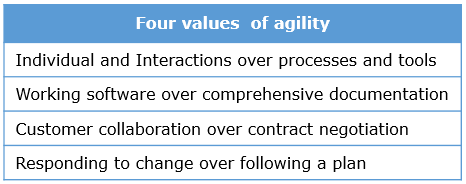
In all the instances of project mentioned earlier, changes are a given and responding to it quickly to ensure quick ROI(return on investment) is the key.

Agile is a time boxed, iterative approach to software delivery that builds software incrementally from the start of the project, instead of trying to deliver it all at once at the end.

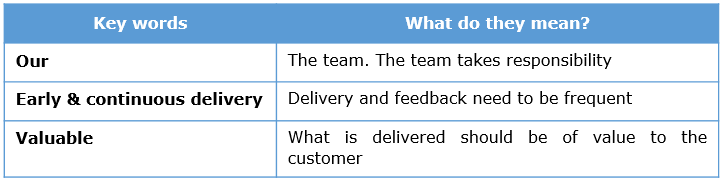
Agile is governed by the four values and twelve principles as provided in the [Manifesto for Agile software development](http://agilemanifesto.org/)

The four values are listed below.

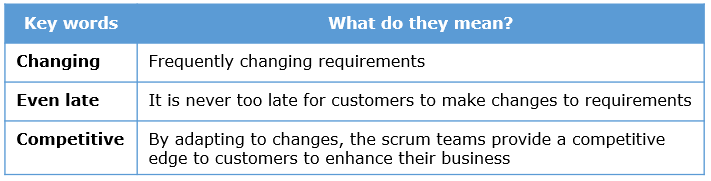


Let us look at the 12 principles in the manifesto for Agile software development

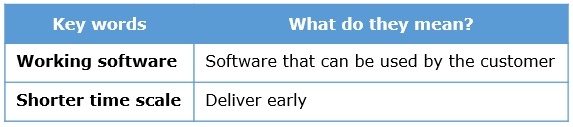
**1) Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.**

****

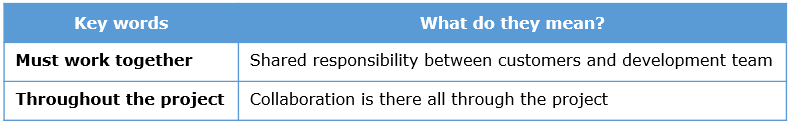
**2) Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.**

****

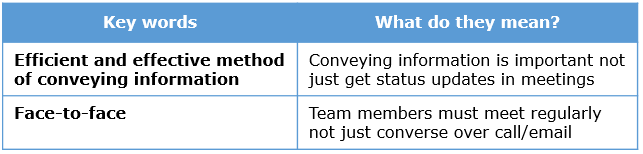
**3) Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.**

****

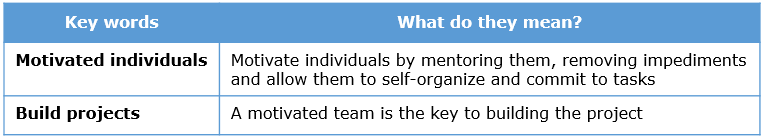
**4) Business people and developers must work together daily throughout the project.**

****

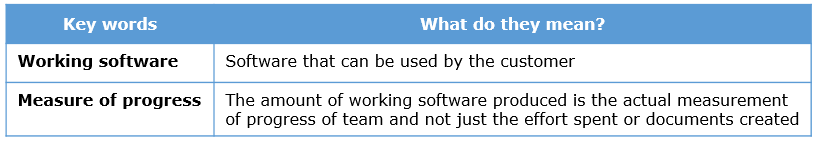
**5) The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.**

****

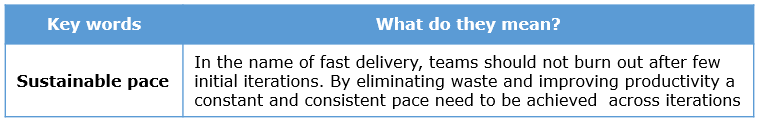
**6) Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.**

