Proposal to implement CI/CD into Udapeople

Sunil Mehta
Agile Coach

Agenda

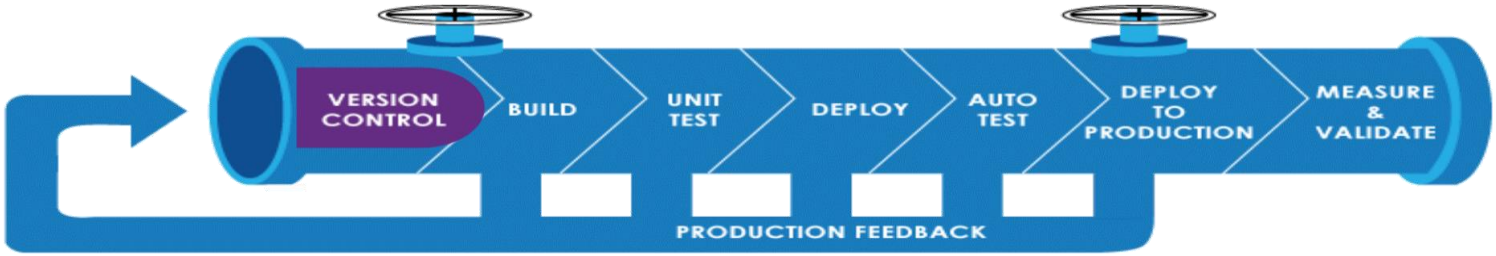
- Problem statement
- What is CI/CD
- Business Value
- Benefit of CI/CD

Problem Statement

There are several issues in current software delivery process which result into less customer satisfaction, less time to market value, less employee motivation and more cost to delivery the value. There are following major issues.

- Investing **more time** in a release cycle than delivering value
- **Manual** builds
- **Unhappy** clients
- **Slow** release process
- **Insufficient** testing & Visibility
- **Code gets lost** because of botched merges
- Deployments contribute to **schedule slip**
- **Friction** between ops and development departments
- *Deployments are **not cause for motivation** among team and stakeholders*

What is CI/CD



CI and CD stand for continuous integration and continuous delivery/continuous deployment. In very simple terms, CI is a modern software development practice in which incremental code changes are made frequently and reliably.

Continuous integration (CI) is practice that involves developers making small changes and checks to their code. Due to the scale of requirements and the number of steps involved, this process is automated to ensure that teams can build, test, and package their applications in a reliable and repeatable way. CI helps streamline code changes, thereby increasing time for developers to make changes and contribute to improved software.

Continuous delivery (CD) is the automated delivery of completed code to environments like testing and development. CD provides an automated and consistent way for code to be delivered to these environments.

Continuous deployment is the next step of continuous delivery. Every change that passes the automated tests is automatically placed in production, resulting in many production deployments.

Business Value

Technical Issues	Business Value	Description
Catch Compile Errors After Merge	Reduce Cost	Less developer time on issues from new developer code
Catch Unit Test Failures	Avoid Cost	Less bugs in production and less time in testing
Detect Security Vulnerabilities	Avoid Cost	Prevent embarrassing or costly security holes
Automate Infrastructure Creation	Avoid Cost	Less human error, Faster deployments
Automate Infrastructure Cleanup	Reduce Cost	Less infrastructure costs from unused resources
Faster and More Frequent Production Deployments	Increase Revenue	Value-generating features released more quickly
Deploy to Production Without Manual Checks	Increase Revenue	Less time to market
Automated Smoke Tests	Protect Revenue	Reduced downtime from a deploy-related crash or major bug
Automated Rollback Triggered by Job Failure	Protect Revenue	Quick undo to return production to working state

Benefit of CI/CD

- Removing Human/manual Error
- Deliver software with less risk
- Release new features more frequently (better time to market value)
- Better quality products
- Deliver the product that users need.
- Improve developer productivity