

End-to-End CI/CD Pipeline with AWS Cloud Integration

One-Page DevOps Project Case Study | Jenkins • SonarQube • Docker • AWS • Git • Linux

Overview

Implemented a production-style CI/CD pipeline automating code integration, quality enforcement, and containerized builds. The project emphasizes real-world DevOps practices and cloud-based execution over isolated tool usage.

Problem → Solution

Manual deployments and missing quality checks cause unstable releases. This system automates builds, applies SonarQube Quality Gates, and ensures consistent Docker-based delivery using AWS infrastructure.

Tech Stack

CI/CD	Jenkins (Pipelines)
Quality	SonarQube + Gates
Build	Frontend & Backend
Containers	Docker
Cloud	AWS EC2
SCM	Git / GitHub
OS	Linux (Ubuntu)

Pipeline Workflow

- Git commit triggers Jenkins pipeline
- Automated frontend & backend builds
- Static analysis via SonarQube
- Quality Gate controls pipeline flow
- Docker image creation
- Execution on AWS EC2

AWS Integration

Jenkins and pipeline workloads run on AWS EC2, providing a cloud-native CI/CD environment. Security groups and IAM permissions were configured for secure access and execution.

Challenges & Debugging

Resolved Jenkins pipeline failures, SonarQube scan errors, Docker dependency conflicts, and AWS permission issues through detailed log analysis and configuration tuning.

Outcome & Skills Gained

- End-to-end automated CI/CD system
- Production-grade DevOps workflow
- Cloud-based CI/CD execution
- Strong debugging & pipeline optimization skills

Author: Aditya Sanjay Dakare | Role Target: DevOps / Cloud Engineer