Mphasis Tutorials point reading materials Agile

Contents

[AGILE – PRIMER 4](#_Toc47248739)

[What is Agile? 4](#_Toc47248740)

[Roles in Agile 4](#_Toc47248741)

[Scrum Master 4](#_Toc47248742)

[Product Owner 4](#_Toc47248743)

[Cross-functional Team 4](#_Toc47248744)

[How an Agile Team Plans its Work? 5](#_Toc47248745)

[Iteration planning 5](#_Toc47248746)

[User Stories 5](#_Toc47248747)

[Review 5](#_Toc47248748)

[User Story 5](#_Toc47248749)

[Relationship of User Stories and Tasks 5](#_Toc47248750)

[When a Story is Done 5](#_Toc47248751)

[What are Acceptance Criteria? 5](#_Toc47248752)

[How the Requirements are Defined? 6](#_Toc47248753)

[References 8](#_Toc47248754)

# AGILE – PRIMER

## What is Agile?

It is a SDLC where the software is built incrementally, with each incrementations happening in short iterations of 1-4 weeks. This means that as the needs of the client changes or becomes clearer or more specific, the development process can keep up with the changes, and the requirement plan can be adjusted accordingly.

Instead of a single-pass development of 6 to 18 months where all the requirements and risks are predicted upfront, Agile adopts a process of frequent feedback where a workable product is delivered after 1 to 4-week iteration.

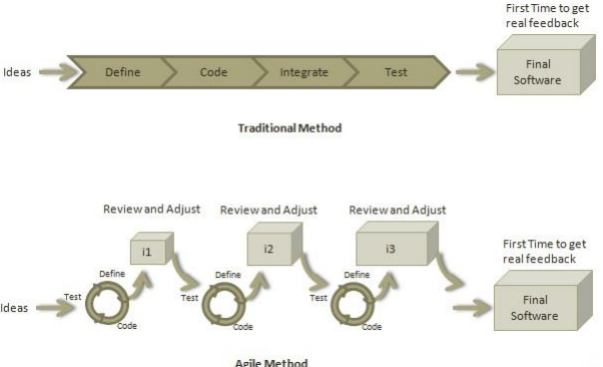


Figure 1 Agile Method

## Roles in Agile

There are 2 main leadership roles in an Agile team

### Scrum Master

Team leader, helps the team to follow scrum practices. Works as an interface between the technical and business aspects of the team. Ensures close cooperation between members, and ensures that agile ‘Inspect and Adapt’ practices are leveraged properly:

1. Daily stand-ups
2. Planned meetings
3. Demo
4. Review
5. Retrospective Meetings

### Product Owner

They drive the product from the business perspective. Their job to ensure the product satisfies all the business requirements. Their responsibilities are:

1. To define the requirements and prioritize their values.
2. To determine the release date and contents.
3. To take an active role in iteration planning and release planning meetings.
4. To ensure that team is working on the most valued requirement.
5. To represent the voice of the customer.
6. To accept the user stories that meet the definition of done and defined acceptance
7. criteria.

## Cross-functional Team

An agile team generally consists of developers, testers, one technical lead, one product owner, and one scrum master. Product Owner and Scrum master are considered to be a part of Team Interface, whereas other members are part of Technical Interface.

## How an Agile Team Plans its Work?

An Agile team works in iterations to deliver user stories where each iteration is of 10 to15 days. Each user story is planned based on its backlog prioritization and size. The team uses its capacity − how many hours are available with team to work on tasks − to decide how much scope they have to plan. In each release, there are 3 stages.

### Iteration planning

Planning what is going to be in each user story.

### User Stories

Cycles of Define, Code, Test, Define, Code, Test, etc.

### Review

Review of the work done, usually culminating in an After-Action Review plan document.

In each release, a number of User Stories are selected, and points given to them in a rough estimate. In iteration planning, User Stories are broken down into tasks and subtasks.



A Point is the amount of time a team can commit to the software user story. A point is usually 8 hours. Each story is estimated in points. The total amount of points adds up to the amount of time a team can work in the fixed time, generally 2 weeks.

Capacity is the amount of time an individual can commit. Total capacity of the team is equal to the total amount of points x (hour/point), and is equivalent to the amount of time a team can work in the fixed time, generally 2 weeks.

## User Story

A user story is a requirement which defines what is required by the user as functionality. A user story can be in two forms:

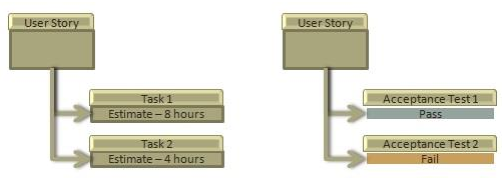
* As a <User Role> I want <Functionality> so that <Business Value>
* In order to <Business value> as a <User Role> I want <Functionality>

User story is a requirement that a particular role wants to realize the need for a particular business value. During release planning, a rough estimate is given to a user story using relative scale as points. During iteration planning, the story is broken down into tasks.

