Currency Exchange Portal

Duration	:	5 Weeks
Wizard	:	Moiz Mohammed & Amit Garg
Source	:	Finoptsys

Main Heading

Problem Statement:

To develop an automated CICD pipeline for Integration and deployment

Objective:

To build a pipeline that creates the images and deploys on to required environment.

Activity:

Stage 1: Account Setup

Associates have to have a working gitlab environment and access to the required modules.

Stage 2: Understand Docker and containerization.

Gain knowledge about Docker, containers, and containerization concepts. Docker allows you to create and manage lightweight, isolated environments known as containers.

Stage 3: Define your application's Docker file.

Create a Docker file that specifies the necessary steps to build your application image. This file typically includes instructions to install dependencies, copy files into the image, and define the runtime environment

Stage 4: Configure your GitLab pipeline.

In your GitLab repository, define a gitlab-ci.yml file to configure your pipeline. This file specifies the stages, jobs, and steps needed to build and deploy your Docker image. GitLab CI/CD supports declarative syntax to define these pipeline configurations.

Stage 4.1: Set up a GitLab Runner

Install and configure a GitLab Runner on a machine or cloud instance that will execute your pipeline jobs. The Runner listens for pipeline triggers and executes the defined jobs on the specified environment.

Stage 4.2: Configure your pipeline stages and jobs

Define the stages and jobs in your gitlab-ci.yml file. Common stages include building, testing, and deploying. Each job specifies the specific actions, commands, and scripts to be executed within that stage.

Stage 4.3: Build and push Docker image.

Configure a job in your pipeline to build your Docker image using the Docker file. Once built, push the image to a container registry or Docker registry to make it available for deployment.

Stage 4.4: Deploy the Docker image.

Define a deployment job in your pipeline to deploy the Docker image to your desired environment. This could involve deploying it to a container orchestration platform like Kubernetes or directly to a server using tools like Docker Compose.