

Agile Methodology

What is Agile Methodology?

AGILE methodology is a practice that promotes continuous iteration of development and testing throughout the software development lifecycle of the project. Both development and testing activities are concurrent unlike the Waterfall mode

The agile software development emphasizes on four core values.

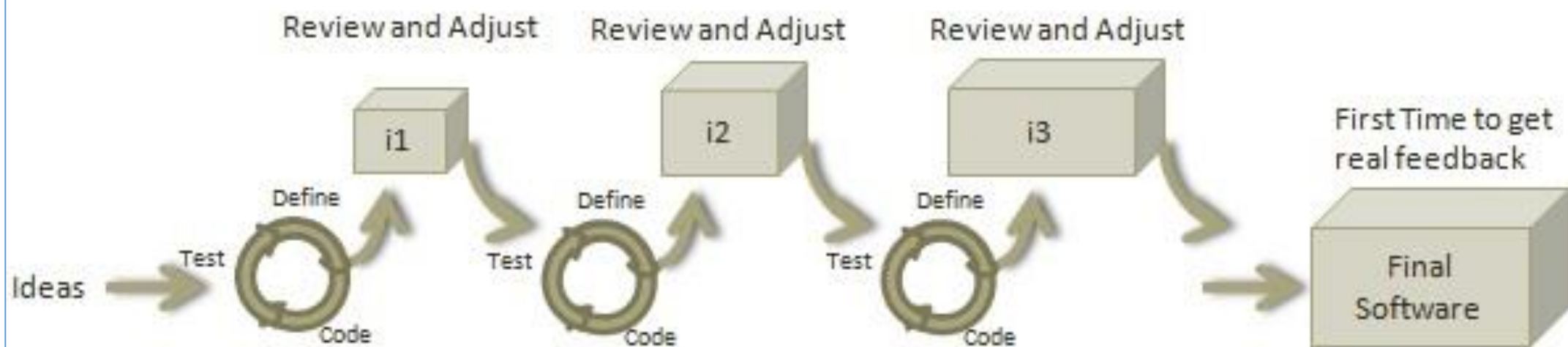
- Individual and team interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

Five reasons

- **Early Time to Market need**
- **Ever changing requirement - difficult to adopt.**
- Basic Systems in place (thru waterfall). Most of the projects were on incremental modifications. Need of quick implementation.
- Need of continuous interaction with Customer.
- Waterfall was heavy on documentation.



Traditional Method



Agile Method

- **Customer Satisfaction** – Highest priority is given to satisfy the requirements of customers through early and continuous delivery of valuable software.
- **Welcome Change** – Changes are inevitable during software development. Ever-changing requirements should be welcome, even late in the development phase. Agile processes should work to increase customers' competitive advantage.
- **Deliver a Working Software** – Deliver a working software frequently, ranging from a few weeks to a few months, considering shorter time-scale.
- **Collaboration** – Business people and developers must work together during the entire life of a project.
- **Motivation** – Projects should be built around motivated individuals. Provide an environment to support individual team members and trust them so as to make them feel responsible to get the job done.
- **Face-to-face Conversation** – Face-to-face conversation is the most efficient and effective method of conveying information to and within a development team.

Agile Testing Methodology

Scrum

Crystal Methodologies

DSDM (Dynamic Software Development Method)

Feature driven development (FDD)

Lean software development

Extreme Programming (XP)

Agile vs Waterfall

Agile Model	Waterfall Model
Agile method proposes incremental and iterative approach to software design	Development of the software flows sequentially from start point to end point.
The customer has early and frequent opportunities to look at the product and make decision and changes to the project	The customer can only see the product at the end of the project
Agile model is considered unstructured compared to the waterfall model	Waterfall model are more secure because they are so plan oriented
Small projects can be implemented very quickly. For large projects, it is difficult to estimate the development time.	All sorts of project can be estimated and completed
Development process is iterative, and the project is executed in short (2-4) weeks iterations.	The development process is phased, and the phase is much bigger than iteration.

