SWE645 – HW2 – Installation and Setup Instruction

Team Members:
Preeti Bhattacharya G01302375,
Uday Kumar Kamalapuram G01340201,
Paras Kanth G01333828.

A. **Setting Git Repository**

1. We are using the Survey form which we created in Homework1.
2. To use this survey form we are pushing our project in the git repository.
3. To push we are using “Teams” in eclipse for that src->[Right click] ->Team->Commit.

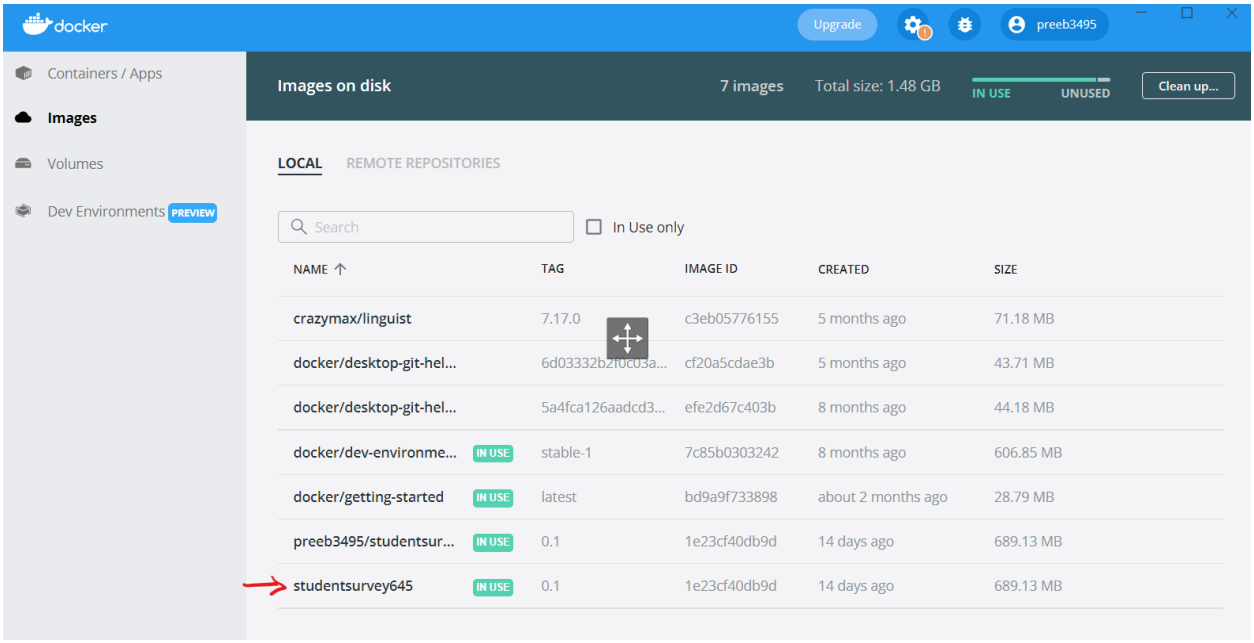
B. **Creating Docker Image and Pushing it in Docker hub**

1. Install “docker desktop” on your desktop.
2. Create your account using “https://hub.docker.com/”.
3. In Eclipse, create a file called “Dockerfile”.
4. Put the Survey form war file in the same folder as docker file. Here the name is Survey_Form.war
5. In the docker file write the below code.

```
1 FROM tomcat:9.0-jdk15
2 COPY Survey_Form.war /usr/local/tomcat/webapps/
3
```

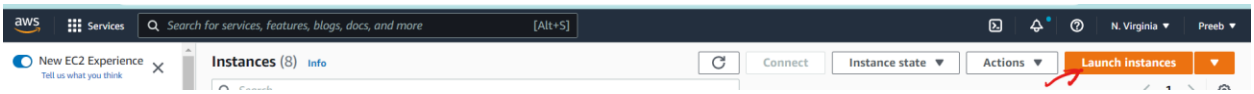
6. Open Cmd and use this command “docker build --tag studentsurvey645”.
7. We are verifying by running “docker run -it -p 8182:8080 studentsurvey645” to check whether the image is running properly.