****

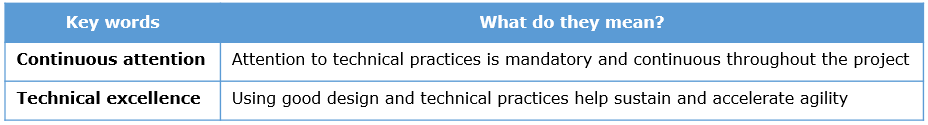
**7) Working software is the primary measure of progress.**

****

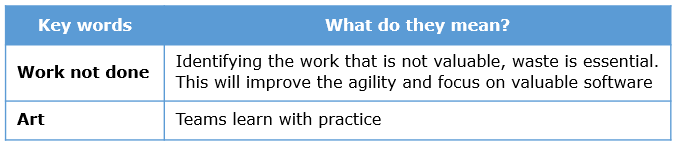
**8) Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.**

****

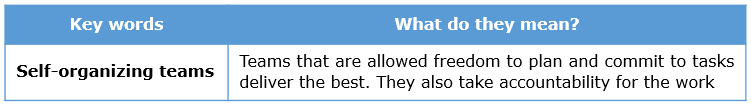
**9) Continuous attention to technical excellence and good design enhances agility.**

****

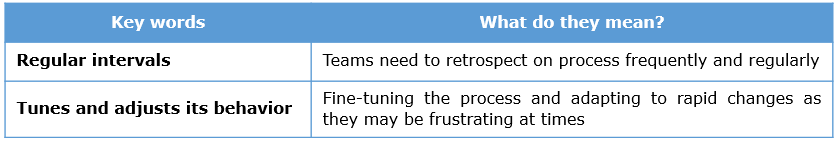
**10) Simplicity--the art of maximizing the amount of work not done--is essential**

****

**11) The best architectures, requirements, and designs emerge from self-organizing teams.**

****

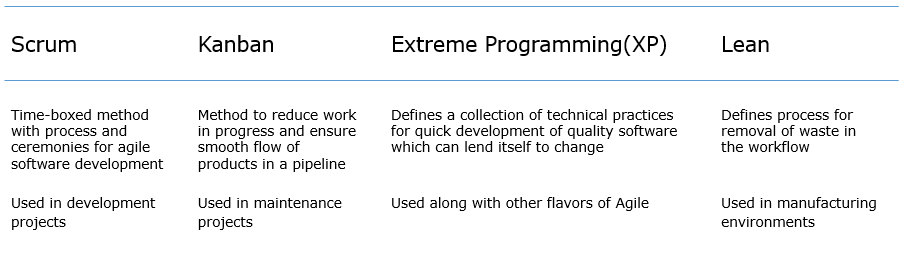
**12) At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.**

****

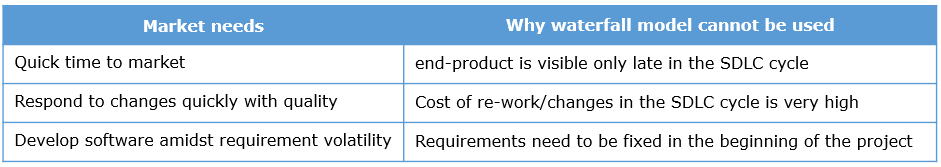
**Additional reading :**

[**Evolution of Agile practices**](https://www.agilealliance.org/agile101/practices-timeline/)

For all the project instances discussed, agility can be achieved by choosing one the given methodologies. The applicability of different agile flavors and its salient features are listed below.



Waterfall model cannot be used to address the market needs for the reasons given below.