Agile vs Waterfall

Agile Model	Waterfall Model
Documentation attends less priority than software development	Documentation is a top priority and can even use for training staff and upgrade the software with another team
In agile testing when an iteration end, shippable features of the product is delivered to the customer. New features are usable right after shipment.	All features developed are delivered at once after the long implementation phase.
At the end of every sprint, user acceptance is performed	User acceptance is performed at the end of the project.
Testers and developers work together	Testers work separately from developers
It requires close communication with developers and together analyze requirements and planning	Developer does not involve in requirement and planning process. Usually, time delays between tests and coding

SCRUM

Scrum Methodology

- Scrum is a framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value.
- The Scrum framework consists of Scrum Teams and their associated roles, events, artifacts, and rules. All events are time-boxed events, such that every event has a maximum duration.

Sprint

- The heart of Scrum is a Sprint, a time-box of two weeks or one month during which a potentially releasable product increment is created. A new Sprint starts immediately after the conclusion of the previous Sprint. Sprints consist of the
 - Sprint planning - the work to be performed in the Sprint
 - daily scrums meeting - 15-minute time-boxed meet daily
 - the development work - develop and test the change
 - the Sprint review - Review the increment
 - the Sprint retrospective - inspect itself and create a plan for improvements

Project Inception

TO Planning

Requirements
Refinement

Product
Backlog

Sprint 1- n

Continuous
Integration

Integration
Regression
Testing

