Agile Testing Process

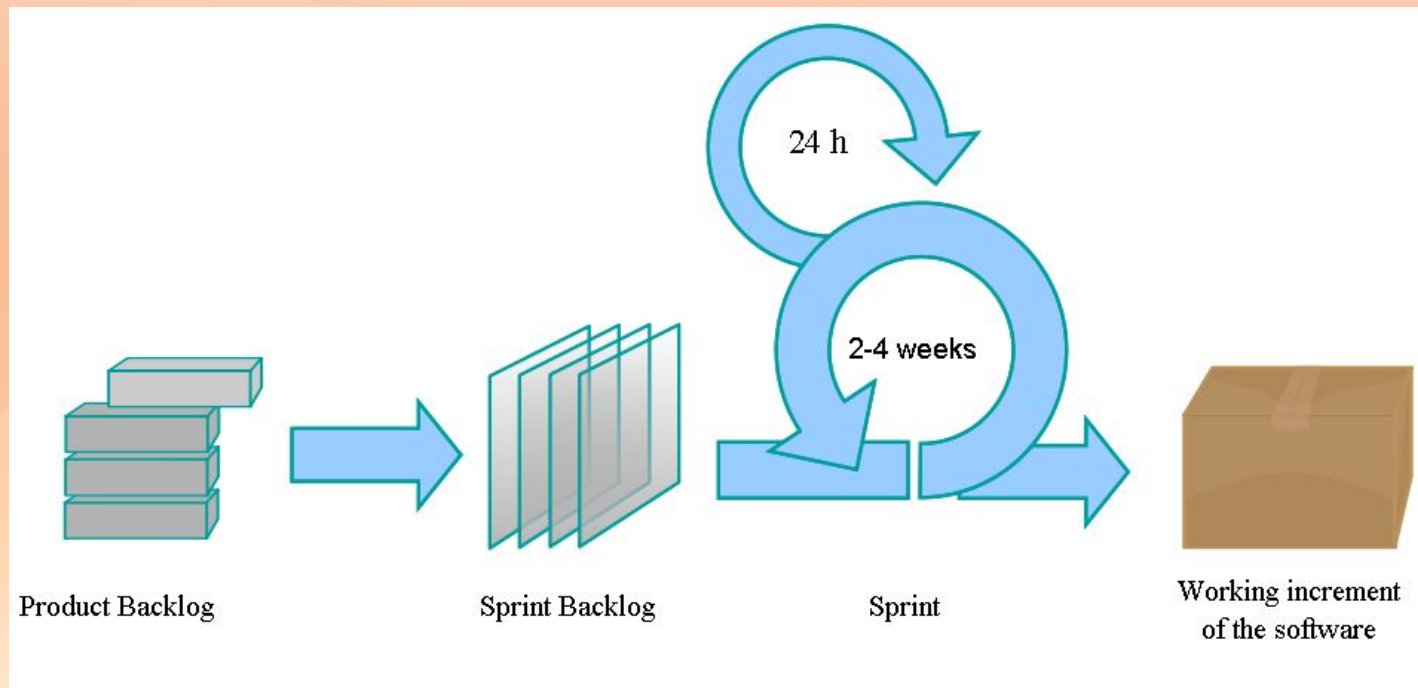## Process Overview

Software development methodology

- Agile – iterative development methodology, where requirements evolve through collaboration between the customer and self-organizing teams. Agile business approach aligns development with customer needs.

Product Backlog   Sprint Backlog   Sprint   Working increment of the software

24 h

2-4 weeks

Wrong ways of using a tester

- A tester is not the one who can not program and causes breaks only.

- Inadequate use of the person's potential and experience.

- A tester does not participate in planning.

- A tester's work starts after the functional development.

Why teams switch to Agile

- Simplicity of principles and apparent easiness of their employment.
- More transparent communication with the customer and accurate planning.
- Agile is in fashion.
- Examples of other teams.

Problems of switching to Agile

- Difficulty of leaving the traditional understanding of roles.

- Resistance to changes.

- In the adaptation phase the basic Agile principles may be destroyed.

- Agile is not a panacea to solve all the problems. Agile principles just reveal the problems, and it is people who are to solve them.

Adequate approach to a tester

- A tester is a team member!
- The tester's experience in development should be fully used.
- The tester's functions are largely determined by the tester himself.
- Participation in planning and requirements analysis.

Changes in testing

- Start of a sprint – start of testing.
- Testing activities are the same as in other methodologies.
- Testing tasks – usual sprint tasks.
- All the team is responsible for the result, including quality.
- Testing tasks – common tasks of all the team.
- Maximum communication.
- Provide continuous feedback.
  - To the customer.
  - To the programmers.

Types of testing

- Test planning.
- Build acceptance testing.
- Functional testing.
- Regression testing.
- Demo testing.
- Test automation.

Off-testing time

- Analysis of requirements and risks; test planning.
- Test automation.
- Setting up test environments.
- Preparing the necessary documentation.
- Work with external resources.
- Providing help to developers.

Problems with testing shortage

- Lack of testing resources.
- Inefficient test planning.
- A lot of manual testing.
- Inefficient testing process.
- Increasing functional for regression testing.

Ways of problem solving

- Detailed test planning.
- Automation, possibly with the developers' help.
- Increased unit test code coverage.
- Implementation of stabilization sprints.
- Increase in the amount of testing resources.

- Less risk of compressed test period.
- Test all the time, not just at the end.
- Work together as one team towards a common goal.