

① Difference between XP & Scrum

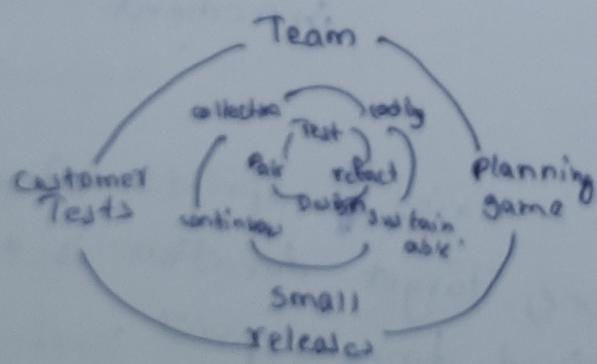
Extreme programming:

Extreme programming (XP) was proposed by Kent. The fact that best practices to extreme levels.

core values of XP

- Communication
- Simplicity
- Feedback
- Courage
- Respect

XP Practices



XP Practices: Planning game

Planning for upcoming iteration. User stories provided by the customer. Technical persons determine schedules, estimates, costs etc.

XP Practices: Small releases

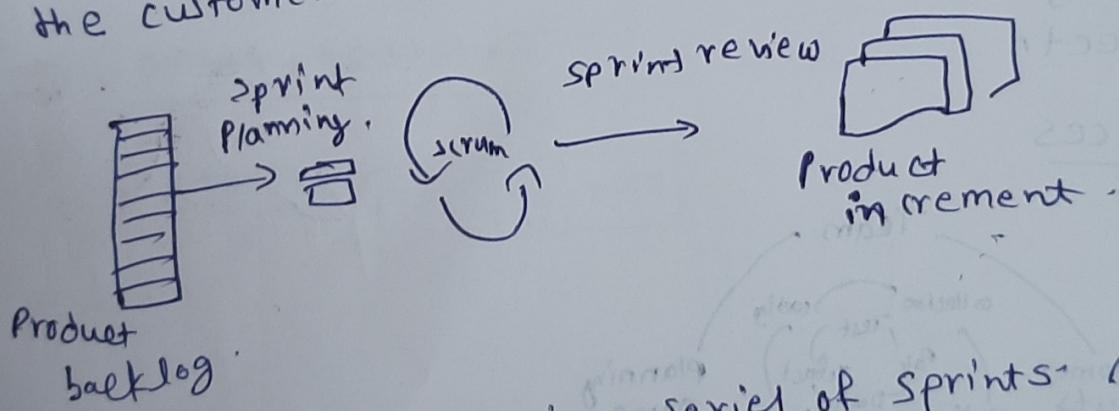
Small in terms of functionality. Team releases running, tested software every iteration.

XP practices: Metaphor

The oral architecture of the system. XP teams develop a common vision of the system. Define common systems of names. Ensure everyone understands how the system works, where to look for functionality, or where to add functionality.

Scrum

Scrum is a distinguishing features. Development work is partitioned into packets. Testing & documentation are on going as the product is constructed. Demos are delivered to the customer with the time-box allocated.



Scrum projects progress in a series of sprints. Analogy to XP iterations or time boxes. Target duration is one month. No changes entertained during a sprint.

Scrum Master:

Ensure that the team is fully functional & productive. Enable close cooperation across all roles & functions. remove barriers.

Sprint planning

Goal is to produce sprint Backlog. product owner works with the team to negotiate what backlog items the team will work on in order to meet. scrum master ensures team agrees to realistic goals.

Sprint review meeting

Informal

- 2-hour prep time rule

Participants:

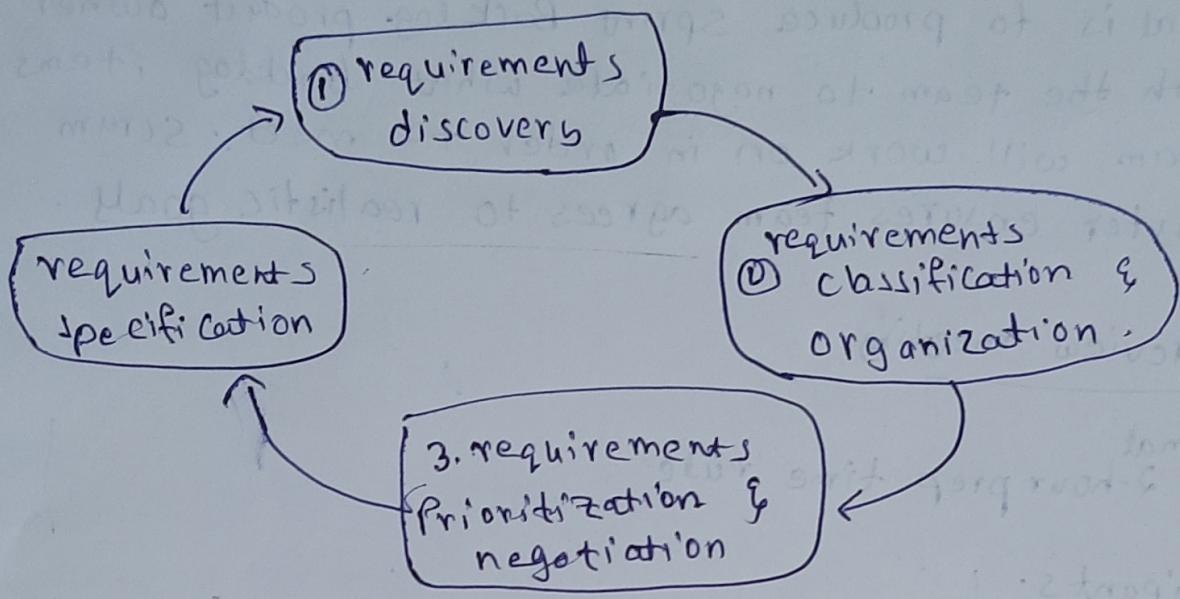
- customers
- management
- product owner
- other team mates

