

CI/CD

Continuous Integration and Continuous
Deployment

Overview

- **Continuous Integration (CI):** It's the practice of automating the integration of code changes from multiple contributors into a single software project. It allows developers to frequently merge code changes into a central repository where builds and tests are run.
- **Continuous Deployment (CD):** This is a software development approach in which every code change goes through the entire pipeline and is put into production automatically, resulting in many production deployments every day.

Benefits of CI/CD

- **Deploy to Production Without Manual Checks:** CI/CD pipelines eliminate the need for manual checks during the software development life cycle reducing the time to market for new features.
- **Catch Compile Errors After Merge:** New errors are detected as soon as the code is committed and pushed to the central repository which reduces the time spent by developers on issues introduced by new code.

Cont'd

- **Automate Infrastructure Creation:** Having the infrastructure scripted and stored in a central repository allows new infrastructure to be provisioned much quickly and with less human errors.
- **Faster and More Frequent Production Deployments:** Effective deployment strategies allow for new value-generating features to be released more quickly while receiving valuable feedback which can be incorporated in future releases.

Cont'd

- **Automated Rollback Triggered by Job Failure:** Quick undo when an error is detected in production allows a return to working state thus preventing a prolonged downtime.
- **Detect Security Vulnerabilities:** Early loophole detection helps in keeping production secure thus preventing additional costs that would arise in the event a loophole is detected in production.