Working Glossary

Acceptance Criteria or Acceptance Tests 3

Burn Rate 3

Burn Down Chart 3

Burn Up Chart 3

Claiming of Points 3

Typical Project Meetings 4

Standup 4

Pre-plan 4

Iteration Planning Meeting (IPM) 4

Showcase 4

Retrospective 4

Product Owner (or Customer) 5

Done 5

Gold plating 5

Inception 5

Iteration (or Sprint) 5

Iteration 0 6

Iteration Planning Meeting (IPM) 6

Master Story List (aka Product Backlog) 6

Narrative 6

Non-Story Work 7

Pairing (or pair programming) 7

Product Backlog 7

Re-design 7

Re-factoring 7

Release 7

Release Planning 8

Retrospective 8

Run Rate 8

Showcase (or Review) 8

Spike 8

Story 8

Story Point 9

Velocity 9

# Acceptance Criteria or Acceptance Tests

Acceptance Criteria provide additional criteria and boundaries for the Story from the customer’s perspective, focusing on what must be implemented. This may include things like business rules, validations, and inputs/outputs. A successful set of acceptance tests signifies the completion of a Story.

# Burn Rate

Rate at which some budget or fund (usually money) is spent/used on a project per unit of time. Could include consultant fees, expenses, and full time employee costs.

# Burn Down Chart

A burn-down chart is a graphical representation that shows the progress made during the development cycle.

The Burn Down Chart can be used to show outstanding work for a Release or for an Iteration and in both cases, the chart represents the amount of work remaining for the completion of the Release or Iteration versus time.

# Burn Up Chart

A burn-up chart tracks how much work is done. It generally shows more information than a burn-down chart because it also has a line showing how much work is in the project as whole (the scope as workload), and can indicate the point at which the project team’s projected progress will catch up to the scope line. Both scope and velocity can change.

# Claiming of Points

The claiming of points occurs within an Iteration. Developers demonstrate to Product Owners & QA that they have coded and successfully tested functionality to the details listed in the respective Story. The related Stories are made available to QA for testing. When QA has validated that the Story meets the stated Acceptance Criteria, then the Story Points can be claimed. One cannot claim partial points of a story, nor can partial Story points be carried into another Iteration. Either all the points for a Story are claimed within an Iteration or the entire Story (with points) is placed in the Product Backlog for re-prioritization. [See Story Points for a definition of that term.]

# Typical Project Meetings

## Standup

15 mins maximum, daily. Each team member identifies what they did the day before, what they will do today and most importantly they identify any risks and/or impediments

## Pre-plan

~1hr, once per Iteration. Product Owner, Iteration Manager, Business Analyst and Developers discuss possible Stories for the next Iteration. Developers can start tasking and Spiking areas of technical risk. May split into more than one meeting if Stories are complex or further discussion is needed. Development team typically sends a tech and BA representative instead of the entire team attending.

## Iteration Planning Meeting (IPM)

~1hr, once per Iteration. Team commits to the work for the next Iteration. They compare capacity to the estimated work and confirm that the stories that were identified are the ones for the team to work on in this Iteration. Developers may "task" the stories and refine the estimates.

## Showcase

~1hr, once per Iteration. Development team shows Iteration results to Product Owners and other stakeholders. Agenda usually consists of Review of past iteration, Demo of new functionality, Review of next iteration, Review of any risks/issues.

## Retrospective

~1hr. Team review of how things worked in the Iteration or some defined period - what to keep, what to stop doing, time to thank folks, call out new ideas. Iteration Manager facilitates or teams can rotate facilitation or teams can ‘borrow’ an external facilitators.

# Product Owner (or Customer)

Represents the business stakeholders of the project. Is responsible for:

* Facilitation of multiple stakeholders requests
* Writing Stories (including Narrative and Acceptance Criteria)
* Owning Stories
* Prioritizing Stories
* Explaining requirements to developers (or through analyst)
* Defining and passing acceptance tests (or through analyst)

# Done

Done is defined by the team to describe the point in which the team can claim points for a completed story card. Most teams define ‘done’ as the point in which it has been reviewed by the BA, the Product Owner and then passes QA review.

# Gold plating

Adding more to the system than specified in the stories (e.g. building them an Audi instead of the Kia they really want and need). Functionality should hit the business need - watch out for adding on to features / functionality that don't add real value or that stray from the original minimal feature set.