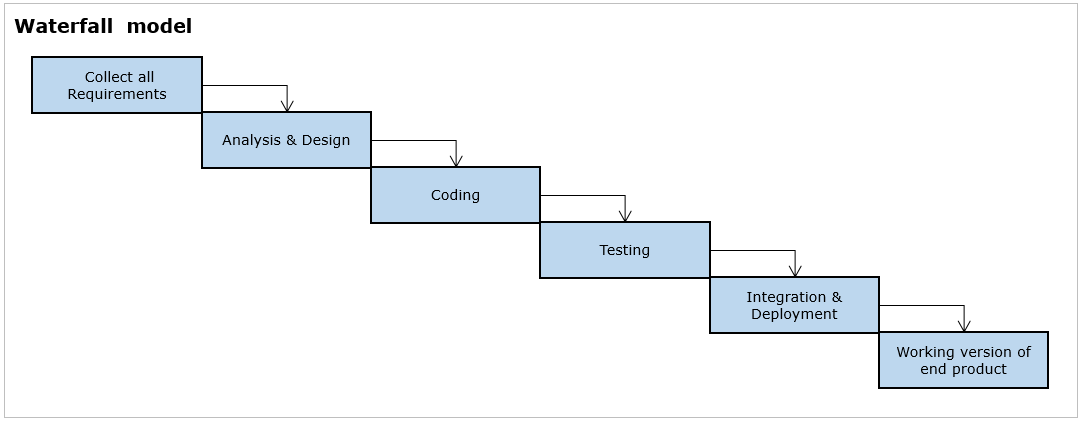


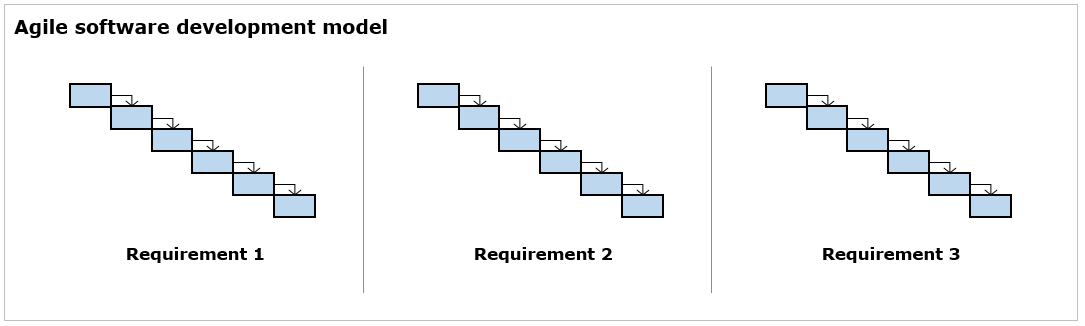
We can conclude that waterfall model cannot address the market needs and hence we need the Agile approach to software development.

In the waterfall model, each software development life cycle(SDLC) phase has a specific deadline and deliverable that flows into the next phase. Hence all requirements are collected, then designed, code written, tested ,integrated and deployed at the end.

In Agile, the requirements are broken down into small parts and all the SDLC phases are completed for it. Hence small pieces of working software get released frequently.

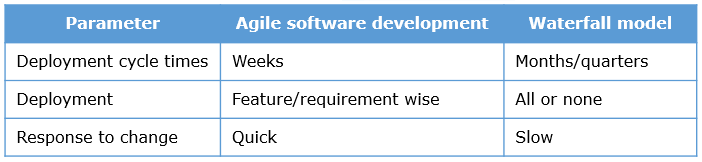
This difference is illustrated in the figure below.