Development

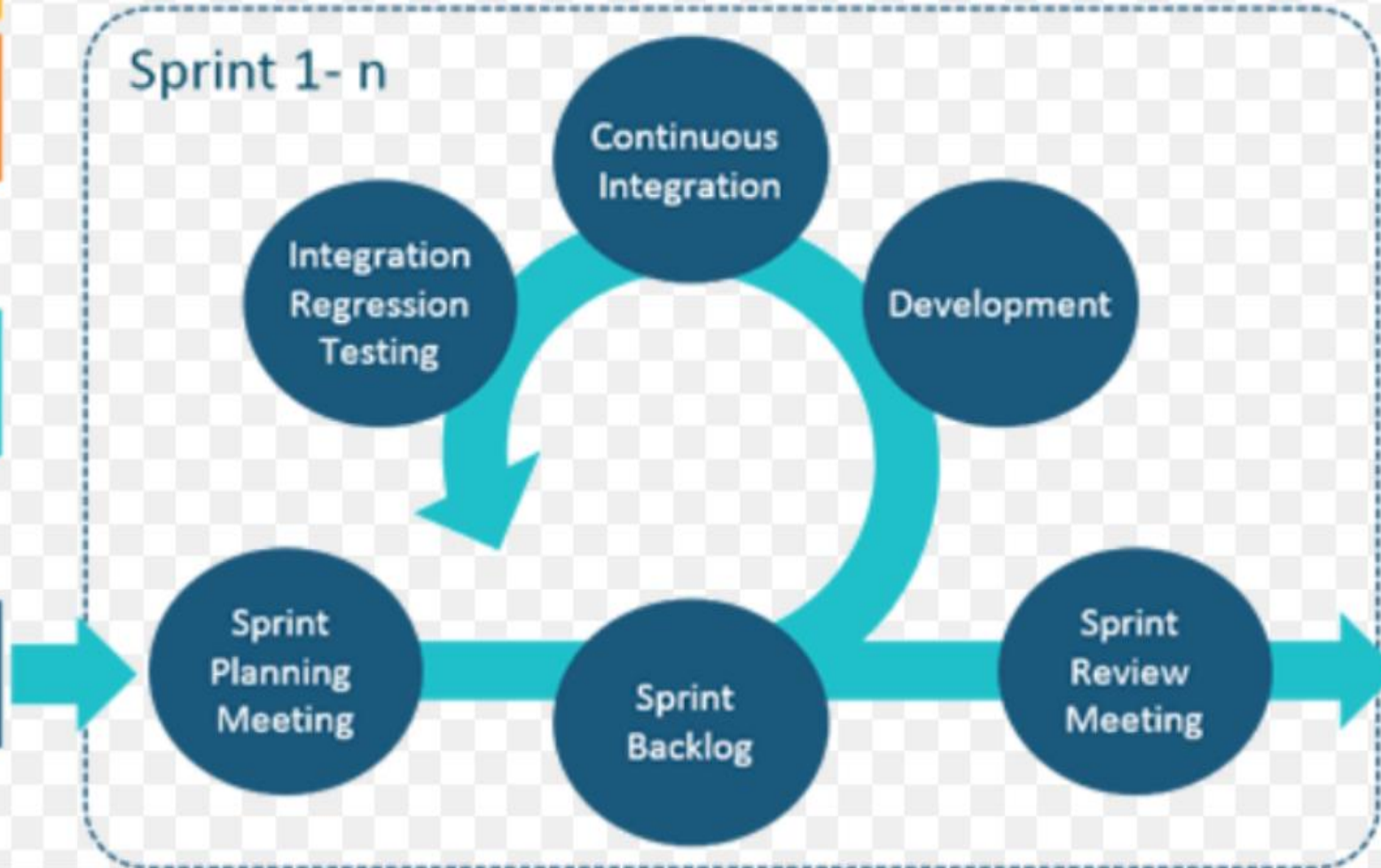
Sprint
Planning
Meeting

Sprint
Backlog

Sprint
Review
Meeting

Transition

Functionally Tested
Software



Roles in Scrum

Scrum Master

- 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



Artifacts in Scrum

1. Product Backlog

- This is a repository where requirements are tracked with details on the no of requirements to be completed for each release.
- A product backlog is a list of items to be done. Items are ranked with feature descriptions. In an ideal scenario, items should be broken down into user stories.
- It helps to prepare estimates, roadmap and prioritization of features.

2. Sprint Backlog

- The Sprint Backlog is the set of Product Backlog items selected for the Sprint

3. Sprint Burn-Down Chart

- At any point in time in a Sprint, the total work remaining in the Sprint Backlog can be summed. The Team tracks this total work remaining for every Daily Scrum to project the likelihood of achieving the Sprint Goal. useful technique in monitoring the Sprint progress towards the Sprint Goal.

4. Increment

The Increment is the sum of all the Product Backlog items completed during a Sprint combined with the increments of all previous Sprints. At the end of a Sprint, the new Increment must be a working product, which means it must be in a useable condition.

The Scrum framework consists of Scrum Teams and their associated roles, events (time boxed), artifacts, and rules.

Sprint

- The heart of Scrum is a Sprint, a time-box of two weeks or one month during
- A potentially releasable product increment is created.

Roles in Scrum

Scrum Master

Responsible for setting up the team, sprint meeting.

Product owner

Creates product backlog, prioritizes the backlog and is responsible for the delivery of the functionality at each iteration.

