



DataArt

Testing in Agile vs Waterfall

Test results analysis. Reporting.

Natalia Yakhina
Senior QA, QA Lead at DataArt
Author of @qa_career_up



Lecture roadmap

Part 1: Testing in Agile vs Waterfall

- SDLC: Software Development Life Cycle
- Software Development Methodologies
- Testing in Agile
- Testing in Waterfall

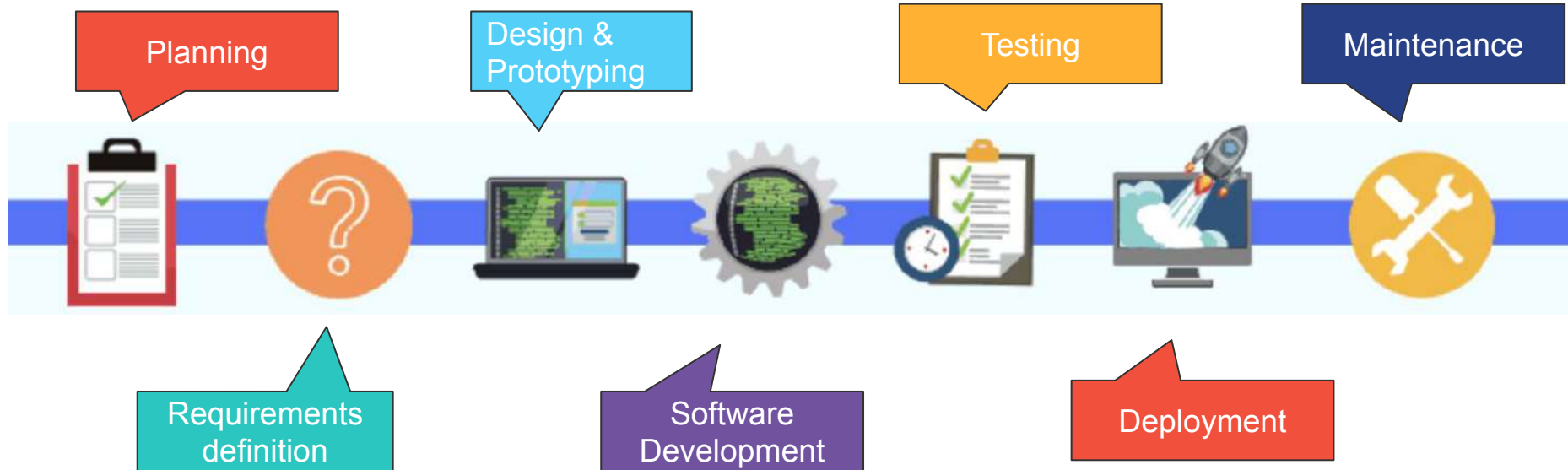
Part 2: Test results analysis. Reporting.

- Test Monitoring and Control: Why?
- Metrics Used in Testing
- Test progress report
- Test summary report
- Reporting in Agile vs Waterfall
- Reporting Audiences

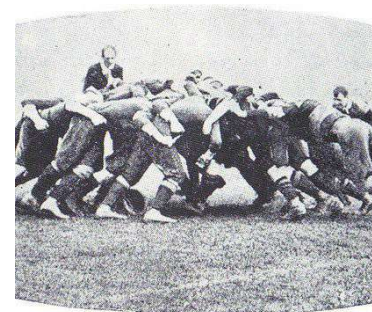
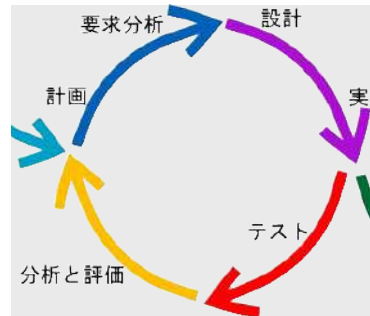
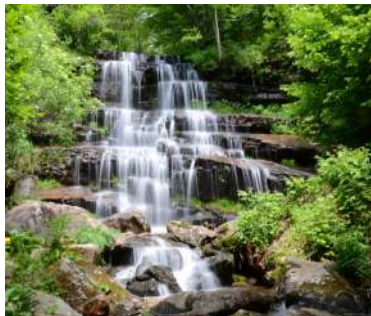