8. To check, open the browser at “http://localhost:8182/Survey_Form”.
9. In cmd login docker using “docker login -u <your username>”.
10. Change the name of your image to be <your username on dockerhub>/<name of the app>:<image tag> using the docker tag command.
11. In our case: ‘docker tag studentsurvey645:0.1 preeb3495/studentsurvey645:0.1
12. Verify your image on dockerhub and it should be accessible through the internet.



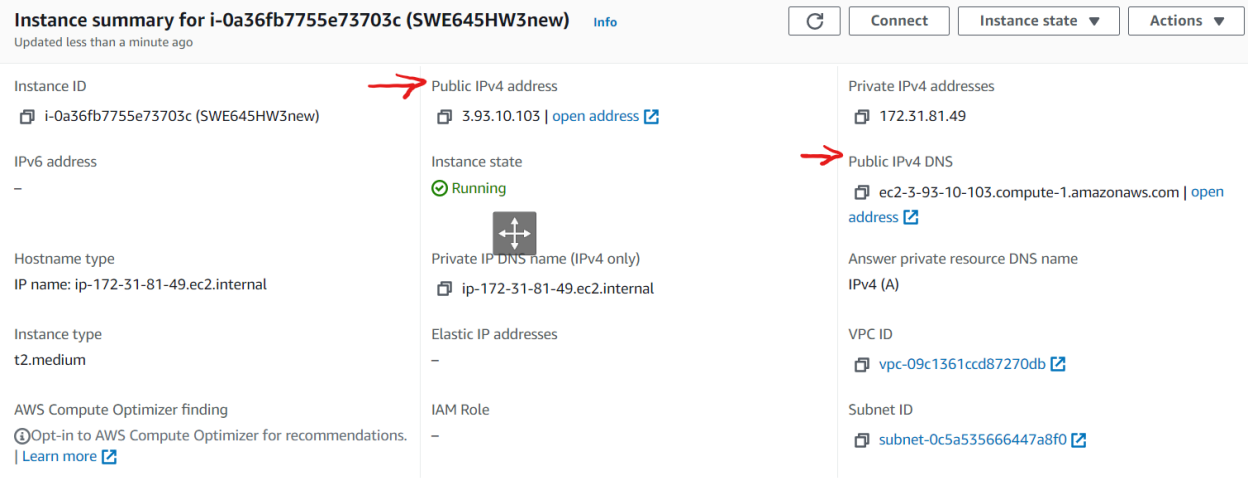
C. **Creating AWS EC2 instance and installing Rancher**

1. Log in to the AWS console https://aws.amazon.com/ and create an account.
2. Navigate to EC2.
3. Launch a new instance.



4. Search Ubuntu AMI.
5. Search for Ubuntu AMI and select the Ubuntu Server 20.04 LTS (HVM), SSD Volume Type image and click Select.
6. Select t2.medium to support Kubernetes.
7. Then click at Next: Add Storage.
8. Then click at Next: Add Tags.And then we have given Key as “Name” and Value as “SWE645HW3new”.
9. Then click at Next: Configure Security Group.
10. Then create SSH,HTTP,HTTPS security groups with “anywhere” accessible and also add Custom TCP port 8080 for Jenkins to work with and “anywhere” accessible.
11. Then click Review and Launch.
12. Review your configure and Launch.
13. Create a new key pair to access EC2 machine and download it for further use.
14. Then click Launch Instances.

15. Then click View Instance after that you can see your instance up and running after some time.
16. To see its configuration like public IPv4 address, Public IPv4 DNS click on instance ID.