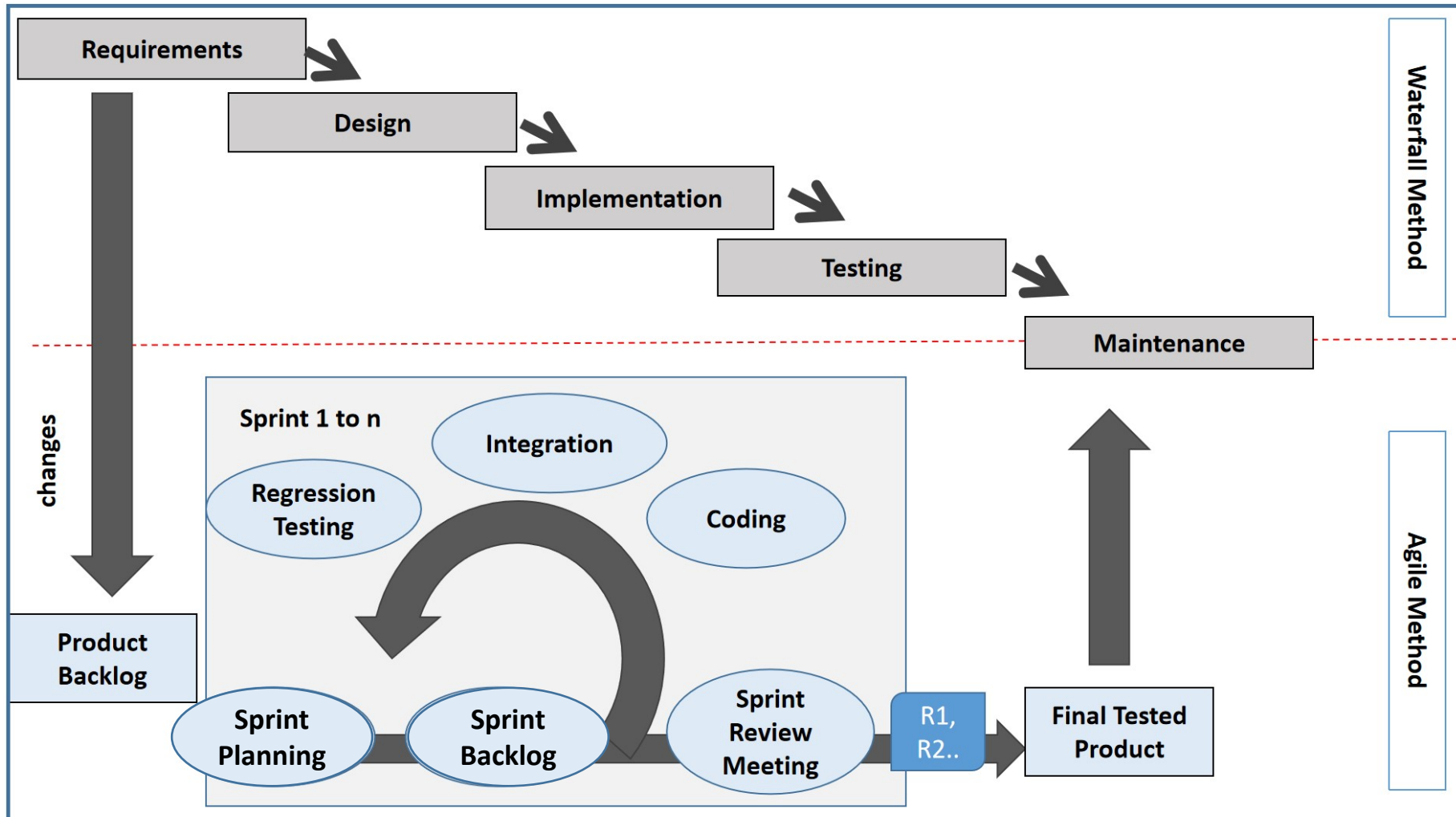
Scrum Team

Team organizes and manages its own work to complete the sprint.

Activities in Sprint

- Sprint planning
- daily scrums meeting
- the development work
- the Sprint review
- the Sprint retrospective

The shift : TSD to ASD



Constructivism

- A cognitive theory in education ,where the learners construct new understandings and knowledge, integrating with what they already know.
- In Agile method, software is developed in short iterations and the team incorporated all new knowledge gained through feedback from preceding iterations.

Fig 3: Comparison of Traditional Software Development (TSD) and Agile Software Development (ASD) process

Artifacts in Scrum

Basic Concepts

- **Product Backlog** : A product backlog is a list of items(features/user stories) to be done.
- **Sprint Backlog**: Subset of Product Backlog items selected for the Sprint.
- **Sprint/Release Burn-Down Chart** : shows the rate at which work is completed and how much work remains to be done.
- **Increment** :The Increment is the sum of all the Product Backlog items completed during a Sprint combined with the increments of all previous Sprints.

