|  |
| --- |
| **What is Agile?** |
| **Introduction** The word ‘agile’ means ‘to quickly change direction’. The agile methodology refers to a group of software development methodologies that promote development, iterations, open collaboration and process adaptability throughout the lifecycle of a project.  There are many agile development methods and most minimise risk by developing software in multiple repetitions (or 'iterations') of short timeframes (known as timeboxes). Software developed during one unit of time is referred to as an iteration, which typically lasts from two weeks to four weeks.  Each iteration passes through a full software development cycle, including planning, requirement analysis, designing, writing unit tests and coding until the unit tests are passed and a working product is finally demonstrated to the stakeholders.  https://evs01prd03.evalueserve.com/TMS/content_pics/images/neeta.jain/Agile%20Methodology.gif   * At the end of each iteration, stakeholders re-evaluate project priorities with a view to optimise their return on investment. * Agile methods emphasise face-to-face communication over written documents.\* * The team composition in an agile project is usually cross-functional and self-organising, without consideration for any existing corporate hierarchies or corporate roles of team members. * Each agile team will have a customer representative. * The progress of software development relies on the current state of the working software. |
| The following are a few principles behind the Agile Manifesto:   * Customer satisfaction through a rapid and continuous delivery of useful software * Working software delivered frequently (the concept of interim deliverables) * Working software becomes the principal measure of progress * Late changes in requirements accepted * Close, daily cooperation between developers and those involved in business * Face-to-face conversation: The best form of communication (co-location) * Projects built around motivated individuals who should be trusted * Continuous attention to technical excellence and good design * Simplicity * Self-organising teams   Regular adaptation to changing circumstances |

A number of project management tools are specifically aimed at agile development. Theyse are designed to help plan, track, analyzse, and integrate work. These tools play an important role in agile development, as a means of Knowledge Management.   
Common features include:

* version control integration
* progress tracking
* easy work allocation
* integrated release and iteration planning
* discussion forums

reporting and tracking of software defects

|  |
| --- |
|  |
|  |
| **Agile Tools** |
| * Rally Software * Versionone * TargetProcess * Assembla * Acunote * PPTS * Mingle * Gatherspace * Visionproject |

The following are a few popular agile methods:

* **Scrum** is an iterative incremental process of software development commonly used with agile software development.
* **Agile Modelling** is a practice-based methodology for modelling and documentation of software-based systems; it is a collection of values, principles, and practices for modelling software, which can be applied on a software development project in a more flexible manner than traditional modelling methods.
* **Agile Unified Process (AUP)** is a simplified version of the IBM Rational Unified Process (RUP). It describes a simple, easy-to-understand approach of developing business application software using agile techniques and concepts, yet remaining true to the RUP at the same time. The AUP applies agile techniques, including test-driven development (TDD), agile modelling, agile change management and database refactoring, to improve productivity.
* **Agile Data Method** defines strategies that IT professionals can apply in a wide variety of situations and work together effectively on data aspects of software systems. However, this does not mean that agile data (AD) is a “one size fits all” methodology. AD is a collection of philosophies that enable IT professionals within an organisation to work together effectively with respect to data aspects of its software-based systems.
* **Test-Driven Development (TDD)** is a software development technique consisting of short iterations, where new test cases that cover the desired improvement or new functionality are written first, followed by the implementation of the production code necessary to pass these tests. Finally, the software is refactored to accommodate changes. The availability of tests before the actual development ensures rapid feedback after any change. Practitioners emphasise that test-driven development is a method for designing software and not merely a method for testing.
* **Feature-Driven Development (FDD)** is an iterative and incremental software development process and is one of the various agile methods for developing software; it forms part of the Agile Alliance. FDD blends a number of industry-recognised best practices into a cohesive whole. All these practices are driven from a client-valued functionality (feature) perspective with the aim to deliver tangible, working software repeatedly in fixed intervals of time.
* **Behavior-Driven Development (BDD)** is an agile software development technique that encourages collaboration between developers, QA and non-technical or business participants in a software project. It was originally developed in 2003 by Dan North as a response to TDD, and has evolved over the last few years.
* **Extreme Programming (XP)** is a software engineering methodology (and a form of agile software development) that prescribes a set of daily stakeholder practices that embody and encourage particular XP values. Proponents believe that exercising these practices—traditional software engineering practices taken to so-called "extreme" levels—leads to a development process that is more responsive to customer needs ("agile") than traditional methods, creating software of better quality.

**DSDM** is a software development approach originally based on the Rapid Application Development (RAD) methodology. DSDM is an iterative and incremental approach that emphasises continuous user involvement. Its goal is to deliver software systems on time and within the budget, while adjusting for changing requirements along the development process. DSDM is an agile method for developing software and forms a part of the Agile Alliance.