Part 1: Testing in Agile vs Waterfall

SDLC: Software Development Life Cycle

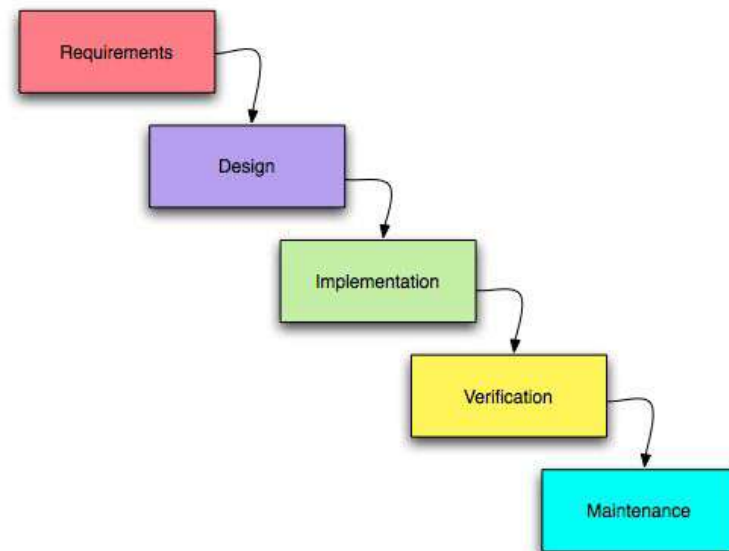


Software Development Methodologies



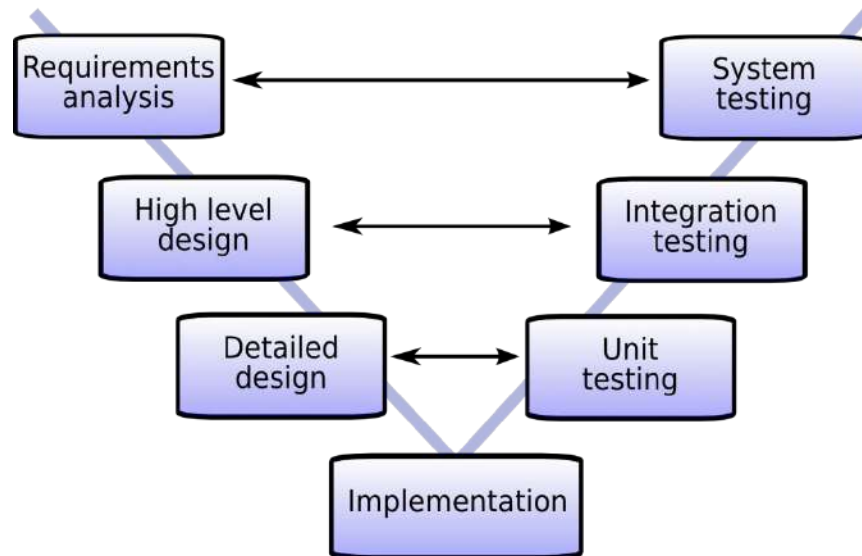
Waterfall model

- One phase must finish before another can begin;
- Limited in speed;
- Predictable, well-documented;
- Is used with:
 - fixed timelines;
 - fixed requirements;
 - high cost of risks.