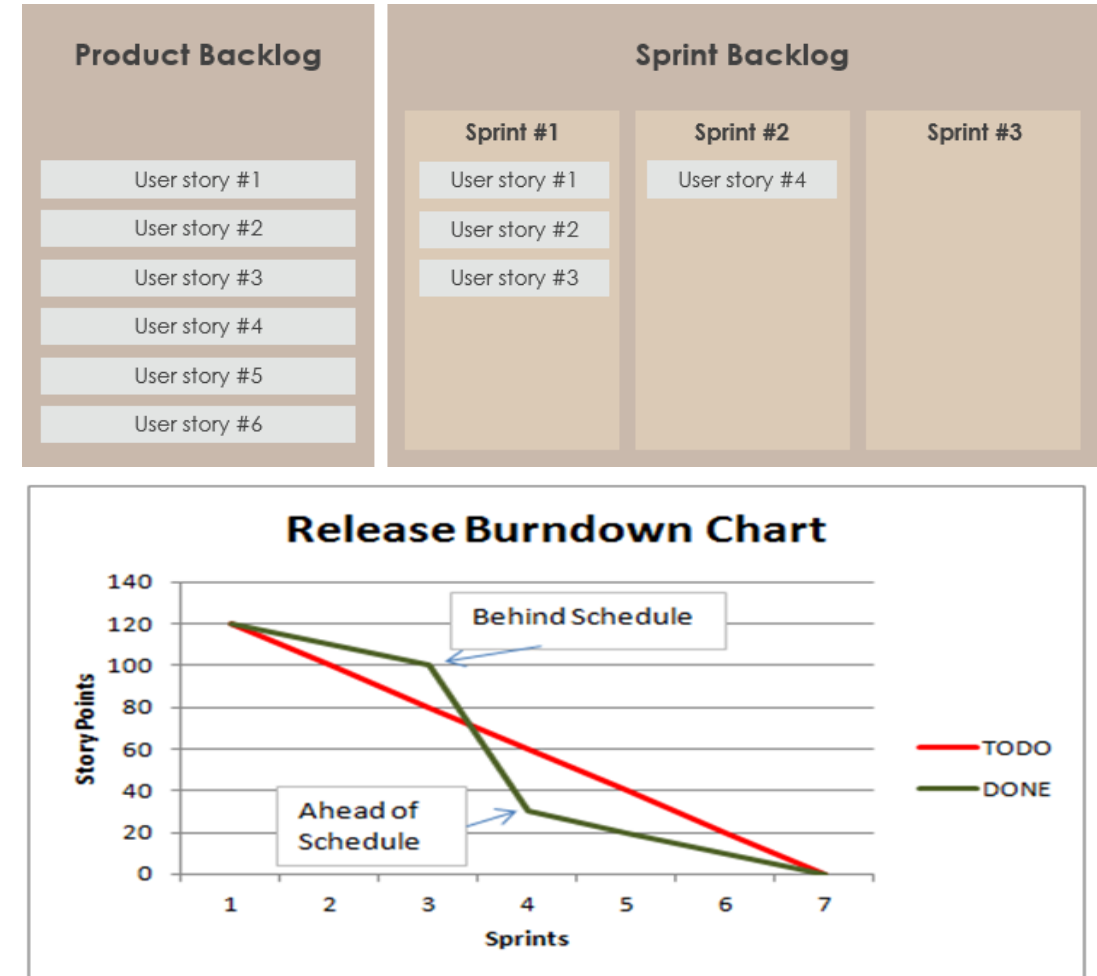


Fig 4 : Important artifacts of Scrum

Process flow of Scrum Methodologies

Process flow of scrum testing is as follows:

- Each iteration of a scrum is known as Sprint
- Product backlog is a list where all details are entered to get end product
- During each Sprint, top items of Product backlog are selected and turned into Sprint backlog
- Team works on the defined sprint backlog
- Team checks for the daily work
- At the end of the sprint, team delivers product functionality

eXtreme Programming (XP)

Extreme Programming (XP) is based on the five values –

Communication

Simplicity

Feedback

Courage

Respect

Extreme Programming is a systematic approach with a set of values, rules and practices for rapidly developing high quality software that provides the highest value for customers.

