Deployment Process

As part of this task, I have taken the Azure VM to deploy the Docker compose file for 3 tier application which consists of Frontend as React (Sample Hello world app), Backend as Python ( sample API update request) and Database ( Postgres ) application.

Below are the steps followed to deploy the docker compose application.

1. Login to Azure Portal and Create VM with Ubuntu 24.04 image.
2. Install the docker and docker-compose application.
3. Clone the github repository to pull the code into the server.
4. Run the below commands to deploy the application.

Docker-compose up – This will validate the docker compose yml file and run the containers as a service.

1. We can validate the docker containers by using the below command.

Docker ps -a – to check the status of containers running inside the server.

Below are the steps to deploy docker compose application using CI/CD Pipeline.

In this document, I am taking Azure Devops tool to create the CI/CD Pipeline.

1. Push the code to Azure Repo.
2. Create the Build pipeline with the below stages.
3. Compilation of React code
4. Checking if all Unit test cases are executing successfully.
5. Check the Quality of code using sonarqube analysis.
6. Check the Quality gate of Sonar for the code quality analysis.
7. Publish the artifacts to azure artifacts.
8. Create Release pipeline with the below stages.
9. Configure Deployment pools to connect to Azure VM where we want to deploy the docker compose file.
10. Add the files from repository and Build pipeline artifacts to Release pipeline.
11. Add the docker stage to build the images and push to ACR.
12. Add the deploy stage to run the docker compose file to deploy the containers inside the server.
13. Add the stage to validate the containers running inside the server.

In the Code, For security reason i have mentioned the variables to connect other services. Variable values are stored in the Azure Keyvault. So as part of pipeline setup, I have integrated the keyvault with Azure Devops. This makes our deployment process more secure.

As part of the CI/CD process, we can also integrate the security tools like SAST and DAST for Code Security scanning and also we can integrate the Keyvault for storing the secrets for all sensitive information like Database information username, password, dbname, host url and backend configuration. This makes the setup more secure interms of code and infrastructure where we host the application.

As part of the Networking, we can enable the private endpoint and open firewall rules to secure the connectivity with any other services. Also we can enable the Service principals and Managed identity setup to secure commination with different services.

Also I am more strong expertise in Provisioning the Infrastructure with Terraform and other IAC tools.