# Inception

The collaborative planning exercise, that takes place prior to development of a new project or set of features. The main deliverables are: prioritized stories, a Release plan, a staffing plan, a candidate architecture and an estimate for the work.

# Iteration (or Sprint)

A time-boxed period during which work happens. We recommend iterations are 2 weeks in length, starting on Wednesdays.

# Iteration 0

Iteration 0 is the agile term for the initial period necessary between when the project is scoped and development begins. It is the ramp-up time for the team in terms of infrastructure, builds, test environments, team building, agile process establishment and getting the first stories ready for Iteration 1.

# Iteration Planning Meeting (IPM)

Detailed scheduling of stories for the present Iteration. Stories are presented by the Product Owner and discussed by the team. Estimates are reviewed. May also include task planning by the team. Concludes with agreement from the team on the work to be done in this Iteration.

# Master Story List (aka Product Backlog)

The Master Story list (Product Backlog) holds the stories for the project. As new stories are discovered, they are placed on the list and marked with the Release it is felt they should best be handled.

# Narrative

Additional information appended to or written on a Story that provides much more detail than the Story itself. It is created by the Product Owner or BA and reviewed within the team before development work begins, but can be updated as implementation decisions are made. The following can be included in a narrative. Note that some attributes are mandatory:

1. Acceptance Criteria (Mandatory)
2. Items out of scope
3. Process Flows
4. Estimate (Mandatory)
5. Data Migration
6. Non-functional requirements
7. Screen shots or other visual depictions
8. Notes and references to existing documentation.

# Non-Story Work

Any work done by the team that is not specifically focused on meeting a business need is non-story work. This work is planned and tracked separately from stories to ensure clear identification and delineation of business-facing scope. This can include documentation, training, UAT, etc.

# Pairing (or pair programming)

A software development technique in which two programmers work together on a story or defect at one work station. One types in code while the other reviews each line of code as it is typed in. They regularly switch between driver and observer.

# Product Backlog

See “Master Story List”.

# Re-design

The process of re-working parts of the code to allow for smoother implementation or to support new functionality. Re-design typically requires design and discussions with team members to plan the best time for this work.

# Re-factoring

The process of changing a software system in such a way that it does not alter the external behavior of the code yet improves its internal structure. Refactoring should be a integral part of developing stories and should not alter the story estimate. **See Redesign.**

# Release

A set of developed and tested features that are put into a production (or production-like) environment for customer use. An **Interim Release** designates a ‘pre-production’ cut of code that is used for demos or end-user-like testing purposes such as UAT and Business Simulation Testing. Typically defines the minimal marketable feature set or the minimum viable product.

# Release Planning

Scheduling of stories for one or more Release. Could include initial scheduling of stories to Iterations. **See Inception.**

# Retrospective

A ritual held at the end of a project or Iteration to learn from the experience and to plan changes for the next effort. The premise for Retrospectives is that everyone has done the best they could to move the project forward.

# Run Rate

Similar to Burn Rate, but sometimes implies forecasted rate as opposed to actual rate

# Showcase (or Review)

The point at which the Product Owners and other stakeholders review and agree the working (built and tested) software derived from the cards for the Iteration, noting any defects/changes required

# Spike

A development spike is a technical proof-of-concept. It is not intended to produce production-ready code. Instead it is to enable developers to either validate their estimates or test/prove their solutions.

# Story

A story (aka user story, story card, card) is the unit of work – or scope - for most agile projects, described from the point of view of a user to whom it is valuable. It generally represents some useful software to be written, but can describe other kinds of work. They generally include a title, a story number, an estimate, a priority, and of course a description (written in the “As an X I want to Y so that I can Z” format). Mike Cohn (User Stories Applied) says that a good story follows the INVEST principle in that it has the following attributes: Independent, Negotiable, Valuable (of business value to customers), Estimable, Small, and Testable

# Story Point

A Story Point is a measure of relative effort required to complete a story. We recommend the Fibonacci sequence to “size” our stories (1, 2, 3, 5, 8, 13, etc.). [See Claiming of Points for more information about when and how points are applied as a measure of progress.]

# Velocity

Velocity is the total number of points that a project team completes in an Iteration. Given a velocity and a list of estimated Stories, a team can project how long it will take to complete those Stories and to project Release dates. **Projected Velocity** is calculated by taking the average of the previous three iterations’ velocity.