Testing in Agile Projects – Case Study

Armands Baranovskis
Sanda Linde

TietoEnatorAlise

We are Agile - we don't need testing!

Agile Principles

- Incremental development
  - Delivering a series of (useful, usable) deliveries
- Iterative development
  - Allowing requirements to evolve and be added to
  - Responding quickly to new business priorities
- People-oriented development
  - People's qualities rather than heavy process
  - Allowing the agile team to manage themselves
  - Daily interactions between developers and the business
- Technical excellence and built-in quality
  - Personal responsibility, craftsmanship and simplicity
- Lean approach
  - Measure progress by delivered product
  - Minimize waste

Case 1

Project description	Type: maintenance Number of people: ~20 Agile for 2 years

Organization	• Small number of automated unit tests
	• Manual testing:
	  • I phase – testing of changes (including bug fixes)
	  • II phase – regression tests of critical functionality

Client involvement	Does not adapt agile practices. Testing is carried out independently.

Case 2

Project description	Type: development Number of people: ~12 Agile for 2 years
Organization	Testing begins after development sprint • I phase – testing of changes (including bug fixes) • II phase – regression tests
Client involvement	Client tests only critical changes

Case 3

Project description	Type: implementation Number of people: 5 Agile since the beginning
Organization	Testing is separated from other activities, but is rather a part of every activity: • Functionality is checked after every change for errors • Functionality is checked for accordance with user stories
Client involvement	Client does the initial acceptance during on site demo

Case 4

Project description	Type: development Number of people: 3 Agile since the beginning
Organization	Product quality is verified through: • Test driven development • Automated unit tests • Automated integration tests • Manual testing according to user stories
Client involvement	Specific business case testing after every delivery Regression testing after each major delivery

We are Agile – we don’t need testing!

Do we?

How agile does it?

Case 1	• Automated unit tests • Manual change testing • Manual regression tests
Case 2	• Manual change testing • Manual regression tests
Case 3	• Manual change testing • Manual testing according to user stories • High client involvement in testing process
Case 4	• Automated unit tests • Automated integration tests • Manual testing according to user stories

Agile versus Waterfall

Agile	**Waterfall**
Testing guidelines	Testing plan
Continuous integration	Verification at the end
Much greater need for testing automation	Automated testing is seldom effective
Small amount of documentation => exploratory testing	Large amounts of detailed documentation
Team does the testing	Tester is often detached from the rest of the team
Users (clients) are continuously involved	Users (clients) are involved only during analysis and acceptance phases

Benefits

- Quality of the product is insured at the end of every iteration
- All team members gain significant business knowledge
- All team members are multifunctional
- Both, project team and client, have more impact on the product and the outcome
- Satisfied client due to transparency and trust
- Less is more

Problems

- Historical stratification among roles
- Agile is especially hard for testers:
  - Why should I test something that isn't ready?
  - Why should I do someone else's job?
  - How can I test without documentation?
- Agile demands high individual and team discipline and initiatives

Testing is a function, not a role.

Thank you.

Armands Baranovskis
Sanda Linde

TietoEnator Alise
www.alise.lv