

# Agile Model

## 1 Manifesto

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

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

## 2 Principles

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

## 3 User stories

### Definition: User stories

User stories are part of an agile approach that helps shift the focus from writing about requirements to talking about them. All agile user stories include a written sentence or two and, more importantly, a series of conversations about the desired functionality

- Captures the spirit
- Ignores details
- Make sense to customer
- Delivers value to customer

- End to end (full stack)
- Independent
- Testable
- Small (1-5 days) so easy to estimate

### 3.1 Behaviour Driven Development

**Definition: Behaviour Driven Development**

An agile process what supports and encourages collaborative development

Built on TDD (Test Driven Development) and ATDD (Acceptance TDD), plus:

- Where to start in the process
- What to test and what not to test
- How much to test in one go
- What to call the tests
- How to understand why a test fails