DevOps & Docker Containers

Overview:

One of our customer has decided to adopt digital transformation in to their technical landscape to reap the benefits of automation, low management cost and improvement in the agility. The idea is to transform their existing traditional web application into smaller microservices. The technical team has engaged in preparing the architecture and design for this transformation and the design should consist of:

- Application containerization (Web Tier, App tier, Database tier)
- Implement continuous delivery and continuous integration (CI/CD) pipeline for to automate application release process
- Stateful containers through docker storage techniques
- Load balancer and fail over of containers through container orchestration

Solution environment:

The application environment comprises of an application server, DB server as docker image s along with all necessary server configs, port openings, deployment details of sample web app, SQL schema, log output, etc. The docker image customization are done by writing docker files.

- 2 Linux servers for the kubernetes stack, one hosted as master node and the other as worker node along with docker installed in it
- A separate windows server is provisioned to setup DevOps engine
- Download any java based source code for the sample dummy web app
- Create the github account to maintain the source base of the sample web app
- Setup code build tool (ex: maven) to enable code build automation through jenkins
- Implement an IDE (ex: eclipse) and integrate it with the VCS (version control system) – Git / SVN
- Create the CI/CD pipeline with the integrations(Maven for code build, Jenkins for code pipeline platform)

Solution requirements:

The team has to demonstrate the sample three tier stateful web application in the kubernetes platform. Upon code commit in VCS (Git / SVN), the CI/CD pipeline needs to be triggered automatically from DevOps engine to build new source base and create new docker images using the docker files in Linux servers. Later, this new updated docker images should be deployed automatically in the kubernetes platform as containers (POD) with the help of pipeline with least downtime. The web application is expected to address:

- Perform the scaling
- Application rollback
- Application version history
- Application failover, etc. features of the new image in the kubernetes environment.

Solution task items:

The above transformation solution has various tasks needs to be performed across various technical areas.

Sl. No.	Task	Remarks
1	Create docker file for application server containers along with the sample web application war file	
2	Create docker file for the DB server with necessary schema	
3	Ensure appropriate DB connection string is given in the application source code	
4	Create the above 2 docker images	
5	Create docker containers using the images in the kubernetes environment	
6	Create the CI/CD pipeline in the jenkins devops engine along with necessary integrations (maven, git, remote kubernetes and or docker environment) Note: Use Kubernetes plugin in jenkins, which will automatically install dependency plugins	
7	Setup eclipse IDE and create project with the sample web application source code. If possible integrate the IDE with VCS system to perform code change, commit activities	
8	Ensure possible security aspects are implemented across all areas	
9	Upon code commit in VCS, new source base is built and is used to create new Docker image, which is then rolled out into containers in kubernetes environment	

Important points to remember

- You can use the public cloud(Free tier of AWS ,AZURE and Google cloud) for hosting the Linux and windows servers don't use the native kubernetes from public cloud
- Entire setup of kubernetes set up should be done in virtual machine for master and worker node
- CI/CD automation trigger should be done for the application deployment
- Kubernetes open source tool can be installed in the vm

POINTS TO REMEMBER

The below listed will be the upcoming events/ process next week onwards for Inframind Season 3 Round 2:

- For each tower, there will be a webinar session organized with the TCS SMEs to help you further understand the problem statement. the Webinar announcements/links will be shared via emails ,Campus commune channels and text messages.
- For further more details regarding Inframind Season 3, students can visit the “Inframind Season 3 “and “IT Infrastructure Services - Powering IT Infrastructure Globally - Be the Future!” channels on Campus Commune.
- Pls utilize the webinars to help enhance your understanding of the case study; your queries will also be addressed on this platform. Mentors will be assigned to you as required during these Webinars.