

6 Use Case Modeling and User Interface Design

Use Case Modeling is a powerful modeling technique that can be used in different stages of system and software development. Its understandability by technical and non-technical people alongside its simplicity, high flexibility, and readability has made it a first choice to capture and organize requirements. It can also be used in analysis, design, and testing processes. Use case modeling is supported by Unified Modeling Language (UML) as well.

Use case modeling plays a great role in capturing requirements, as well as in presenting the behavior of the system. It can be used both in understanding the current system, whether it is manual or automated, and to model a new system. Having both diagrammatical view and explanatory document, makes it an excellent tool for communication between developers and users. Again, like other subjects in this book I will explain the concept of Use Case Modeling and User Interface Design in the context of final year projects.

6.1 Use Case

A use case has two formats. The first one is a diagrammatic format, which includes a watermelon shape, a matchstick man, and a line between these two. It depicts a major functionality in the system and the way that a user, which is called an actor, would communicate to the system to acquire the mentioned function. The second format is a textual description, which explains what will be happening when the actor (user) requeststo use the “use case”, in other words, what would be the case of using system by the actor.

Usingthe examples of the previous chapters, you can find out a Use Case, which corresponds to the first requirementof the sample 1 in the chapter 5 inFigure 6-1.

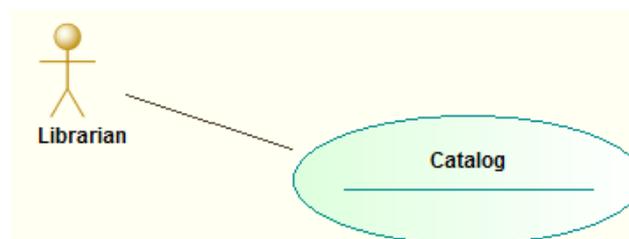


Figure 6-1 Use Case sample (diagrammatic view)

These simple items, as you have learnt before, show that a Librarian communicates with a system to maintain the library catalog through the Catalog use case. Figure 6-1 cannot tell you how this happens though. This is not its responsibility to do so. In order to use case to be able to tell about the detail of its functionality, it should be accompanied by a textual explanation, which will be discussed in the next section.

6.2 Use case Model

Use case modeling is a dynamic activity. You would identify use cases and actors and you would put them on the context. As a first attempt on the sample 1 of our requirements in chapter 5, you may come up with the first-cut use case model as you can see in Figure 6-2.

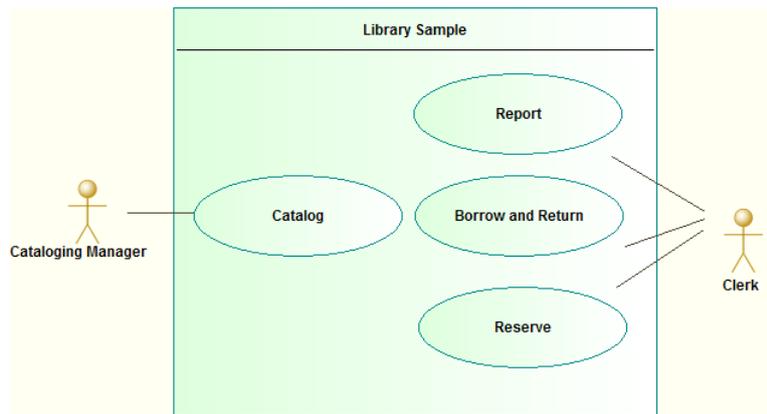


Figure 6-2 Use Case Model (sample 1)

But, as you can see in section 6.5 this model may need some refinement. I would like to repeat that use case modeling, and perhaps all modeling activities, is a dynamic activity, which you would build your model through consecutive steps until it fits your case.

6.3 Use Case Template

This explanation should follow a template. You can find many templates for use case and perhaps you have seen some up to this point of your study. Most of these templates have several items in common, which together forms their main characteristics. Some of these templates are too detailed and some are very simple. Below I will introduce a template, which I have found suitable for your final year project.

Use Case Document Template	
ID:	Give an identification number that enables you to make the use case traceable.
Name:	The name that you have used in the use case model.
Aim:	The aim of use case.
Main Actor:	
Pre-condition:	What is the expected situation before the use case can be started?
Main Scenario:	Main scenario, which use case performs when it is started.
Alternative Scenarios:	Alternative scenarios, if there is any.
Extension Points:	Other use case to which this use case can extend.
Post condition:	What is the expected situation after the use case is finished.

Here is my specific advice on how to use this template. Add your user interface design to the above template where you intend to explain the use case scenarios. What does this mean? I will explain this in the next section.

6.4 User Interface and Use Case

Assume that you want to document the use case in Figure 6-1. It would be very helpful your customer/supervisor, and for you/developer as well, to show a similar window (i.e. dialog box, grid, table, tabular sheets, etc.) that you are planning to use as user interface, when you are describing it. This way you would give a real sense of what is happening to your documents reader. In addition, you can role-play what is happening during the use case course. Furthermore, you can focus on the possible data entry as well. I can add more to these pros of combining user interface and embed it into use case. But, it is better for you to try it and see how it works.

