Udapeople

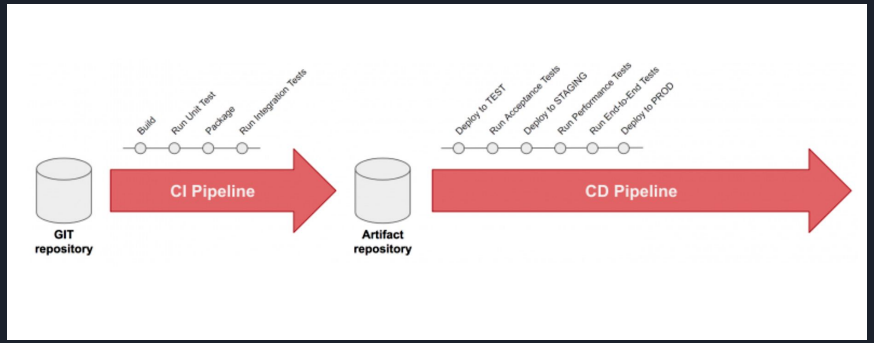
CI/CD Benefits

help small businesses care better for their most valuable resource: their people

Agenda

1. Continuous Integration, Deployment and Delivery
2. The release cycle today
3. Potential for improvement
4. The value proposition of CI/CD
5. CI/CD Best Practices
6. Benefits of CI/CD

Continuous Integration, Deployment and Delivery



Integration Phase:
- code is developed and stored in the source repository
- then it is compiled and tested
- … validated, checked for vulnerabilities

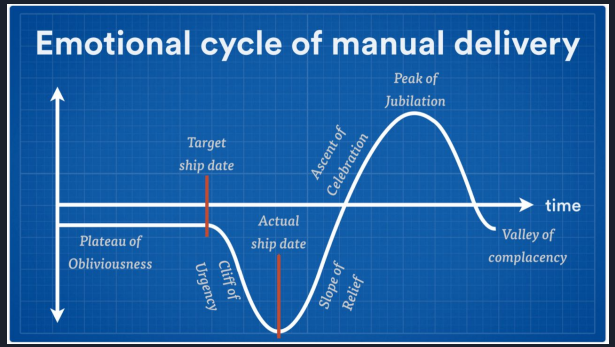
Deployment Phase:
- binary code is packaged
- deployed to the test environments
- tested for functionality / acceptance
- performance and penetration testing

… ALL fully automated (or so we would like it to be)

Last step: deploy to production (also automated process)

The Release Cycle Today

- Release process today: mostly manual
- Typical Lead Time: 3-6 months
- Most of the time spent during manual (re-)testing
- Lots of resources spent on fixing last minute issues
- Little time left for testing automation
- Huge costs with development and testing
- Lost opportunities for including the Feedback of the stakeholders



https://www.iamsensoria.com/2019/10/emotional-cycle-of-manual-delivery.html

Potential for Improvement

1. Reduce errors caused by manual deployments by automating and testing them multiple times in different production-like environments
2. Decrease the time spent for deploying the software by relying on standard automated deployment processes
3. Manual deployments offer poor visibility into the exact steps and require elevated human-access to mission-critical systems: solution - automate the deployment process
4. Current deployments are sometimes left broken for days/weeks leading to an increased amount of time spent to discover the cause of the issue: solution - use smoke tests to check the deployment combined with automatic rollbacks: no more broken deployments
5. Too much time is spent doing manual testing: solution - automate the acceptance tests, leaving time to implement also automated performance tests

The Value Proposition of CI/CD

Increases Development Velocity: reduced lead times

Increases Productivity: better motivation for the development teams

Increases stakeholder feedback: faster reaction time and better adaptability to new requirements

Fewer late releases: on time and on budget (and less frustration for operations' colleagues)

Fewer unknown bugs in production: reduced maintenance costs

Less human errors during deployments: reduces costs

Faster releases -> Less time to market -> Increases Revenue

CI/CD Best Practices

1. Automate everything: no more manual configuration changes

   Invest time into automation, with great benefits after the first release cycle

2. Keep configuration in code

   No more hidden knowledge on deployments, better documentation

3. Fail fast: rapid feedback on failed deployments

   Give developers instant feedback on their deployments and a good chance to fix their mistakes

4. Only way to deploy: through automation

   Eliminates the need for error-prone manual deployments increasing the chances for a successful production deployment

Benefits of CI/CD

Reduced Costs
- fewer undetected bugs
- less time spent in analyzing broken deployments

Avoids costs
- prevent security holes through automated vulnerability scanning
- less human errors through increased infrastructure automation

Increase Revenue
- shorter lead times: more frequent deployments
- less time to market: automated testing