② Techniques used for requirement elicitation

Requirements elicitation is the practice of collecting the requirements of a system from users, customers and other stakeholders. The practice is also sometimes referred to as "requirement gathering".

Stages include

- requirements discovery,
- requirements classification & organization,
- requirements prioritization & negotiation,
- requirements specification.



- ① interacting with stakeholders to discover their requirements. Domain requirements are also discovered at this stage.
- ② groups related requirements & organises them into coherent clusters.
- ③ prioritising requirements & resolving requirement conflicts through negotiation
- ④ requirements are documented & input into the next round of the spiral

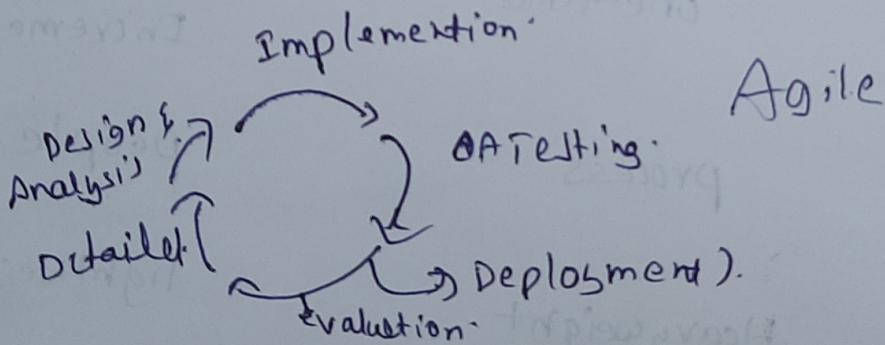
③ Agility

effective response to change, effective communication among all stakeholders + Drawing the customer onto the team.

Agile is a time boxed iterative approach to software delivery that builds software incrementally from the start of the project, instead of trying to delivered it all at once near the end. It works by breaking projects down into little bits of user functionality called user stories, prioritizing them.

why use Agile methods

- Improve customer involvement
- Increase Quality.
- Simplify releases
- Drive Down Risk.



Requirements →

Design

Code

Integrat

waterfall,

Test

Deploy

Agile vs traditional methods

changes.

Traditional method

resisted & controlled

Agile methods

welcomed & Adapted

Binding

Flexible

one-time

Iterative &
Incremental.

process

people.

Heavyweight

lightweight

orientation

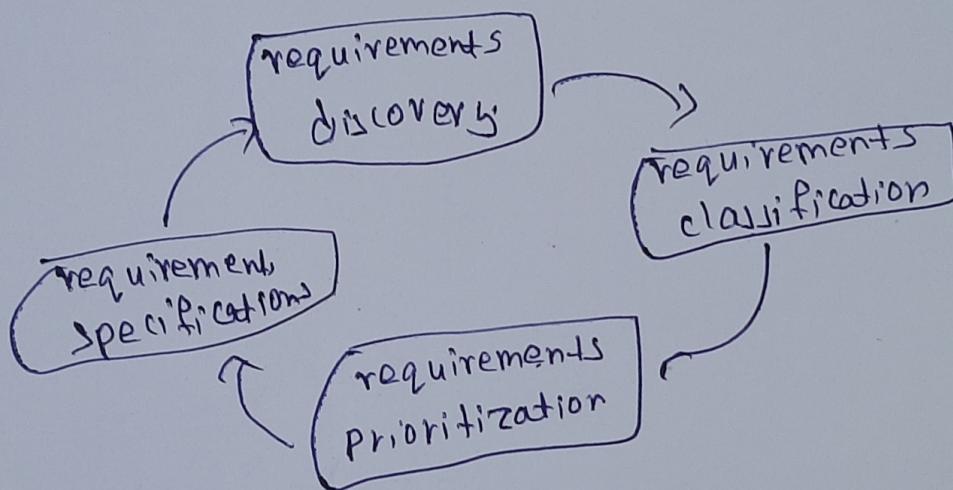
process

④ Requirement Specification

Software engineers work with customers & system end-users to find out about the application domain.

Stages include

requirements specification



requirements are documented & input into the next round of the spiral.

there are generic activities

- requirements elicitation
- requirements analysis
- requirements validation
- requirements management

Requirements elicitation is the practice of collecting the requirements of a system from users, customers & other stakeholders.