

PHASE II | PART I | Guideline

Requirements

- Principle 1
 - requirements describe what needs to be done, input for design/implementation/management, but also an artifact for discussion, feedback, and clarification.
- Principle 2
 - requirements cannot eliminate discussion, the requirements engineering process does not replace project management (guidelines for quality assurance and control, some automated support, e.g., generating input/output test cases, documentation/contract).

Features

Features are simple to understand, testable, and have value (e.g., business, social). They directly explain what the user will be able to do while interacting with the service and therefore have to be grouped in themes depending on the functionality they provide to the user. **User Stories** (user-focused description) and **Use Cases** (interaction/technical description of feature) will then provide a clearer image of the features.

How to Features: Use the **Connextra notation** in order to describe a feature.

Example:

Feature name

As a [the role of the user]

So that [I can achieve some goal]

I want to [do some task]



We then want to specify a scenario in which we want to apply this feature. This then creates the user story.

Feature Search tweets by topics
As a Journalist
So that I can find tweets about topics for my monthly magazine
I want to search tweets by topics

Scenario:

GIVEN I am on WeChat interface of TwittertopicBot
WHEN I ask "search tweets about health budget"
THEN the bot should reply with a list of 10 tweets on "health budget"
WHEN I ask "show me more"
AND I press "More" button
THEN the bot should reply with 10 more tweets on "health budget"

Further Specifications

Once a feature is defined and a user story is ideated, and we know that the specific feature is in line with the SMART guideline (Specific, Measurable, Achievable, Relevant, Time-Boxed), we want also to specify how we will implement it. This means that we will specify whether we will have to use an API, reuse a component we already developed or which is available to us, or whether we have to create it anew.

Normally, with simple features like **logins** or **account registration**, in which we can reuse general components and the workflow is known by all stakeholders in advance, we won't be using an in-depth description with user stories or an extensive feature specification as it slows development unnecessarily.