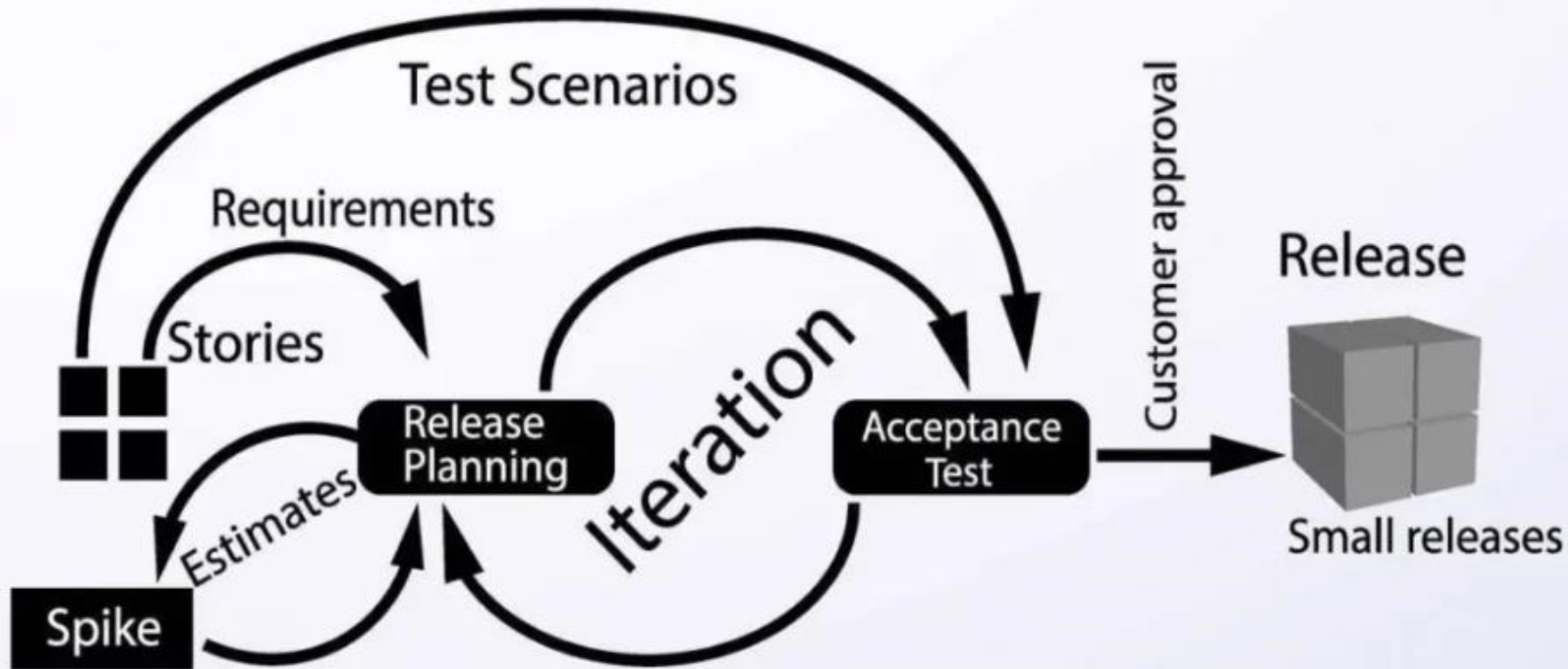
eXtreme Programming (XP)

- Extreme Programming technique is very helpful when there is constantly changing demands or requirements from the customers or when they are not sure about the functionality of the system.
- It advocates frequent "releases" of the product in short development cycles, which inherently improves the productivity of the system and also introduces a checkpoint where any customer requirements can be easily implemented.
- The XP develops software keeping customer in the target.

Basic principles of Extreme programming:

- **Business requirements are gathered in terms of stories.** User stories are simple and informal statements of the customer about the functionalities needed. All those stories are stored in a place called the **parking lot**.
- Based on User stories, the project team proposes **Metaphors**. Metaphors are a common vision of how the system would work (design).
- The development team may decide to build a **Spike(like prototype)** for some features.
- In this type of methodology, releases are based on the shorter cycles called Iterations with span of **14 days time period**. Each iteration includes phases like coding, unit testing and system testing where at each phase some minor or major functionality will be built in the application.

Extreme Programming



Activities

- Design
- Coding
- Testing
- Feedback
- Pair Programming
- Continuous Integration

Process of eXtreme programming

- **User stories** are the heart of planning in Extreme Programming (XP).
- High level designs or **metaphors** are created from stories.
- **Architectural spikes** or prototypes are used to create a simple overall design.
- High code quality is essential on an XP project.
- Developers receive feedback constantly by working in pairs and testing code as it is written. Managers get feedback on progress and obstacles at the daily stand up meeting

Roles

- Developer (also called Programmer by some teams)
- Customer
- Manager (also called tracker)
- Coach

Scrum vs Extreme Programming

	SCRUM	XP
1	In the Scrum framework, teamwork in iterations is called Sprint which is 2 weeks to 1 month long.	In Extreme Programming(XP), teamwork for 1-2 weeks only.
2	Scrum models do not allow changes in their timeline or their guidelines.	Extreme Programming allows changes in their set timelines
3	In scrum, the scrum master asks the owner of the product to prioritize the tasks according to their requirements.	In XP, customer decides the job priorities being the owner of the product and then analyses the releases
4	The tasks are prioritized by the owner of the product but with the flexibility that the priorities can be changed later on by the development team if required.	The tasks are prioritized by the customer and the task priorities cannot be changed by the development team.
5	Values- Openness,Focus,Commitment	Values- Communication,Simplicity,Feedback