Showstoppers for Continuous Delivery in Small Scale Projects

By Jakob Svemar
Content

• Context & Problem Domain
• Discussion contents
• Summary
Context

• Small Scale Projects
• Different Customers
• Initial Problem
  • ”How do we know when we are done testing?”
  • ”We find bugs too late in development cycle.”
Problem Domain

Lack of documentation

"We find bugs too late in the development cycle"

Waterfall-like Development

"How do we know when we are done testing?"

Manual testing

Ad Hoc Testing
Discussion topics

• Root Causes found:
  • Developers do not see an advantage with automating unit and integration testing
  • Quality Assurance is tied to Releases
  • Improving development practices are not prioritised
Developers do not see an advantage with automating unit and integration testing

- Heavy focus on manual testing
- Undocumented testing
- No need to frequently verify functionality
- GUI testing prioritised

Discussion: With automated testing being a big part of CoDe, what is perceived as most important, unit, system, gui testing?
Quality Assurance is tied to releases

- Bugs found late in development
- Testing as an afterthought
- Bugs slip through to the customer
- Acceptance tests have no clear link to original stories
- Discussion: What is a configuration manager’s role when trying to improve development?
Improving development practices is not prioritised

- Focus is on things that can be delivered to the customer
- Most improvements give benefit over time
- Unclear who is responsible
- Discussion: What should come first, technical CoDe implementation or organisational changes?
My solutions

• Improving Development

• Making Continuous Delivery Viable
Improving Development

• Create an environment where small feedback cycles are used

• Creating test cases is part of development

• involve customer

• Ensure the product is stable enough to show to the customer

• Use automation to test often and cheaply
Making Continuous Delivery Viable

• One project may not be long enough to both implement and benefit from changes

• Focus on improvements that can persist between projects

• Have someone responsible for working on improvements
Summary

• Overall recommendations:
  
  • Start automate testing on a smaller scale, unit and integration before GUI
  
  • Have the customer included in the validation process early on, not as a last stage
  
  • Set up clear goals for how development should be improved