

Title:

How to Write Good Use Cases for Useful Business Analysis

Word Count:

381

Summary:

A use case details a flow of events that are executed in order to accomplish some business task.

Keywords:

Business Analysis, Requirements Engineering, Business Requirements, Business Analysis Training, Requirements Training, Use Cases, Use Case Training, Requirements Elicitation, Requirements Documentation

Article Body:

A use case details a flow of events that are executed in order to accomplish some business task. A use case can be as simple as documenting how a help ticket gets escalated or as complex as defining how a customer gets charged for shipping parts of an order to multiple addresses.

The term "actor" is used to define a role that a person or some object plays in executing a use case. The actor might be a Customer Service Representative who is processing a refund request, or a server that processes credit card transactions.

Writing usable use cases is an excellent way to derive functional requirements and to the software development process as a whole.

Often times new analysts, or people finding themselves in an analyst's role with no training or experience, often wonder what should go into a good use case. Although some organizations may have strict requirements that conflict with the information presented in this article, you can safely assume that you'll cover all of the normal essential elements if you follow these tips:

- Use Descriptive Names

The name of the use case should leave no doubt as to its purpose. A name like "Contact Past Due Customers" is better than "Collections".

- Assign a Unique identifier to Each Use Case

Assigning an identifier or label to each use case makes it easier to refer to a

use case in other project artifacts.

- Create a use Case Summary

Creating an "executive" summary allows other stakeholders to gain a high-level understanding of the use case's purpose without having to read the entire document.

- Use Case Frequency

Document how often this use case is executed by the actors.

- Use Case Preconditions

Document any situations or conditions that must exist prior to the use case being invoked.

- Use Case Post-Conditions

Document any conditions that must be true or false after the use case is invoked.

- Basic Use Case Logic

Document the main or happy path that the user takes throughout the use case.

- Alternate Paths

Document all paths that exist as the result of exceptions or errors in the main or happy path.

- Group Memory

Document all critical decisions that were made during the creation of the use case in order to be able to ensure that the group memory is maintained.