INT334 – Enterprise Application Automation

End-to-End CI/CD For Mern Stack Application

Synopsis Submitted in fulfillment for the requirements of the award of the degree

Of

"BACHELOR OF TECHNOLOGY"

"DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING"

By

Mummana Pavan Sai Kumar

12104904

KO308

Submitted to

Dr. Varsha Sahni

Lovely Professional University

Jalandhar (Punjab)

1. Brief Summary

This project involves the design and implementation of a comprehensive CI/CD pipeline to automate code integration, testing, and deployment for a microservices-based application. The pipeline ensures seamless application delivery, code quality maintenance, and real-time monitoring using cutting-edge DevOps tools.

2. Problem Statement & Solution

Problem Statement: Manual code deployments are error-prone, time-consuming, and lack consistency. Traditional deployment methods often result in increased downtime and reduced productivity.

Solution: The implemented CI/CD pipeline addresses these challenges by automating build, test, and deployment processes. It uses Jenkins for automation, Docker for containerization, and Kubernetes (EKS) for deployment management. Additionally, it enhances code quality through SonarQube analysis and ensures security using Trivy scans.

3. Technologies Used

- **Jenkins**: CI/CD automation
- **Maven**: Build automation and dependency management
- **Docker**: Containerization
- **Kubernetes (EKS)**: Container orchestration
- **SonarQube**: Code quality analysis
- **Trivy**: Container vulnerability scanning
- **ArgoCD**: GitOps-based deployment management
- **Slack/Email Notifications**: Real-time alerts

4. Expected Outcome

- A fully automated, containerized application running on AWS EKS
- Automated build, test, and deployment pipelines
- Improved code quality and security with SonarQube and Trivy
- Efficient monitoring and communication with Slack/email notifications
- Streamlined deployment management using ArgoCD

5. Objectives

- Design and configure a CI/CD pipeline using Jenkins.
- Implement build automation using Maven.
- Containerize the application using Docker for portability.
- Deploy and manage containers using AWS EKS.

- Perform static code analysis with SonarQube.
- Conduct security scanning with Trivy.
- Enable GitOps for automated deployments using ArgoCD.
- Set up real-time notifications for monitoring pipeline status.