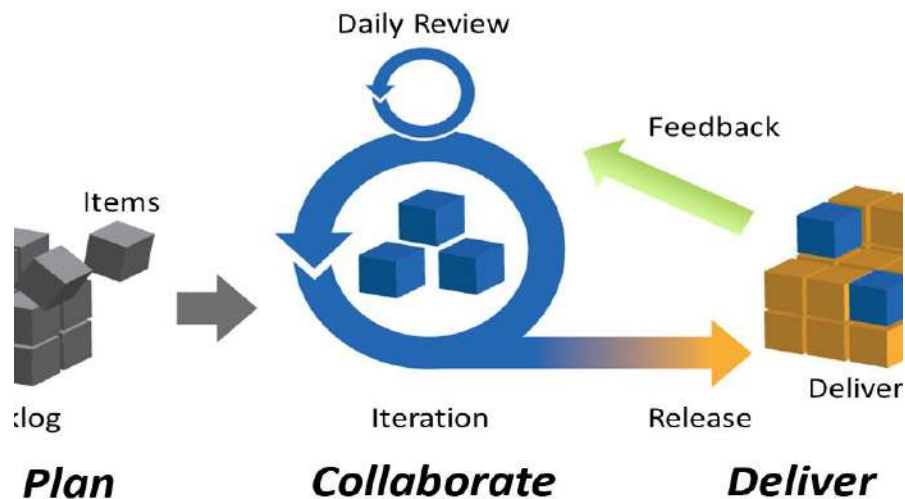
Testing in Waterfall

- Test Planning phase required;
- Requirements testing required;
- Test design and development in advance;
- Testing is after development finish;



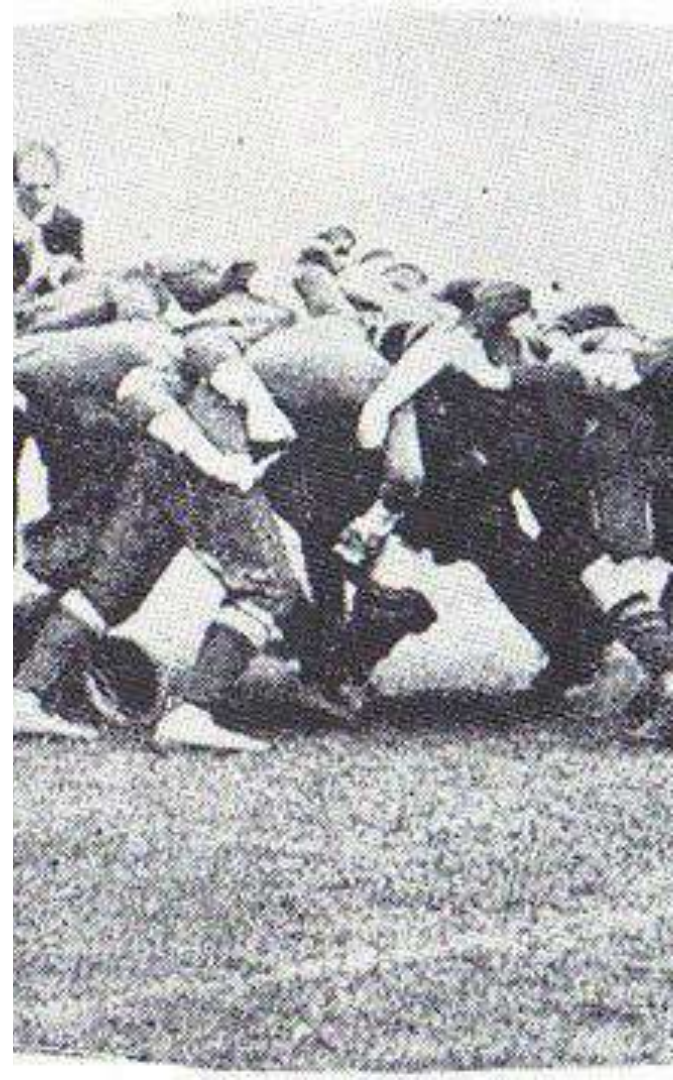
Agile approach

- Family of methodologies
 - Iterative
 - Dynamic requirements
 - Close team collaboration
 - Poor documentation
-
- Is used when:
 - No clear requirements
 - No fixed timelines
 - work in small groups



