

Agile Enablers

Steve Berczuk
Oct 18, 2011

steve@berczuk.com

<http://www.berczuk.com>

<http://www.berczuk.com/blog>

© 2011 Steve Berczuk

Agenda

- Agile Overview
 - Why, What, Key Practices
- QA and Release Engineering
 - Role in Agile Adoption
- Agile QA
 - QAs Evolving Role

© 2011 Steve Berczuk

Steve Berczuk



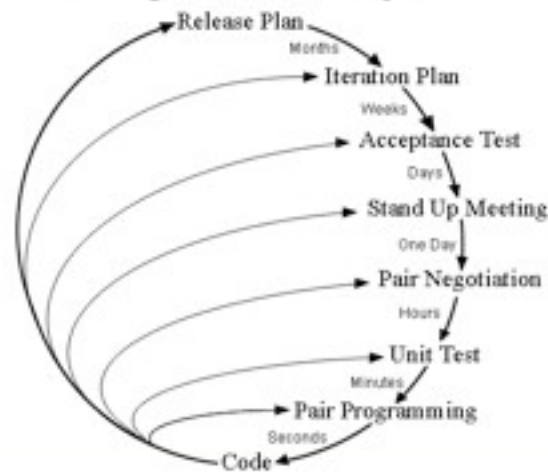
<http://www.berczuk.com>
steveberczuk.blogspot.com



Copyright © 2005, Mountain Goat Software

Agile

Planning/Feedback Loops



Don Wells
Creative Commons Lic.
from Wikipedia

XP

Agile Manifesto

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

© 2011 Steve Berczuk

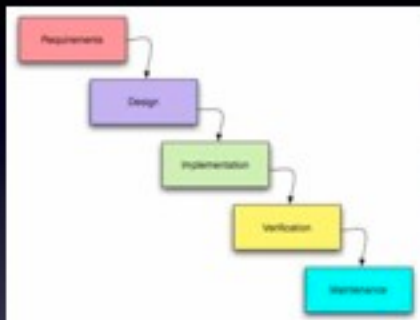
Agile is...



(Frequent) Feedback



Traditional SW Dev



Testing at the End

Agile

- Feedback
- Tracking
- Enabling Infrastructure
- Enabling Processes

© 2011 Steve Berczuk



Tracking

Supporting Functions

- **Build & Deploy**
 - DevOps
- SCM
- QA
- Retrospectives

© 2011 Steve Berczuk

Agile Practices

- **(Always) Working Software**
- **Generalizing Specialists**
- **Unit Testing**
- **Time Boxes**
- **Tracking (Burn Down)**
- **Sprint Reviews**

© 2011 Steve Berczuk

Agile Enablers

- End to End Features
- Measurable Goals
- Incremental Approach. (YAGNI)
- Feedback (working software)

© 2011 Steve Berczuk

QA Roles

- Testing Code: Identify Impediments
- Testing Requirements: Validate Backlog
- Testing “Hard Stuff”:
 - Automate
 - Manual Exploratory Testing

© 2011 Steve Berczuk

QA Removes Impediments

Verify that Code is Deployable
Discover gaps in developer testing



Measurable Goals

- Verify that Stories are Testable
 - Definition of Done
 - Reasonable Size
 - Reasonable Scope
- Discover gaps in requirements

QA Roles

Traditional

- Test
- Report
- Retroactive

Agile

- QA Developer Tests
- QA Requirements
- Test “hard stuff”

© 2011 Steve Berczuk

Working Software

- Buildable
- Deployable
- Passes Automated Tests
- Does “something”

© 2011 Steve Berczuk

Agile SCM

- Fewer Code lines
- Tests, not branches for stability
- Enables change

© 2011 Steve Berczuk

Agile Build

- Continuous, not Nightly
- Build includes Test
- Automated Deploys

© 2011 Steve Berczuk

Collaboration

- Dev Ops
- Cross Functional Teams
- Communication

© 2011 Steve Berczuk

Summary

- Agile = Quality Everywhere, All the Time
- Define Done
- QA Early
- QA Often

© 2011 Steve Berczuk



Questions