**PE05: Programming Exercise**

Jenkins is an open-source automation server widely used for continuous integration and continuous deployment (CI/CD). It enables developers to automate the build, test, and deployment processes, ensuring that code changes are efficiently integrated and delivered. By integrating with Docker, Jenkins can build and deploy containerized applications, providing a consistent environment across different stages of the development lifecycle. Coupled with GitHub, Jenkins can automatically trigger builds and deployments in response to code changes, streamlining the development workflow and improving collaboration among team members. This combination of Jenkins, Docker, and GitHub enhances software development agility, reliability, and scalability.

First, we initiate the process by setting up the docker file and docker-compose. This involves defining the necessary configurations and dependencies for the Docker environment. Next, we launch the Jenkins server. We then added two credentials accounts, one for GitHub using SSH keys and the other for Docker Hub using username and password.

Next, we defined our CI/CD pipeline stages. We duplicated the stage of pushing the container image to the docker hub. We also changed the name of the image name.

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

We then built the job.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

The image gets pushed to the hub docker.

A screenshot of a computer

Description automatically generated