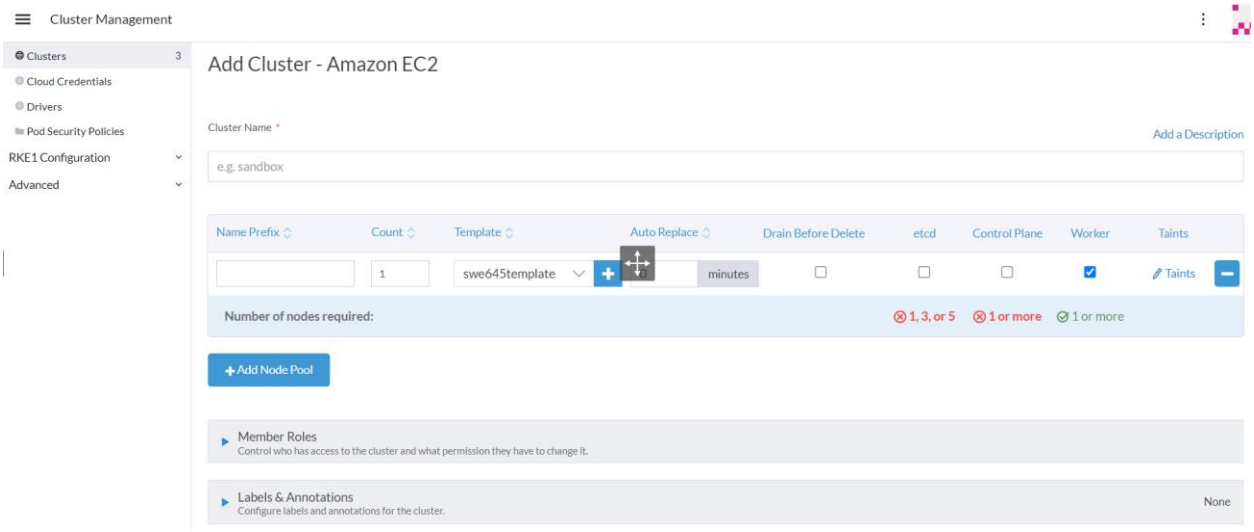
C. **To create an IAM user**

1. Navigate to the IAM user dashboard.
2. Click on users on the left panel.
3. Click Add users.
4. We have given the username as “swe645hw3”.
5. And under AWS access type, select Access Key- Programmatic access.
6. Then click Next:Permissions.
7. Then click on Attach existing policies directly and click next
8. Then review configuration and click create User.
9. Click on Download .csv and download the Access Key and Secret Key required for cluster configuration.

D. **Setting up Rancher to create Kubernetes Cluster**

1. Connect to your EC2 machine using the Public IPv4 DNS and Private Key file using the command: ssh -i .pem ubuntu@<publicIPv4DNS>
2. Run command after login: sudo apt-get update.
3. Install docker using sudo apt install docker.io
4. Verify docker using sudo docker-v
5. Start rancher using command: sudo docker run --privileged=true -d -- restart=unless-stopped -p 80:80 -p 443:443 rancher/rancher.
6. Verify rancher container is running using command sudo docker ps -a.
7. Copy the AMES ID and save it for later.

