**What is the difference between a theme, an epic, a feature, and a user story?**

**User Stories** are short requirements or requests written from the perspective of an end user. A user story (or just a “story”) is a specific task within an epic. For instance, we may have such user stories as “Sign Up with Email”, “Sign Up with Facebook or Google”, “Log In with Email”, “Log In with Facebook or Google”, “Forgot Password”, and “Log out” in the broad epic that can be called “Registration & Authentication”.

User stories should be short and clearly defined.

**User Story consists of 3 parts**

***Value statement****:* As a (user role) ***(Who)***, I want to (activity) ***(What)***, so that (business value)**(Why)**

***Acceptance criteria***: Given (context), when (action performed), then (observable consequences)

***Definition of done***: The Development Team prepares a checklist like a quality control process before presenting the work expected from them to the Product Owner. This list is common for all User Stories and all finished jobs must go through this procedure. This is called “DoD Definition of Done” for short. This list usually includes the following controls;

Unit tests

QA tests

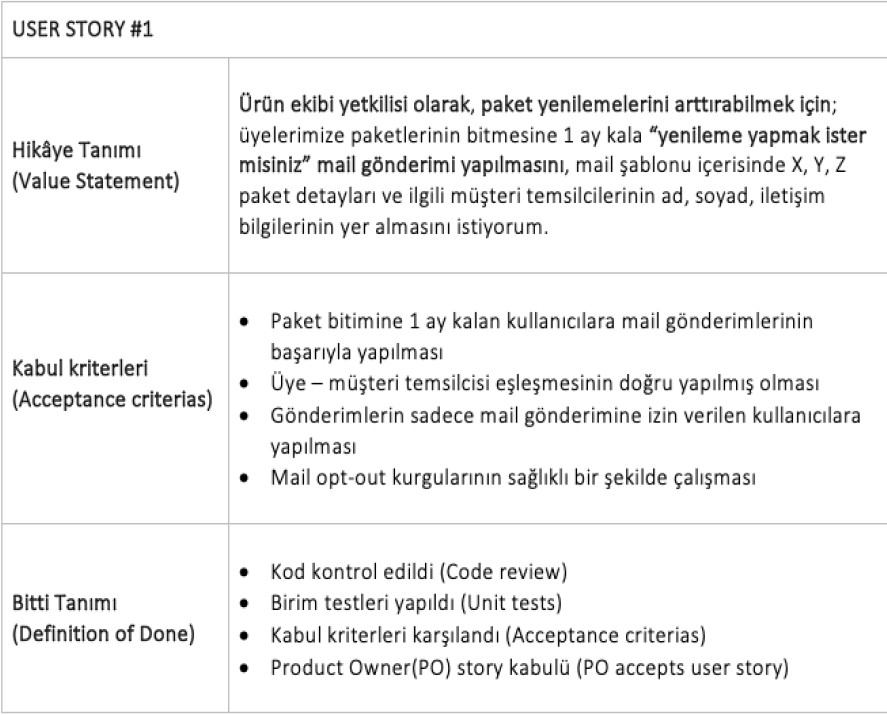
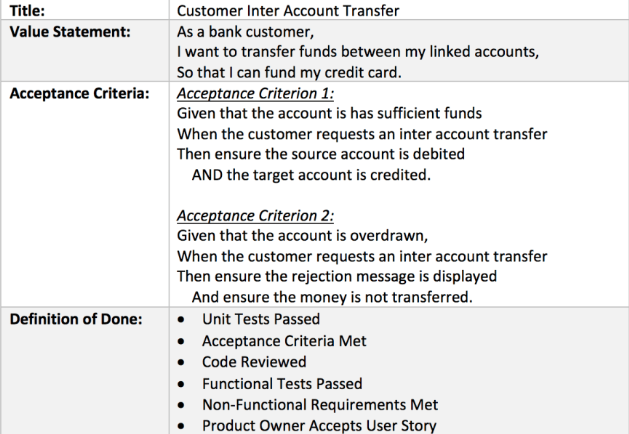
Updating all documents in User Story

Reviewing the code

Meeting the User Story acceptance criteria

Product Owner approval

The DoD checklist can also be prepared to be used to check whether certain completion criteria have been met at the end of the Sprint or Release.



**User Story Scoring**

User stories are scored by the team while they are included in the sprint backlog. At the sprint backlog meeting or refinement meeting, the User Story is read and all members of the development team give points to the user story. The points given are shown by writing on the cards. Points are given according to the numbers in the Fibonacci Series; 1 2 3 5 8 13 21 34

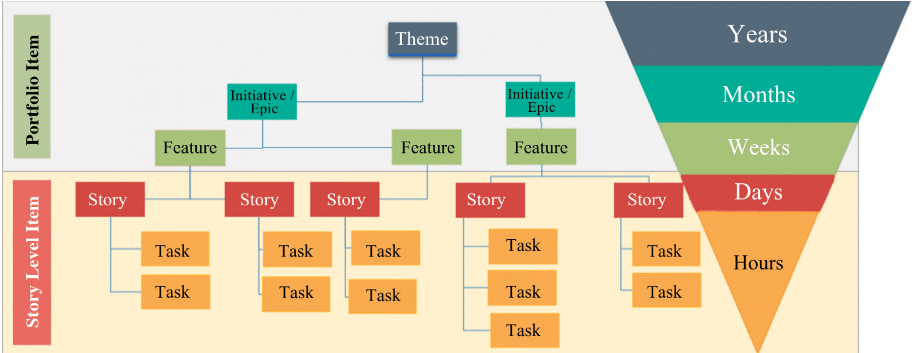
Each member explains why he gave this point and the point of the user story is determined by the joint decision of the team. Team capacity is determined according to the number of developers and QA in the team. 1 point is 1 day's work, that is 8 hours.

**Epics** are large bodies of work that can be broken down into a number of smaller tasks (called user stories). Epics is a grouping of one or more new functions or features that are planned to be made. Basically, an epic is one big piece of product functionality. Usually, it is too big to be completed in one sprint and should be split up into smaller bodies of work. For instance, an epic may constitute the pieces of code that are responsible for registration & authorization, authentication, user profile, etc. As simple as that.

**Feature**: These are modules chosen by the PO from Epics and added to the Product Backlog, each containing more than one user story and expected to be completed in more than one week. It is delivered to the end user as a finished product. Features could be defined as characteristics or events of the product or service.

**Initiatives** are collections of epics. Initiatives in agile management are parts of a theme. In other words, a theme is a set of initiatives united by a common goal. In the above example, under the theme of strength and safety, an initiative could be reducing the bridge vibration by 50% or reducing the building costs.

**Themes** are large focus areas that span the organization. Themes can be understood as larger focus areas that the whole organization identifies with. In our example, the larger objective of the project is to build a bridge. So the theme could be “strength and safety”. Themes in agile can also be defined as the high-level goals of the team as a whole.



**Product backlog:** a list of requirements provided by the customer in the form of Epic, Features, Stories, etc. generally captured in an excel sheet format.

**Epic**: Large level requirements/ needs of the business.

**Features**: Divisions of Epic, precisely defines the needs of the customer.

**Stories**: It refers to the detailed definition of each requirement/ Feature.

**Tasks**: Defines the solution of each Story. They are the actions taken on “Story”.

**Portfolio Backlog:** The Portfolio Backlog is an ordered list of all strategic themes. A portfolio backlog is similar to a product backlog; however, while a product backlog contains items relevant to only one product, a portfolio backlog describes multiple products, programs, or projects for which development has been approved but not yet begun.