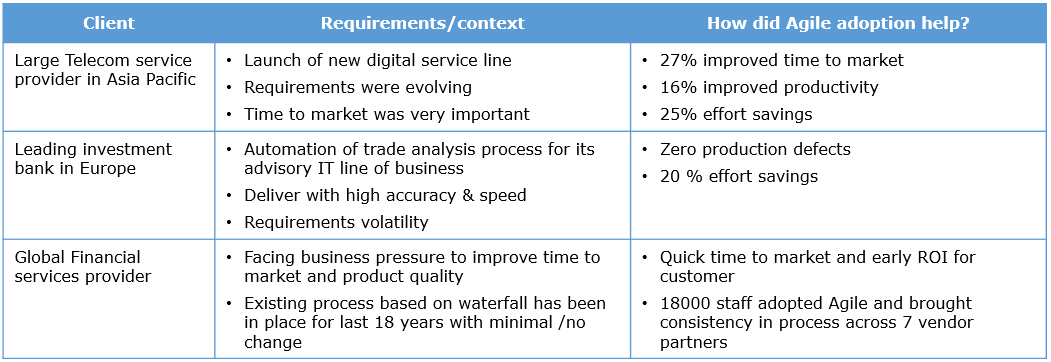
****

**Summary**

The differences between the two software development approaches with respect to key parameters is given below.



**Projects at Infosys**



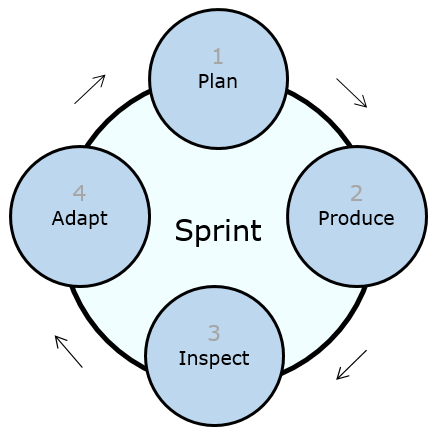
EasyGarb is a prospective internet based retail marketing establishment offering a range of textile products from daily-wear, party-wear, ethnic to top notch fashion wear based in 3 locations in Asia.

To remain competitive in the market the management team decides to develop an online solution. High requirement volatility is expected and features are to be developed and deployed at high speed with good quality based on inputs from customers and market conditions. The teams identified for software development have worked in waterfall mode before.

This is a scenario of a development project, the deliverables are expected to be deployed in time-boxed periods. Hence Agile approach has to be adopted and **Scrum**would be the most suitable one. Kanban does not apply here because the requirement is not to reduce work in progress. XP is a set of practices which can be followed along with Scrum to bring good quality.

Let us now understand Scrum and its jargon.

Let us understand the scrum concepts in detail. Scrum Cycle is the iteration in Scrum popularly referred to as ‘**Sprint**’. A sprint consists of Plan-Produce-Inspect-Adapt. Application development, is achieved by sprints in a sequence with no gaps in between them.



1. **Plan**- The work to be done(how much and what) during the sprint and the deliverable to be given to the customer at the end of the sprint are decided by the team. Planning is done for the sprint at hand
2. **Produce**– Whatever has been planned is implemented by the development team
3. **Inspect**– At the end of the sprint, the customer along with the stakeholders verify the product to see if it meets the requirements as per agreed conditions called definition of done (more on this later)
4. **Adapt**– The team retrospects on what went well in the current sprint, what can continue and what can be done better in the next sprint. Improvements in the process are discussed here. Doing this would help them to continuously improve the way of working across sprints

The commonly used terms in Scrum are listed below.

**Sprint**– The timeboxed iteration in which deliverables are produced by the development team. Sprints happen in a sequence.

**Cross functional** – Refers to the diversity of skill sets present in a development team, ranging from developers,architects,testers,analysts etc.

**Self Organized** – Refers to the capability of development teams

1. To take decisions on how much and what to commit in each sprint
2. Resolve conflicts on their own
3. Take collective ownership and commit to tasks rather than being assigned work

**Definition of Done** – DoD is a generic checklist of conditions agreed upon between the customer and teams on what can be considered done.

Example

* Code complete
* Code reviewed
* Unit Tested
* Integration Tested
* Acceptance Tested
* System Docs and User Docs updated
* No Priority 1 or Priority 2 defects remaining

The commonly used terms in a Sprint are listed below. The details will be discussed subsequently.