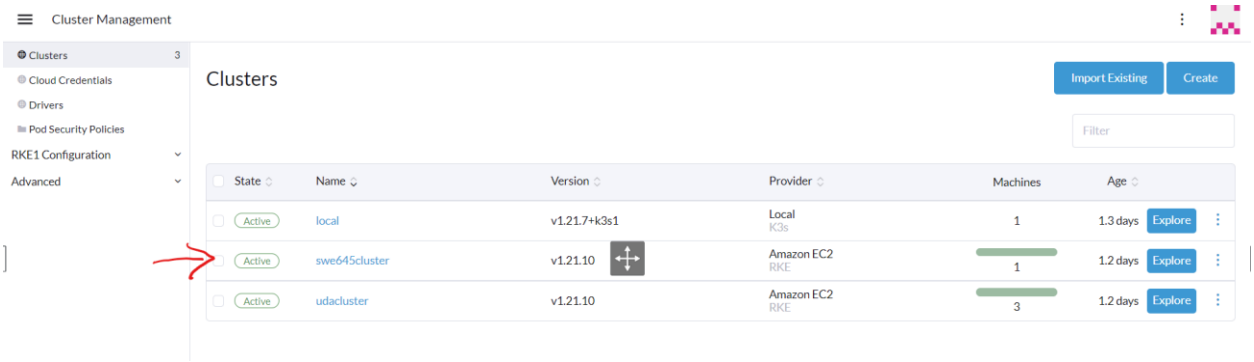
8. And go to your instance and click Public IPv4DNS.
9. After starting up rancher.It will take a couple of minutes for UI to come up.
10. To run Use command sudo docker logs <container Id>. Container Id is the AMES ID which we saved earlier.
11. Once the UI is up it will show a command for getting password sudo docker logs<container Id> 2>&1 | grep “Bootstrap Password:” .
12. Run the command as sudo user as we have installed docker as sudo user and it should output the password.
13. Paste the password in the UI and then click: Set the specific password to use.Save and generate the new password as you desire.
14. Click continue.
15. Navigate to cluster management on the left panel.
16. Click on Cloud Credentials and Click Create.
17. Give your Access Key and Secret Key from the earlier saved file.
18. Then select your “Amazon” as cloud provider.
19. Enter Name,Access Key, Secret key and select region as us-east-1.
20. Click create.
21. You should now see your credentials on the Cloud Credential Dashboard.
22. Then we will now navigate to Cluster Management and click on create cluster
23. Select Amazon EC2 and click next.



24. Then give cluster name and create node template with previous cloud credentials created.
25. Click create and dont change the VPC or security group.
26. In “IAM Profile Name” given AWS role name created.
27. To create IAM role:
28. Login to AWS console using created user.
29. Click Create Role
30. Select EC2.
31. Click on 'Generate Policy'
32. Give a policy name and paste the following in the JSON,

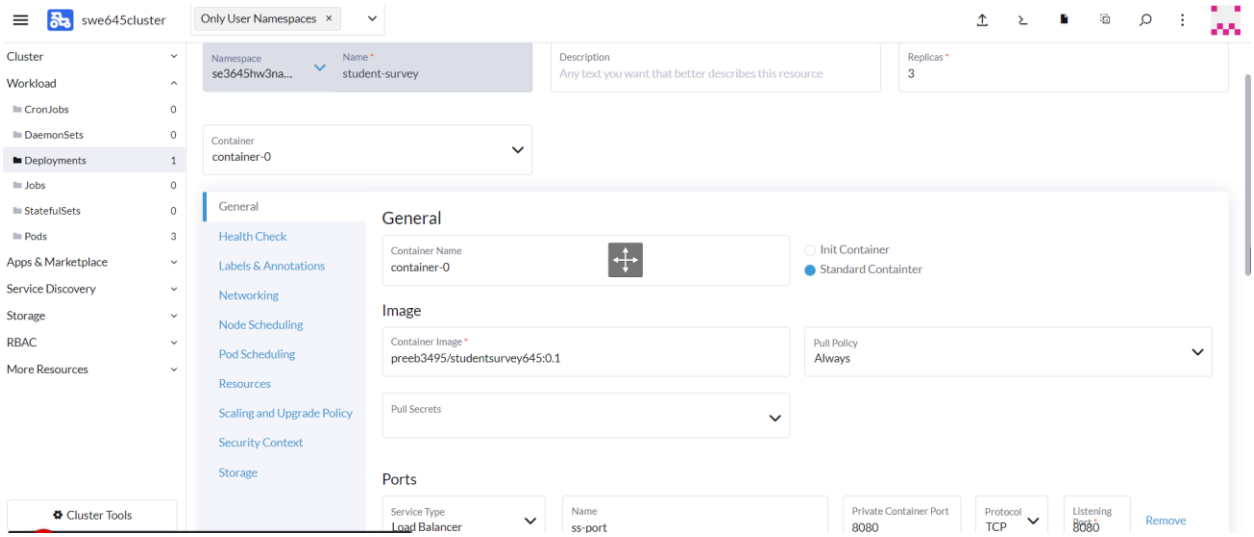
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```

33. Complete creating Policy and Role.
34. We created “swe645Role” role
35. We gave “swe645Role” as "IAM Instance Profile Name"
36. Then given name to template and click on create.
37. Given Name Prefix for cluster and selected etcd, Control Plane, Worker checkbox and click on create button.
38. In some time the cluster will be in an active state from the provision state.

