**Agile -** combination of iterative and incremental method.Agile Methods break the product into small incremental builds. These builds are provided in iterations. Agile includes continual planning, learning, improvement, team collaboration, evolutionary development, and early delivery.

**Agile Manifesto :** The Agile Manifesto was written in 2001, describing **four key values and 12 core principles**

**Agile Values :**

-Individuals and interactions over processes and tools

-Working software over comprehensive documentation

-Customer collaboration over contract negotiation

-Responding to change over following a plan

**Agile principles :**

1. **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 method 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 maximizing the amount of work not done - is essential
11. The best architectures, requirements, and designs emerge from **self-organizing teams**
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly

**advantages of agile software development**

-Reduced risk of failure

-Reduced time-to-market with predictable delivery

-Improved satisfaction of development teams

-High quality

-Collaborative environment for faster delivery

**Disadvantages of agile software development**

-Less predictability

-Lack of necessary documentation

-No finite end

-Fragmented output

**Agile Methodology :**

**Kanban:** Kanban is a simple, visual means of managing projects that enables teams to see the progress so far and what’s coming up next. Kanban projects are primarily managed through a Kanban board, which segments tasks into three columns: “To Do,” “Doing,” and “Done.” Is used to deliver in a continuous flow and are not tied to a definite schedule.

**Extreme Programming (XP) :** prioritises customer satisfaction over everything else. This methodology offers trust to the developers by motivating them to accept changes in the customer’s requirements, even if they arrive in a later stage of the development cycle.

**Feature Driven Development (FDD)** : This method is focused around "designing & building" features.It has more rigorous documentation requirements than XP, so it’s better for teams with advanced design and planning abilities. FDD breaks projects down into five basic activities:

-Develop an overall model

-Build a feature list

-Plan by feature

-Design by feature

-Build by feature

**Dynamic Systems Development Method (DSDM) :** which focuses on full project delivery and based on the Rapid Application Development (RAD) methodology. What makes DSDM different is, active involvement of the user and the decision making power is with the teams working on it. The teams are empowered to make decisions.

**Lean Software Development :** It aims at increasing speed of software development and decreasing cost. Lean development can be summarized in seven steps.

-Eliminate waste

-Build quality in

-Create knowledge

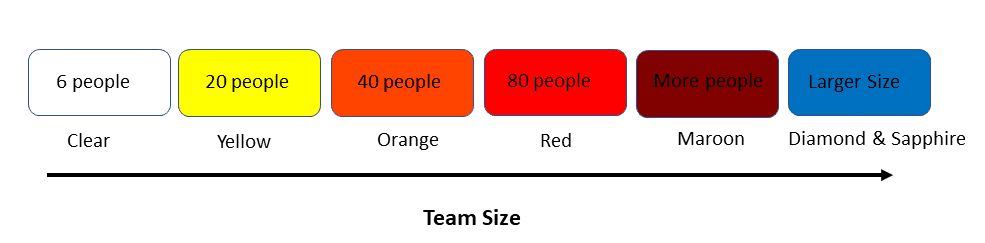
-Defer commitment

-Deliver fast

-Respect people

-Optimize the whole

**Crystal Method:** is a family of methods(Crystal Clear, Crystal Yellow, Crystal Orange, Crystal Red) .Crystal has got different methods for each project type, based on the number of people that are working in the team.



**Scrum :** is a simple light weighted agile project management framework that enables product teams to build products incrementally in an iterative fashion through effective team collaboration.Scrum focuses on breaking a project down into sprints and only planning and managing one sprint at a time.Scrum typically uses a Scrum board, where tasks grouped into columns based on progress.

**Scrum values:**

Courage

Focus

Commitment

Respect

Openness

**Roles in the Scrum**

**Scrum Master**- is responsible for setting up the team, sprint meeting and removes obstacles to progress

**Product owner**- The Product Owner creates product backlog, prioritizes the backlog and is responsible for the delivery of the functionality at each iteration.

