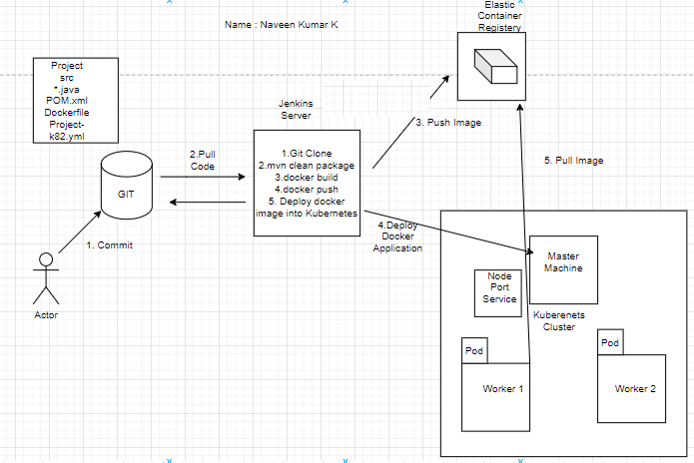
KPMG\_ASSESMENT

Challenge #1 Solution

===================

I have used <https://app.diagrams.net/> to create below illustration



1. Developer has written Java code and committed the code into GITHUB

2.We are using CI/CD server as Jenkins , Jenkins job will pull the code from the github. Once the code is available in jenkins servers, we are trying to compile the code and we are trying to generate Java artifact. Once the Java artifact is available we are creating the docker image.

Steps inside the CI/CD job

(i) Git Clone - Clones the code

(ii) We want to build the package - mvn clean package , which will generate the java artifact .

We are implementing dockerization as part of the application , We will be creating the docker image , from the docker image we will deploy our application as the containers in kubernets cluster

(iii) We will build docker image with the help of docker file , docker file will be part of source code .

(iv) Once the image is build , the image should be uploaded to artifact repository (AWS Elastic container register or Nexus) - docker push

Using Jenkins we doing continous intergration and continous delivery

We will be deploying into kuberenetes cluster (For eg : 1 Master , 2 worker machine)

(V) We will deploy the docker image into Kuberenetes cluster

3. From jenkins we will be interacting with kubernetes master machine , As part of the source we will be having the kubernetes manifest files

4. As part of the Project source code we will be having

\* Java code (source code)

\* pom.xml (build script)

\* Dockerfile (helps to build the docker image)

\* Project-k8s.yml (we will be defining how to deploy the application in kubernetes cluster)

5. Once the jenkins server started to connecting with kubernetes cluster, Based on the manifest files PODS will be created. In the manifest file we can mention the number of pods to be created . Inside the pods our application will be running

6. If we want to access the application from outside , we will be connecting with kubernetes cluster, we need to create a service (loadbalancer or node port).

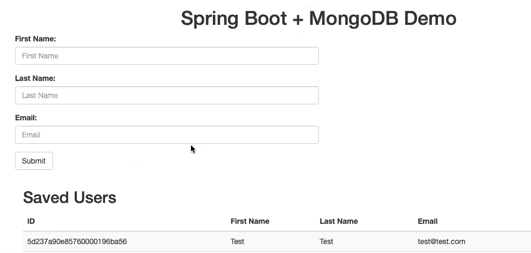
The request will go to service , the service will redirect to actual pod where our application is running and the end user will get response from the pod.

7. So in the Jenkins server we will be installing docker and Maven, Maven is required to compile the java code, build and create the java artifact. Docker should be installed because we need to create the docker images in the jenkins server and then trying to push the docker image to the repository.

8. The images will be download from the repository to the kubernetes cluster, after pulling the image it will create the POD

Kuberenets is container orchestration tool , it will takecare of POD lifecycle. If the POD has any issue or the application running inside the pod has any issue , the pod will be replaced with another pod , that is the role of kubernetes cluster

**Output:**



Challenge 2 # Solution

===================

Used Python programming

-------------------------------------

import boto3

aws\_mag\_con=boto3.session.Session(profile\_name="ec2user")

ec2\_con\_re=aws\_mag\_con.resource(service\_name="ec2",region\_name="ap-south-1")

f1={"Name": "instance-state-name", "Values":['running','stopped']}

f2={"Name": "instance-type","Values":['t2.micro']}

for each in ec2\_con\_re.instances.filter(Filter=[f1,f2]):

print(each)

**Output:**