## Relationship of User Stories and Tasks

User story talks about what is to be done. It defines what a user needs. User story is divided into tasks during planning. Task talks about how it is to be done. It defines how a functionality is to be implemented. Stories are implemented by tasks. Each story is a collection of tasks. Tasks are estimated in hours, typically from 2 to 12 hours.

Stories are validated using acceptance tests. Tasks are validated using unit, integration and regression testing.



## When a Story is Done

The team decides what done means. The criteria may be:

* All tasks (development, testing) are completed.
* All acceptance tests are running and are passed.
* No defect is open.
* Product owner has accepted the story.
* Deliverable to the end-user.

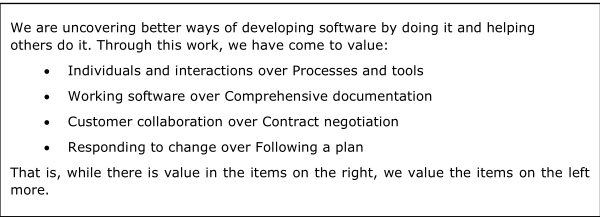
## What are Acceptance Criteria?

Criteria defines the functionality, behavior, and performance required by a feature so that it can be accepted by the product owner. It defines what is to be one so that the developer knows when a user story is complete. It is the minimum number of requirements that must be satisfied by a story for the product owner to be ok with it.

## How the Requirements are Defined?

Requirements are defined as a User Story, with acceptance criteria, and tasks to implement the story.

# AGILE – MANIFESTO



Basically, hippie shit.

## Principles of agile

Customer Satisfaction - Highest priority is given to satisfy the requirements of customers through early and continuous delivery of valuable software. Customer’s business needs are the priority.

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. Change in requirements should be encouraged.

Deliver a Working Software - Deliver a working software frequently, ranging from a few weeks to a few months, considering shorter time-scale. Getting the software working properly should be the priority, with frequent releases to the customer.

Collaboration - Business people and developers must work together during the entire life of a project. Devs, customers, and product owners should work together closesly to ensure the sft is exactly what is needed.

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.

Measure the Progress as per the Working Software - Working software is the key and it should be the primary measure of progress. Nothing else matters more than if the software works or not—whether it satisfies all the customer’s requirements.

Maintain Constant Pace - Agile processes aim towards sustainable development. The business, the developers, and the users should be able to maintain a constant pace with the project. The keyword is constant—proper and practical planning should not assume crunches or miracles. Breakneck speeds are not sustainable.

Monitoring - Pay regular attention to technical excellence and good design to enhance agility. Nothing can compensate for technical faults, and good design adds value everywhere.

Simplicity - Keep things simple and use simple terms to measure the work that is not completed. Complex stuff doesn’t really help.

Self-organized Teams - An agile team should be self-organized and should not depend heavily on other teams because the best architectures, requirements, and designs emerge from self-organized teams. No one is coming to help.

Review the Work Regularly - Review the work done at regular intervals so that the team can reflect on how to become more effective and adjust its behavior accordingly. Regular reviews ensure that the team is on the right track.

# AGILE – CHARACTERISTICS

There are 3 main characteristics

Iterative/incremental and Ready to Evolve

Face-to-face Communication

Feedback Loop

Iterative/incremental and Ready to Evolve

Most of the agile development methods break a problem into smaller tasks. There is no direct long-term planning for any requirement. Normally, iterations are planned which are of vary short period of time, for example, 1 to 4 weeks. At the end of iterations, the sft is handed over, and this is called a release. Planning only occurs until a release, further plans are not concrete. After demo, review comments are taken and are planned to be incorporated in the working software as required.

Face-to-face Communication

“Each agile team should have a customer representative such as a product owner in scrum methodology. This representative is authorized to act on behalf of the stakeholders and he can answer the queries of the developers in between iterations. An information radiator (physical display) is normally located prominently in an office, where passers-by can see the progress of the agile team. This information radiator shows an up-to-date summary of the status of a project.”

Face to face communication is emphasized, but the real idea is effective communication. Hashing out most of the details over email, and talking out the rest in a meeting is also fine, if it is more effective than face to face communication.

And instead of an information radiator, a dashboard or a checklist can also be used if it is more efficient.

Feedback Loop

Daily stand-up is a common culture of any agile development; it is also known as daily scrum. It is a kind of a brief session where each team member reports to each other regarding the status of what they have done, what to do next, and any issues they are facing.

A feedback loop must be established, incormporating the cross-functional team. Other loops ma also be established to communicate with stakeholders.

# References

**There are no sources in the current document.**