**Scrum Team** - Team manages its own work and organizes the work to complete the sprint or cycle

**Sprint** - Agile projects are broken down into iterations and each of these iterations are called sprint.Sprint duration is 1, 2, 3, or 4 weeks, and this varies from organization to organization. Within this timebox, the Scrum team has to finish the agreed set of work.

**Zero Sprint** - is the first step that comes before the first sprint. It includes activities that need to be completed before starting a project, including setting up the development environment, product backlog etc.

**Scrum ceremonies/Activities/Events**

**Sprint Planning** - is the first activity of the sprint and it will be at the beginning of the sprint. In this scrum team take a call on what tasks can be done for the current sprint.

**Daily Stand-up** - scrum team need to give details on what they completed yesterday and what they going to do today and about any blockages.

**Sprint Review/ Demo** - scrum team will show their accomplished work to the product owner.

**Retrospective** - It is the last activity in the sprint and it helps the team understand what worked well–and what didn't.

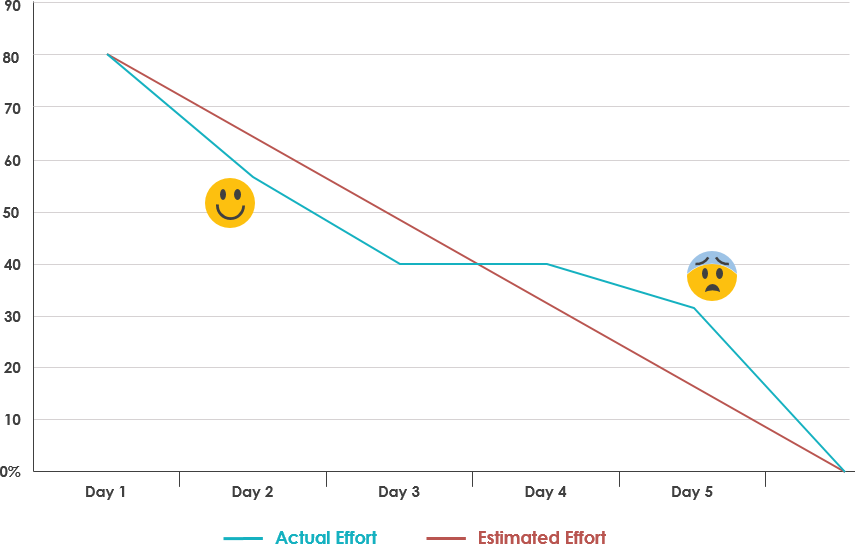
**Scrum Artifacts** - are the information to detail about the product being developed. Main artifacts are Product backlog, Sprint backlog, Product Increment.

**Product Backlog** - is a prioritized list of requirements/user stories that might be needed in the product. The most important items are shown at the top of the product backlog so the team knows what to deliver first.

**Sprint backlog** - is a list of tasks identified by the Scrum team to be completed during the current sprint. The sprint backlog is updated during the sprint planning phase of scrum.

**Product increment** - is the sum of all the Product Backlog items completed during a Sprint combined with the increments of all previous Sprints.

**Burn down chart** is a graphical representation of work left to do versus time.



**Definition of Done** is usually a checklist of all the work that team needs to do before it can call the product increment as “Done”. (Acceptance criteria is specific to story while DOD is at product increment level)

**Velocity** is a measure of the amount of work a Team can tackle during a single Sprint and is the key metric in Scrum

Velocity = (No.of Stories \* Story point ) per sprint

**Agile Estimation Techniques**

**User story** - is an informal, general explanation of a software feature written from the perspective of the end user.

**Epic** - Group of related user stories are called Epic.

**Story point** - is an unit to estimate the difficulty of implementing a given user story. Story point usually based on Fibonacci number (0,1,1, 2, 3, 5, 8, 13, 21…)

Agile estimation techniques are **collaborative**. All appropriate people are included in the process. Its based on **Top-Down approach** where estimation will be on only the features which going to be delivered in next few iterations, not the whole product.

Some of the techniques are,

-Planning Poker

-T-Shirt Sizes

-Dot Voting

-The Bucket System

-Large/ Uncertain/ Small

-Ordering Method

**Planning/scrum Poker** - is a card-based story point estimation technique which is based on a general agreement.