

# Granularity of Use Case. Should sort/search be included?

Asked 12 years, 10 months ago   Modified 12 years, 10 months ago

Viewed 2k times



5



How do I determine what should I add to my use case diagrams? 1 for each button/form? Should things like sort and search be included? Or are they under "list items" for example? Though, a list of items seems understood?

uml

Share

Improve this question

Follow

asked Feb 2, 2012 at 11:15



Jiew Meng

87.9k ● 191 ● 522 ● 830

The normal granularity of usecase is the scenario goal not the step. – [Martin Spamer](#) Feb 6, 2012 at 13:25

2 Answers

Sorted by:

Highest score (default)



4

The Use Case diagram is intended to help define the high-level business tasks that are important, not a list of functions of the system. For example, a system for use in



customer service might involve a research task of looking up information to help someone on a support call.



Most of the literature describes Use Cases as a starting point for defining what the system needs to accomplish.



The temptation has always been to be as complete as possible; adding ever more details to define the use case down to a functional (code-wise) level. While it is useful to have a comprehensive understanding of the requirements, the Use Case diagram is not intended to provide that level of documentation.

One thing that makes the issue worse is the syntax which I've never seen used in a working project. It isn't that the terms aren't useful, it's due to the lack of consensus over when to use either term for a given use case. The UML artifacts expect a process that is more focused on the business language instead of the implementation language - and by that I do not mean a computer language. The tendency by some has been to approach the diagrams with a legalistic bent and worry about things like when to use for related use cases or how to express error-handling as exceptions to a defined list of process tasks.

If you have ever tried to work through the Automated Teller Machine (ATM) example, you'll know what I mean. In the solar system of UML learning, the ATM example is a black hole that will suck you into the details. Avoid using it to understand UML or the Object Oriented Analysis and Design. It has many of the problems, typical of real-world

domains, that distract from getting an overall understanding even though it would make for a good advanced study.

Yes, code will eventually be produced from the UML artifacts, but that does not mean they have to be debated like a treaty in the Senate.

Share Improve this answer

answered Feb 2, 2012 at 15:12

Follow



Kelly S. French

12.3k ● 10 ● 64 ● 93



1



The [OMG UML spec](#) says:

Use cases are a means for specifying required usages of a system. Typically, they are used to capture the requirements of a system, that is, what a system is supposed to do. The key concepts associated with use cases are actors, use cases, and the subject. The subject is the system under consideration to which the use cases apply. The users and any other systems that may interact with the subject are represented as actors. Actors always model entities that are outside the system.

The required behavior of the subject is specified by one or more use cases, which are defined according to the needs of actors. Strictly speaking, the term “use case” refers to a use

case type. An instance of a use case refers to an occurrence of the emergent behavior that conforms to the corresponding use case type. Such instances are often described by interaction specifications.

An actor specifies a role played by a user or any other system that interacts with the subject. (The term “role” is used informally here and does not necessarily imply the technical definition of that term found elsewhere in this specification.)

Now most people would agree that business and user level interactions are the sweet spot, but there is no limitation. Think about the actors/roles being outside of the main system/systems you are focusing on. But in one view a system could be an actor, but in another the implementer of other use cases.

Share Improve this answer

answered Feb 3, 2012 at 4:39

Follow



**Ted Johnson**

4,343 ● 3 ● 31 ● 31