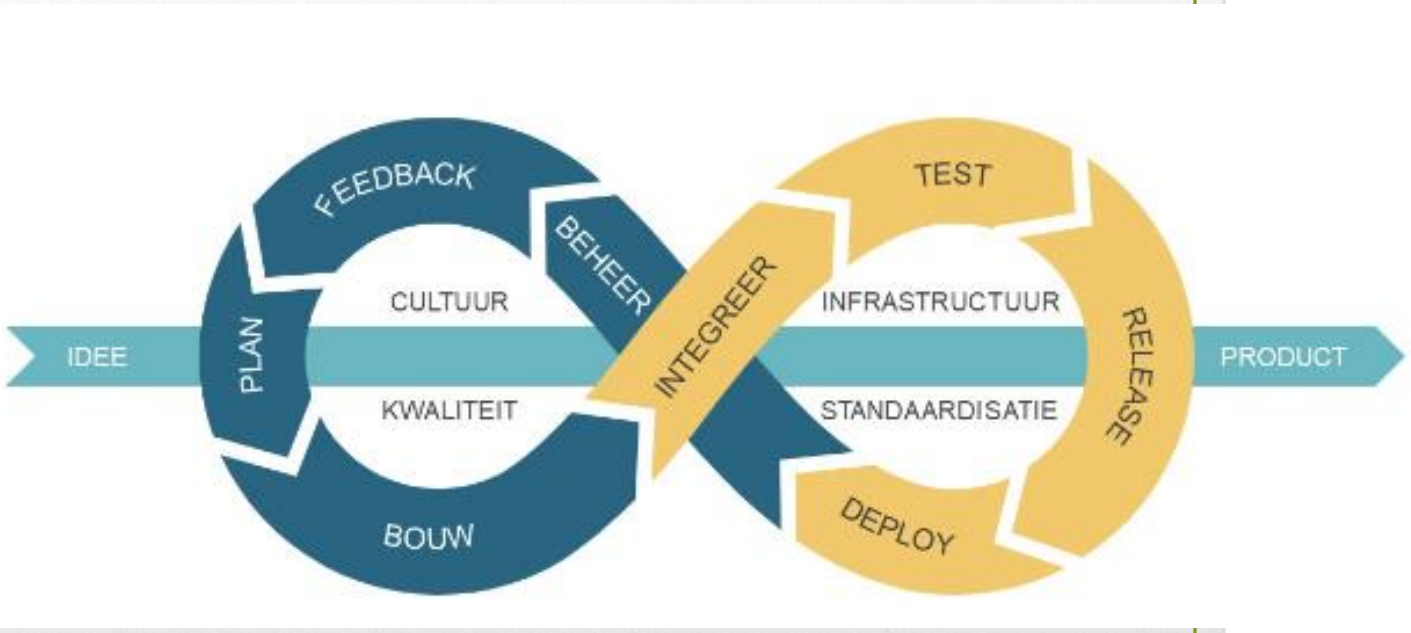
CI/CD


IDEE
PLAN
FEEDBACK
BEHEER
CULTUUR
KWALITEIT
BOUW
INTEGREER
TEST
INFRASTRUCTUUR
STANDAARDISATIE
DEPLOY
RELEASE
PRODUCT

Why Change IT Methodology?