The roles, ceremonies and artifacts follow the 3-4-3 format.

# 3 - Roles (People who are part of a sprint)

1. **Product Owner** - Responsible for vision and goals of release, Return On Investment, date and scope of release
2. **Scrum Master** - Responsible for scrum process, problem solving and protection of development team
3. **Development Team** - Responsible for the work done / craft the per sprint goal

# 4 - Ceremonies (Meetings conducted in a sprint)

1. **Sprint Planning Meeting** - Meeting conducted in the beginning of the sprint to arrive at the sprint backlog
2. **Daily Scrum Meeting**- Everyday meeting in a sprint where team members update on the status of their work
3. **Sprint Review**  - Meeting at end of sprint where the product owner checks the doneness of the sprint
4. **Sprint Retrospection**- Meeting where the development team plans for improvement in process respectively
5. **Product Backlog Refinement**- Sprinting ahead session to refine requirements for the upcoming sprints

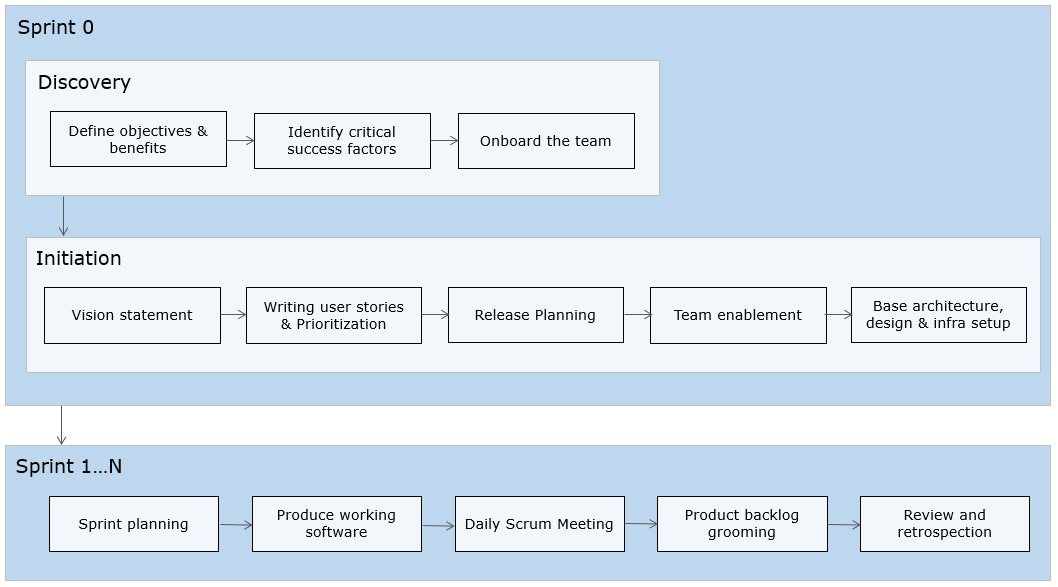
Note : November 2017 Scrum Guide mentions the first four events . Product backlog refinement is part of the sprint progress but we have explicitly mentioned as it is important to be conducted . The words grooming and refinement have been used interchangeably in this course

# 3 - Artifacts (Deliverables produced)

1. **Product Backlog** - Pending list of items to be developed
2. **Sprint Backlog** - List of tasks planned for the current sprint with their effort estimate
3. **Increment** - Working version of the software produced in the sprint

The technical officer at EasyGarb has proposed using Scrum for execution of the project involving the development of the online solution. The management team at EasyGarb wants to understand the process of execution of the project using Scrum. The technical officer provides the big picture as shown in figure below and explains the phases.

**Big picture of a scrum project featuring sprints**



**Note :** Sprint 0 is not mentioned in the official scrum guide. In early stages of transitioning to Agile and in large transformations it is good to have a short sprint 0 to do a high level planning.

Mike Cohn calls it a light weight "project before the project".