In brief, you can design your user interface using the same tool that you are planning to use in the implementation phase and embed snapshots of which in your use case document. Otherwise, you can use other designing tools such as Microsoft Visio to design screen samples and embed them in the use case document, instead. Well, I prefer the former choice, anyway, but it is up to you to choose the one that you prefer.

6.5 Use Case Granularity

How many use cases you should have in your system? This question should be answered carefully. The answer depends on many factors such as the scope of the system, the complexity of the system, the relations with other system, and some other factors. However, some rules of thumb can help us to stay on the right track. Usually, in a project of the size of your final year project, you might have between 5 to 15 use cases.

If you have less than 5 use cases, then it seems that something is wrong. It can be the size of the system, which means that it is not a full system, rather it is a part of a system, or it can be from not understanding use case modeling properly. Whichever is the case, it would be better if you think about it again and to consult with your supervisor about it.

If you have more than 15 use case, I do not think can be because of your project size, rather it is because of your misunderstanding of the definition of use cases. Many students get confused about use cases. Sometimes experts do the same. Anyway, look at your use cases again and make sure that you have not taken a single step as a use case. Use case should fulfill part of a requirement, a requirement, or more than one requirement. But, in any case, it starts from a stable situation of the system and leaves system in a stable situation, when it finishes.

Nonetheless, there are situations that either you have less than 5 or more than 15 use cases and you cannot find any serious problems in your modeling. Well, here I should introduce you a concept, which is called “use case granularity”. If your use cases number is less than 5, and you have a real system, then if it is helpful in managing the system, both from development perspective and from functionality perspective, then try to decompose one or more use cases into other use cases to increase the system granularity. If your use cases number is more than 15, then try to identify or define some subsystems to which you assign use case which are more related to each other. Normally, a subsystem is better not to include more than 10 use case.

Figure 6-3 shows the use case model of Figure 6-2 on which the granularity concept has been considered. However, this model still can be decomposed more and even to be redesigned to include at least three subsystems such as Catalog, Membership, and Borrowing, each of which can include a couple of use cases.

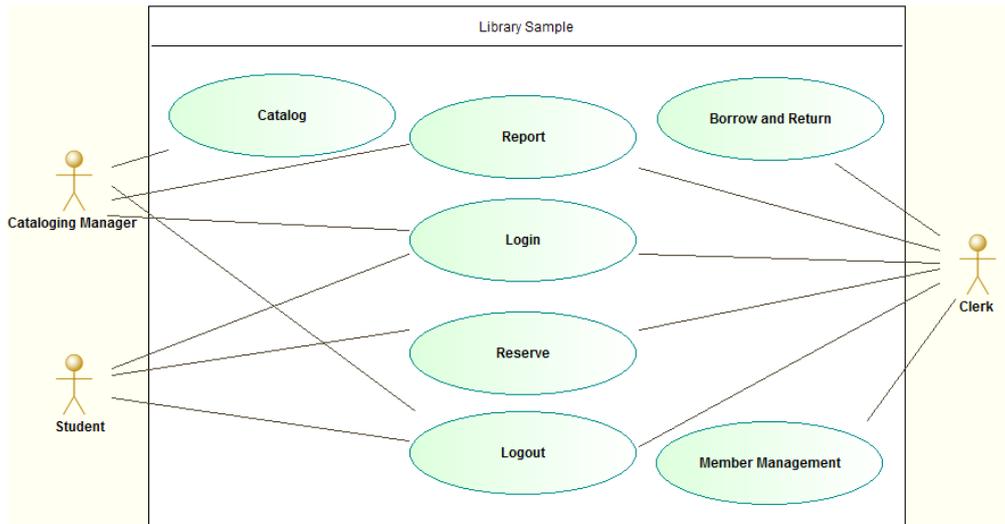


Figure 6-3 Use Case Model (sample 1 – revised)

6.6 More applications of Use Case

As it was mentioned, the main purpose of use cases is to capture requirements and then to use them in the analysis in order to provide a more understandable view of the system, both for developers and customers. However, use case can be used in other stages of the system and software development. One case was user interface design, which was discussed earlier in this chapter. In addition, use case can be used in the testing process. Moreover, they can be used for users guide and system help preparation. The scenarios that you have already explained, and the screen snapshots that you have already embedded into use case explanations can act as a rough material for the mentioned purposes. Furthermore, use case documents can be used as rough material to prepare training documents, as well. Now, can you tell me how amazing this use case thing is? Thank you Ivar Jacobson, for introducing this excellent idea!

6.7 Summary

Use Case Modeling helps developers to capture, organize, and manage requirements. In addition, it is a useful tool in other development stages such as analysis, design, and testing. It can be used in users guide preparing and even in user training process.

Use case is the way that user/actor communicates with the system. Actor asks system to perform a function and system responds to the actor's request through a course of actions. Use case modeling has two perspectives, one is diagrammatic, and the other is textual. In order to simplify the use case explanation, a template was introduced. In addition, it was suggested to combine user interfaces with use case explanation in order to make it easier for all people who are involved in the system development process to understand the analysis and design results. Finally, the use case granularity concept was discussed and some rules of thumb to manage the use case granularity were provided.