Agile: Scrum

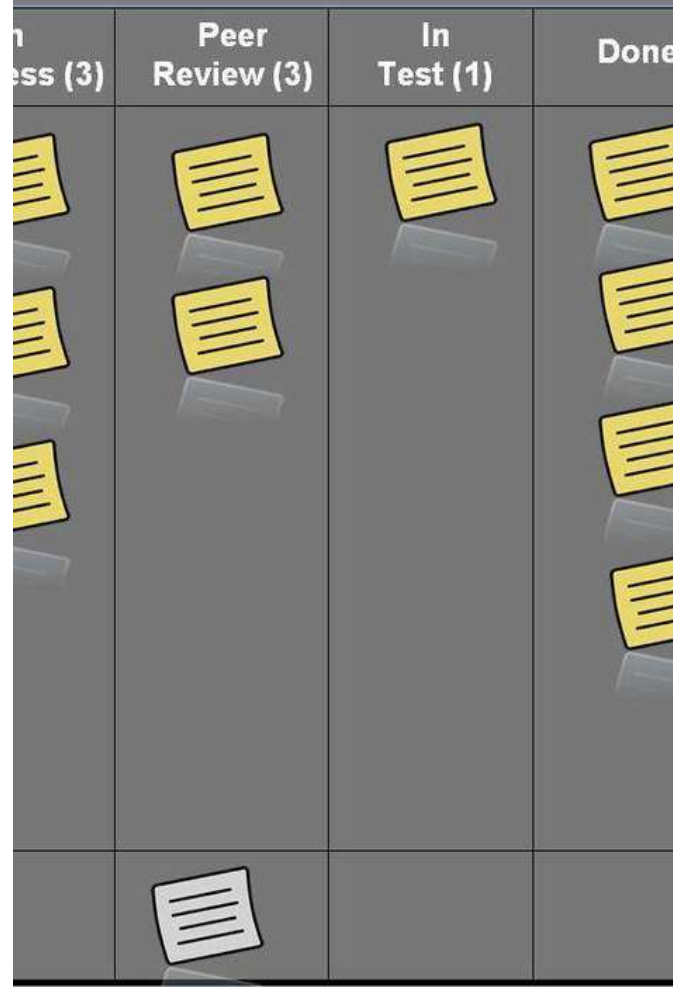
- 2-4 weeks sprints
- Sprint planning, poker planning
- Daily status meetings
- Demo
- Retrospective
- Result after each sprint
- Focus on the process



Agile: Kanban

- Focus on tasks
- Work based on priorities
- Fixed number of tasks in progress
- High time efficiency

Example of a Kanban Board



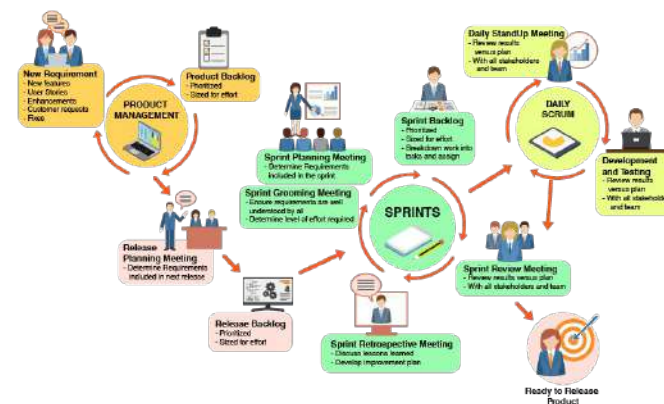
Agile: Extreme programming

- Test driven development
- Pair programming
- Review process
- No testers -> possible non functional problems



Testing in Agile

- Constant collaboration with development team
- Constant requirements clarification
- QA takes over BA role
- No detailed test plan
- Possible use of check lists instead of test cases
- Incremental testing



Testing in Agile vs Waterfall

AGILE

- Poor documentation
- Requirements may be changed on any phase
- No Test Plan
- Lots of communications
- Testing can be performed in parallel with the development



WATERFALL

- Well documented
- Requirements are defined before development start
- Test planning phase is required
- Formal communications
- Testing starts after development finish



Questions