Transitioning to CI/CD

The UdaPeople Way

Daniel Nichols
UdaPeople Engineering Lead

Current State: Manual Deployments

- Current production deployments require a team of individuals to complete
  - Large changes would take a single engineer hours to complete
  - Even smaller changes would require at least two engineers for "second set of eyes"
  - Manually rolling back production from change failure requires several more hours
  - Costs of manual deployments means less deployments per year with less value added
- Manual methods are no longer consistent with today's industry standards
- "Doing more with less" is not feasible under traditional manual limitations
- Some systems have a single SME creating a bottleneck when not available

More time spent Building and Sustaining means less time adding True Value

Proposed State: CI/CD

- Definitions
  - Continuous Integration – merging working code several times a day to a shared mainline
  - Continuous Delivery – producing and releasing value in short cycles
  - Continuous Deployment – delivering frequent value from CI/CD through automation pipelines
  - DevOps – automate and integrate processes between development and operations teams
- Benefits
  - Increased collaboration between teams and stakeholders
  - Small incremental improvements deliver quick wins versus those that take months to deliver
  - Highly available environments with little or no downtime (with blue/green deployment models)
  - Less risk of implementation mistakes caused by manual processes
  - Built-in automated dev system testing reducing risk of production deployment failures
  - Built-in automated prod system testing to validate successful production deployments
  - Built-in automated roll-backs in case of system failure
  - Built-in automated system monitoring

CI/CD Value Framework

(Increase Revenue, Protect Revenue, Reduce or Avoid Costs)

Benefit	Value Added
Automation – catch compile errors on build	Reduces Cost as there is less time troubleshooting code issues
Automation – code scans for vulnerabilities	Avoid Cost from unpatched security threats
Automation – infrastructure creation	Avoid costs per man hour spent hour spent as well as additional time correcting human error
Automation – system development testing	Reduces cost of manual system tests, Increased revenue from "fail fast" enables finding/solving issues enabling quicker time to market working solutions
Automation – production deployments	Reduce cost from time spent deploying new features to production
Automation – smoke tests and rollbacks	Protect revenue by automating validation checks of new deployments and rolling back automatically in case of failure
Automation – Continuous Deployment Pipeline	Increase revenue with faster, more frequent deployments that can deliver incremental value and quick wins for udapeople

CI/CD Enablement

- Incorporate agile, DevOps methodologies into all UdaPeople engineering processes
- Build out automation using Git pipelines integrated with our system environments
- Implement version control over everything
- Enforce adoption of CI/CD principles to improve processes and build in quality
- Enforce targeting Minimal Viable Product to provide small, incremental value
- Ensure all teams are on board and accepting that everyone is responsible

If we can write it, we can Automate it