- We have three objectives

Response Quickly to market pressure

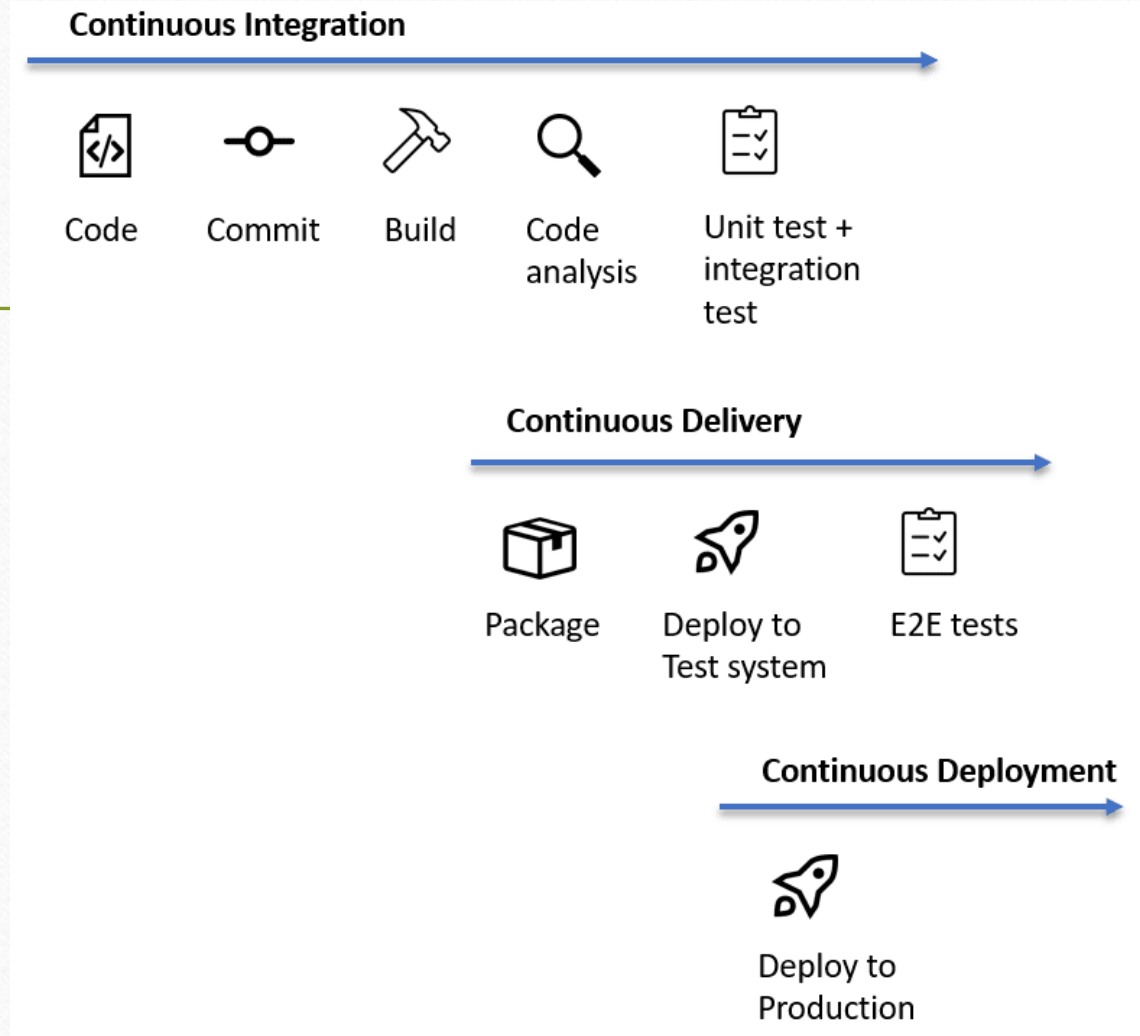
Continues customer satisfaction

Sharpener competitive Edge

What's the Benefit from business perspective

- Increase Revenue
  - This can be achieved by fast response to customer requirement (Fast releases)
- Reduce cost
  - One of things increase cost is time by CI/CD we save a lot of time
  - Automated Infrastructure clean up
- Protect revenue
  - Automated smock test
- Avoid cost
  - Catch failure in a short time

CI/CD Phases


Continuous Integration
Code
Commit
Build
Code analysis
Unit test + integration test
Continuous Delivery
Package
Deploy to Test system
E2E tests
Continuous Deployment
Deploy to Production

Best Practices for CI/CD:

- **Fail Fast**
  - Set up your CI/CD pipeline to find and reveal failures as fast as possible. The faster you can bring your code failures to light, the faster you can fix them.
- **Measure Quality**
  - Measure your code quality so that you can see the positive effects of your improvement work (or the negative effects of technical debt).
- **Only Road to Production**
  - Once CI/CD is deploying to production on your behalf, it must be the only way to deploy. Any other person or process that meddles with production after CI/CD is running will inevitably cause CI/CD to become inconsistent and fail.
- **Maximum Automation**
  - If it can be automated, automate it. This will only improve your process!
- **Config in Code**
  - All configuration code must be in code and versioned alongside your production code. This includes the CI/CD configuration files!