

Use Case Template

Use Case	<i>Use case identifier and reference number and modification history</i>
Description	<i>Goal to be achieved by use case and sources for requirement</i>
Actors	<i>List of actors involved in use case</i>
Assumptions	<i>Conditions that must be true for use case to terminate successfully</i>
Steps	<i>Interactions between actors and system that are necessary to achieve goal</i>
Variations	<i>Any variations in the steps of a use case</i>
Non-Functionals	<i>List any non-functional requirements that the use case must meet</i>
Issues	<i>List of issues that remain to be resolved</i>

Use Case Template

Purpose of the Use Case Template

- ◆ Provide a graphical way to summarize the functionality of the system, to facilitate communication with stakeholders including customers and developers.
- ◆ Use cases capture *who* (actor) does *what* (interaction) with the system, for what *purpose* (goal), without dealing with system internals.

Guidelines for creating a Use Case

- ◆ Give the use case a memorable name
- ◆ Describe the goal that the use case is intended to accomplish
- ◆ List all the actors (human and other systems) that interact with the system in this use case
- ◆ State pre-conditions or assumptions
- ◆ List each of the steps in the use case. A step completes when all its component interactions have completed. A step has the form
 <sequence number><interaction>
If there are multiple steps, then each step must have an integer sequence number showing its position in the list of steps. Steps are initiated in order, in accordance with their sequence number.
- ◆ List any variations to the sequence.
- ◆ Specify non-functional requirements related to the use case.
- ◆ Identify any issues and other notes.
- ◆ See Use Case paper (Malan and Bredemeyer, 1999 in workshop binder) for more details on the Use Case Template.

Uses of Use Case Information

- ◆ *Capture functional requirements:* Use cases capture the functional requirements of the system. A representative set of use cases, covering the major goals of the users and important “corner cases” may be considered architecturally significant. Fine details of low-level use cases would not be.
- ◆ *Used in system structuring:* Early in structuring, use cases play a role to the extent that they enhance the architect’s domain knowledge. They are used in conceptual architecture when identifying related functionality, so that highly cohesive, tightly coupled responsibilities can be clustered together. During logical architecture, the steps in the use cases are fleshed out in collaboration diagrams (or sequence diagrams). This allows the architect to verify the component responsibilities and operations/methods identified so far, and to create new operations as necessary to support the use case step.
- ◆ *Used in validation:* The validation team validates that the architecture does indeed support the use cases in the requirements set, walking through collaboration diagrams for the use case steps. They may also bring up new use cases (e.g., for planned future systems), to assess the impact on the architecture.