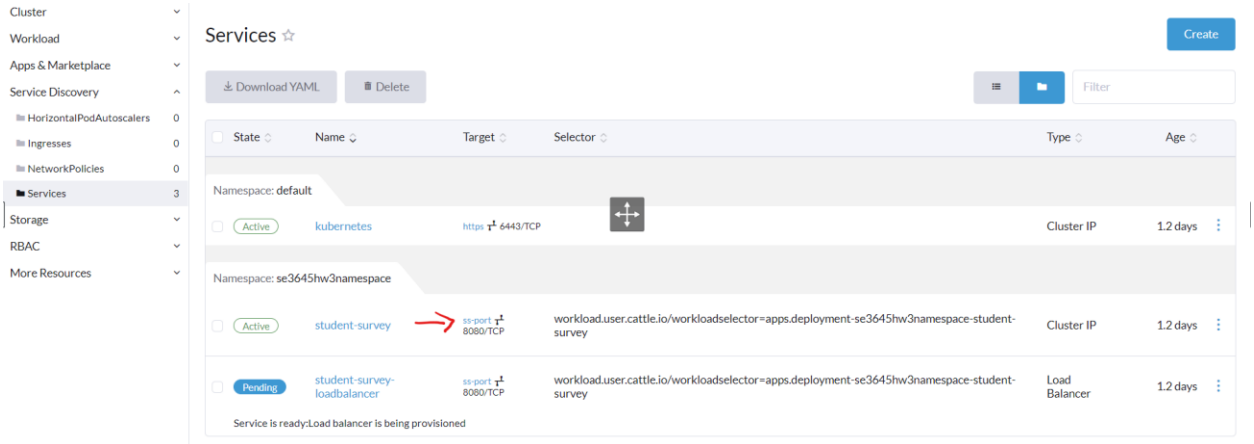


E. **Deploying Docker Image on Cluster.**

1. Once the created cluster in Active, click on created cluster on the left panel.
2. Click on Projects/Namespaces.
3. Create a Project with a unique name and in that project create a unique namespace.
4. After that expand ‘Workload’ on the left panel, under that we will get the “Deployment” option.
5. In deployments click on the create button.



6. Select the namespace we have created previously and give Name for deployment.
7. We selected replicas 3.
8. In container image given the Docker hub container image we pushed previously.
9. In ports added Load Balancer with name “ss-port” and Given Private container port- 8080, Protocol - tcp, Listening- 8080 as shown in above image.
10. Click on Create.
11. Once deployment is done, we will be able to see the pods on the left panel.
12. In the left panel, under Service Discovery.



13. We will be able to see our deployment with Active status. Click on the ss-port hyperlink. It will open a page with an 404 error.
14. Add /Survey_Form (War file name) to the URL .
15. We were able to see our Student Survey Form once we added “/Survey_Form” to end of the URL (hyperlink of ss-port).
16. Initial URL:: https://ec2-3-93-10-103.compute-1.amazonaws.com/k8s/clusters/c-rrmcq/api/v1/namespaces/se3645hw3namespace/services/http:student-survey:8080/proxy
17. Edited URL: https://ec2-3-93-10-103.compute-1.amazonaws.com/k8s/clusters/c-rrmcq/api/v1/namespaces/se3645hw3namespace/services/http:student-survey:8080/proxy/Survey_Form/

We were able to access the docker image Student Form from the rancher deployment :