Use Case Discovery Tips

1. Discover use cases by considering what each actor requires of the system. Remember the system exists only for its users therefore, it should be based on the users' needs. If the requirements specification contains functional requirements for the system, many of the users' needs may already be visible. For each actor, human or not, ask yourself the following questions:

   - What are the primary tasks that the actor wants the system to perform?
   - Will the actor create, store, change, remove, or read data in the system?
   - Will the actor need to inform the system about sudden external changes?
   - Does the actor need to be informed of certain events in the system?

2. The answers to the above questions represent the flow of events that identify use case candidates. Not all use case candidates constitute separate use cases; some of them may be modeled as variants of one and the same use case. However, sometimes it's not easy to tell what is a variant and what is, in fact, a separate and distinct use case. Do not worry about this now, it will be clearer as you describe the flow of events in detail.

3. An information system is a system that is integrated into the operations of a company. As a rule, a company will have conducted analysis of its enterprise, which is documented in some business processing engineering (BPE) effort. By describing how the new information system might be incorporated into the existing operations, this model will provide valuable input for the system development. BPE documentation is a good source of use cases, since it will probably clarify how the system should be used.

4. Most information systems being built have a few major objects/entities upon which use cases can be based. For example, a Policy Writing System has client and policy, and a Claims System has policy and claim. When trying to discover use cases it may be helpful to generate a state diagram/entity lifecycle for each major object/entity. The state diagram/entity lifecycle would view the object/entity from a business perspective and determine the events that cause the object/entity to change from state to state.

5. An information system embodies a set of rules that are used to run the business. These rules should be reviewed to discover potential use cases.

6. Be aware of finding and developing use cases that are CRUD based. Sometimes it may make sense to have create, retrieve, update, and delete use cases, but we believe this to be the exception rather than the rule. Ask yourself, "what does the actor really want the system to do?" or "how does the system enable the actor to perform their job?"

7. There is not one single use case model possible for a system. Indeed, there may be several feasible use case models for the same system. The optimal course of action in finding the "best" model is to develop two or three alternative models, then choose the one you prefer and develop it further. Developing several alternative models also helps you to understand the system better.

8. When you have made your first outline of the use cases, verify that all the functional requirements in the requirements specification have been addressed in the use case model. Scrutinize the specification carefully to ensure that all the requirements have been met by the various use cases.