Docker and Kubernetes Case Studies

[Docker Case Study 1](#_Toc48848458)

[1. Setup A development Environment 1](#_Toc48848459)

[2. Setup a Docker Image 1](#_Toc48848460)

[3. Setup A test Environment 1](#_Toc48848461)

[4. Setup A Production Environment 1](#_Toc48848462)

[5. Setup A Docker Images Registry 1](#_Toc48848463)

[6. Backup DB files 1](#_Toc48848464)

[7. Setup a Cluster 1](#_Toc48848465)

[8. Test Application 1](#_Toc48848466)

[9. Production Application 1](#_Toc48848467)

[10. Verify & Validate Application 1](#_Toc48848468)

[Kubernetes Case Study 2](#_Toc48848469)

[1. Setup A development Environment 2](#_Toc48848470)

[2. Setup A test Environment 2](#_Toc48848471)

[3. Setup A Production Environment 2](#_Toc48848472)

[4. Setup the Replication Controller:- 2](#_Toc48848473)

[5. Blue Green Deployment 2](#_Toc48848474)

[6. Deploy Antivirus 2](#_Toc48848475)

[7. DB Backup 2](#_Toc48848476)

[8. Setup Dashboard :- 2](#_Toc48848477)

[9. Service Accounts:- 2](#_Toc48848478)

[10. Secret Management:- 2](#_Toc48848479)

# Docker Case Study

You are a DevOps Engineer and responsible to setup the infrastructure as per SDLC process. Below tasks are to be performed for an ERP Project which is to be deployed on docker containers in a cluster environment. Use appropriate technology to achieve below tasks efficiently.

1. Setup A development Environment: - In this case, as DevOps Engineer you need to create development machines for the development team. The development machines include webserver and database servers.
2. Setup a Docker Image:- Development team has finished their development and now the publish code they want to run it as Docker container so that it can be given to the test or operation team. You as a DevOps Engineer help them to create a Docker image with all the requirement which is provided by development team.
3. Setup A test Environment:- As per development machine configuration, a Test Environment is to be created, Testing team only need to run this docker image as container so that there should not be any infrastructure issue for them and they can test the application flawlessly.
4. Setup A Production Environment: -When the application is tested successfully then same docker image is to be used for the production environment.
5. Setup A Docker Images Registry: - The valid images should be stored under a central repository of the organization. The images which are in the repository is considered as the latest production images.
6. Backup DB files:- DB application should be running as a container but the data should be saved on the File Server’s DbBackup directory and there should a regular backup for these files so that there should not any data loss.
7. Setup a Cluster:- Setup the cluster where there are 3 Manager nodes and 12 worker nodes. Designate 2 servers as test web servers and 2 servers as test database servers and rest of the servers for production basis. Out of 8 remaining production server 5 reserve as webservers and 3 reserve for database servers.
8. Test Application:- Test the application is running with proper communication between webservers and the database server. The application should be running on designated test worker nodes only.
9. Production Application:- Test application should be running on production server with a given number of replicas. Minimum 2 web application replica is to be set for production.
10. Verify & Validate Application:- Check the container logs and application logs so that the application should be in running state.

# Kubernetes Case Study

You are a DevOps Engineer for Company ABC and the main responsibility of you to create an automation process using CI/CD pipe line. Below is the requirement for one the project which is to be containerized and orchestrated using Kubernetes so that you can fit it in Jenkins pipeline. You must create a production environment where there is a Manager node and 5 worker nodes. For testing you can create the similar environment. The pod network should be calico plugin.

1. Setup A development Environment: - In this case, as a DevOps Engineer you need to create development machines for the development team. The development machines include webserver and database servers. Pods which are to be created by development team should be used by any other team.
2. Setup A test Environment:- As per development machine configuration, a Test Environment is to be created, Testing team only need to run this docker image as container so that there should not be any infrastructure issue for them and they can test the application flawlessly. Pods which are to be created by testing team should be used by any other team.
3. Setup A Production Environment: -When the application is tested successfully then same docker image is to be used for the production environment. Pods which are to be created in production should be used by any other team.
4. Setup the Replication Controller:- Set autoscaling for web application on the basis of the load on worker nodes. If a node is having more than 80% utilization then no more pods are to be created on that worker node.
5. Blue Green Deployment:- Whenever there is change in the application during that time there should be zero down time for the client.
6. Deploy Antivirus :- On each and every worker node on the cluster should run the antivirus software. If a new node is added then this software should be automatically deployed on that machine. Do not deploy Antivirus on development machines.
7. DB Backup:- Take the backup of database application twice in a day, one at 6 AM and other is at 6PM. The backup should be running as a cron job so that there is no manual intervention for this activity.
8. Setup Dashboard :- Setup the Dashboard so that you can check that application or other components of deployments are running fine or not.
9. Service Accounts:- Create different service accounts for different teams (development, testing, operation) so that they will have restricted access to their environment
10. Secret Management:- Setup the confidential in secret management database so that if application needs it should be transferred in an encrypted way.