

The Agile space – key frameworks and concepts

Agile Manifesto

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

Agile Practices

- Definition of done
- Backlog
- Iterations
- Incremental development
- Acceptance testing
- Continuous integration
- Continuous deployment
- Collective ownership
- Info radiators
- Daily meetings
- Sustainable pace
- Collocation

Other key Agile concepts

- User stories
- Affinity, wide band estimating, planning Poker etc.
- Kanban board
- Contracting for Agile

Agile Principles

- Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- Business people and developers must work together daily throughout the project.
- Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- Working software is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- Continuous attention to technical excellence and good design enhances agility.
- Simplicity — the art of maximizing the amount of work not done — is essential.
- The best architectures, requirements, and designs emerge from self-organizing teams.
- At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Scrum Pillars

- Visibility
- Inspection
- Adaptation

Scrum Framework

Roles

- Product Owner
- Scrum Master
- Development team

Artifacts

- Product backlog
- Sprint backlog
- Definition of Done

Events / ceremonies

- Sprint
- Sprint retrospective
- Daily Scrum Meeting
- Sprint review meeting
- Sprint planning meeting

LEAN SW Development

Waste

- Partially done work
- Extra process
- Extra features
- Task switching
- Waiting
- Motion
- Defects

Principles

- Eliminate waste
- Empower the team
- Deliver fast
- Optimize the whole
- Build quality in
- Defer decisions
- Amplify learning

Extreme Programming (XP) core values

- Simplicity
- Communication
- Feedback (failing fast)
- Courage

XP Practices

- Whole team
- Planning games
- Small releases
- Customer tests
- Code standards
- Sustainable pace
- Metaphor
- Continuous integration
- Test-driven development
- Refactoring
- Simple design
- Pair programming

Kanban Development

- Visualize workflow
- Limit Work in Progress
- Manage flow
- Make process policies explicit
- Improve collaboratively