image: prasanna0307/studentsurvey1:latest (which is available on docker hub), click on Add port or service" button under Networking, Provide the following information:

🡪protocol: TCP, private container port: 8080, service type: "node Port," and name: "nodeport."

Leave remaining details as is and click create. It will take few minutes to update and after that it will displays the active status as shown below.

A screenshot of a computer

Description automatically generated

1. Once it is in active status, go inside the deployment and go to services tab.

A screenshot of a computer

Description automatically generated

1. In services tab, under the target option beside the nodeport. It will redirect to the new window. In that for the URL add /studentsurvey

🡪 If I click on 8080 nodeport I got this URL:

https://18.209.25.4/k8s/clusters/c-m-jqfx6dxn/api/v1/namespaces/default/services/http:deployswehw2:8080/proxy/

🡪After adding /studentsurvey the URL is as follows:

<https://18.209.25.4/k8s/clusters/c-m-jqfx6dxn/api/v1/namespaces/default/services/http:deployswehw2:8080/proxy/studentsurvey/>

A computer screen shot of a survey form

Description automatically generated

**Step 5: Set Up AWS EC2 Instance for Jenkins**

1. In this step, Jenkins is used which will be helpful for developing CI/CD pipeline that will automatically build and update our source code.
2. Download and keep aside the KubeConfig file from the Rancher UI i.e. In Cluster Management dashboard

A screenshot of a computer

Description automatically generated

1. We need to install Jenkins for that we need an AWS EC2 instance on the AWS lab. We can use the 2nd instance that we created in Step2 which named as “SWE645\_hw2\_jenkins”.

A screenshot of a computer

Description automatically generated

1. Launch EC2 instance connect by clicking on Connect button shown above. After launching it we need to install java. Use following commands to do so:

🡪 sudo apt update

🡪 sudo apt install openjdk-17-jdk -y

🡪 To install Jenkins: sudo wget -O /usr/share/keyrings/jenkins-keyring.asc\https://pkg.jenkins.io/debianstable/jenkins.io2023.key

🡪 echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] https://pkg.jenkins.io/debian-stable binary/" | sudo tee /etc/apt/sources.list.d/jenkins.list > /dev/null

🡪 sudo apt update

🡪 “sudo apt-get install Jenkins -y

🡪 sudo systemctl status jenkins.service

🡪Expose in port 8080 using “sudo ufw allow 8080”

🡪 To get admin password we need to run the following command: “sudo cat /var/lib/jenkins/secrets/initialAdminPassword” (we need to secure this password).

0270c1d58a9d43289b321b48df68fe84

1. To set up the snapd, set the following commands:

🡪 sudo apt install snapd

🡪To install kubectl using snap : “sudo snap install kubectl –classic”

1. Open the “Public IPv4 address” in new tab. Add 8080 at the end. Change from https to http

🡪 <http://3.211.44.234:8080/>

1. The above URL will open a login page. We need to use the generated password above to open Jenkins.
2. It will ask to choose an option, select install selected plugins. Then it will ask to create admin user with username, password, fullname and email id. Fill and click on “save and next”. Leave everything else as it is and click save and finish. Then click on start Jenkins.
3. We will be able to see Jenkins Dashboard. Return to the EC2 instance to build our configuration file.
4. Run the following command to go to Jenkins home

🡪 sudo su Jenkins

🡪 cd ../.. to go to root directory

🡪 cd /var/lib/jenkins

1. Create a directory and go to that directory

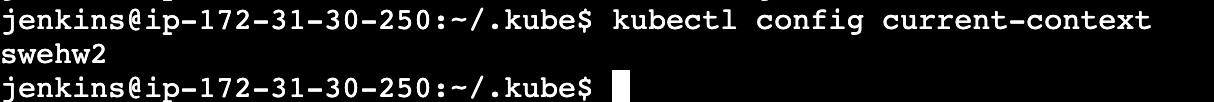
🡪 mkdir .kube and cd .kube

1. Create and open a file by typing “vi config.”. Now copy the contents of KubeConfig file that we downloaded in (2) in this step. Paste contents in the file. Save and quit using :wq

A screenshot of a computer

Description automatically generated

1. Run the command “kubectl config current-context” it will confirm by displaying the cluster name.



1. Type exit to end the Jenkins.

A screenshot of a computer

Description automatically generated

1. Install the docker in this EC2 Instance.

🡪 sudo apt update

🡪 sudo apt install docker.io -y

🡪 sudo apt-get update

This will successfully install the docker on our laptops.

**Step 6: Set Up GitHub Repository for project**