**What is a use case diagram?**

PPT 4

Here are some questions that have been asked frequently in the UML world are: **What is a use case diagram? Why Use case diagram?** or simply, **Why use cases?**. Some people don't know what use case is, while the rest under-estimated the usefulness of use cases in developing a good software product. Is use case diagram underrated? I hope you will find the answer when finished reading this article.

So what is a use case diagram? A [UML](https://en.wikipedia.org/wiki/Unified_Modeling_Language) use case diagram is the primary form of system/software requirements for a new software program under developed. Use cases specify the expected behavior (what), and not the exact method of making it happen (how). Use cases once specified can be denoted both textual and visual representation (i.e. use case diagram). A key concept of use case modeling is that it helps us design a system from end user's perspective. It is an effective technique for communicating system behavior in the user's terms by specifying all externally visible system behavior.

PPT 5

A use case diagram is usually simple. It does not show the detail of the use cases:

* It only summarizes **some of the relationships** between use cases, actors, and systems.
* It does **not show the order** in which steps are performed to achieve the goals of each use case.

As said, a use case diagram should be simple and contains only a few shapes. If yours contain more than 20 use cases, you are probably misusing use case diagram.

As see in the diagram of this slide, shows the UML diagram hierarchy and the positioning of UML Use Case Diagram. As you can see, use case diagrams belong to the family of behavioral diagrams. There are many different UML diagrams that serve different purposes (as you can see from the UML diagram tree above). You can describe those details in other UML diagram types and documents, and have them be linked from use cases. Use cases represent only functional requirements of a system. Other requirements such as business rules, quality of service requirements, and implementation constraints must be represented separately, again, with other UML diagrams.

PPT 6

Use case diagrams are typically develop in early stage of development and people often apply use case modeling for the following purposes:

* Specify the context of a system
* Capture the requirements of a system
* Validate a systems architecture
* Drive implementation and generate test cases
* Developed by analysts together with domain experts

The component of use case diagram are include:

**Actor**

Someone interacts with use case (system function). Named by noun. Actor plays a role in the business. Similar to the concept of user, but a user can play different roles. Actor triggers use case(s). Actor has responsibility toward the system (inputs), and Actor have expectations from the system (outputs).

**Use Case**

System function (process - automated or manual) Named by verb + Noun (or Noun Phrase). i.e. Do something. Each Actor must be linked to a use case, while some use cases may not be linked to actors.

**Communication Link**

The participation of an actor in a use case is shown by connecting a actor to a use case by a solid link. Actors may be connected to use cases by associations, indicating that the actor and the use case communicate with one another using messages.

**Boundary of system**

The system boundary is potentially the entire system as defined in the requirements document. For large and complex systems, each modules may be the system boundary. For example, for an ERP system for an organization, each of the modules such as personal, payroll, accounting, etc. can form a system boundary for use cases specific to each of these business functions. The entire system can span all of these modules depicting the overall system boundary.

Use cases share different kinds of relationships. Defining the relationship between two use cases is the decision of the software analysts of the use case diagram. A relationship between two use cases is basically modeling the dependency between the two use cases. Reuse of an existing use case by using different types of relationships reduces the overall effort required in developing a system.

Use case granularity refers to the way in which information is organized within use case specifications, and to some extent, the level of detail at which they are written. Achieving the right level of use case granularity eases communication between stakeholders and developers and improves project planning.

PPT 7

A **use case** is a static description of some way in which a system or a business is used, by its customers, its users or by other systems. A **use case diagram** shows how system use cases are related to each other and how the users can get at them. Each bubble on a use case diagram represents a use case and each stick person represents a user.

There are several different ways of describing alternative paths through a use case. All routes through the use case- what normally happens, minor variations and significant alternatives- can be illustrated in detail using scenarios. Minor variations can be documented as alternative courses in the use case description. More significant differences can be modelled as separate use cases linked to the original by an <<extend~ relationship. Alternatively, a specialized use case can be used to model different behavior within a use case. Common functionality can be extracted into a separate use case linked by an <<include~ relationship to any use case that needs this functionality. The use case model is a useful starting point for identifying classes and forms the basis of the interaction diagrams. It is also useful as a guide for project management, for checking requirements and for system testing.

PPT 8

**Conclusion**

Use cases model the user~ view of the functionality of a system. The use case model consists of a use case diagram, supported by textual descriptions, use case and actor descriptions, and scenarios. Both the diagrams and the supporting text are simple and intuitive which makes them an ideal vehicle for discussions with the user and for clarifying the developer~ understanding of the users requirements. Actors identified in the use case diagram represent users who interact with the system in some way, who use the system to achieve a particular task. Each use case represents a task or major chunk of functionality. The use case description describes the task in more detail. The description views the task from the user~ perspective, it does not attempt to provide a program specification.

References

1. Carol Britton and Jill Doake, (2005), A Student Guide to Object-Oriented Development, ISBN 0 7506 61232, Elsevier Butterworth-Heinemann Linacre House, , Oxford, Burlington MA o 1803
2. <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/> (2019)
3. O’Docherty, Mike. (2009), Object-Oriented Analysis and Design : Understanding System Development with UML 2.0, ISBN-13 978-0-470-09240-8, John Wiley & Sons Ltd, West Sussex PO19 8SQ, England
4. Kendall, Kenneth E., (2011), Systems Analysis and Design , 8th ed, Pearson Education, Inc., ISBN-13: 978-0-13-608916-2, Prentice Hall, One Lake Street, Upper Saddle River, New Jersey 07458