Demystifying Use Case Modeling

Elizabeth Larson, PMP, CBAP and Richard Larson, PMP, CBAP
Principals, Watermark Learning, Inc.

Industry Article

Watermark Learning

ENHANCED PERFORMANCE. ENDURING RESULTS.

Demystifying Use Case Modeling

By Elizabeth Larson, PMP, CBAP and Richard Larson, PMP, CBAP
Principals, Watermark Learning, Inc.

We all know how difficult it is to achieve project success without complete product requirements. Yet gathering complete requirements without exhausting the project schedule and budget remains elusive for many project managers. In this short article we will provide tips for gathering hidden requirements quickly.

Traditionally, many software developers have embraced new technology. As flat files gave way to hierarchical databases and as those databases gave way to relational tables, as mainframe technology became distributed and as object oriented technology replaced information engineering, many developers jumped at the opportunity to learn new skills. Project managers, however, tended to be more wary. How this new technology would affect schedules and budgets, what the relationship would be between the technology and project risk and quality, how easy it would be to hire staff with technical expertise became tough questions for these managers to answer.

With new technology came new methods known as development methodologies. We went from the Waterfall methodology to the Whale model, to Rapid Application Development to Iterative Project Management, and each new process claimed to reduce the software development cycle time. Each has had both advantages and disadvantages, but none has proven totally satisfactory. One reason for the disillusion is that none addresses gathering and managing requirements both quickly and thoroughly.

One proven approach for quick and complete requirements-gathering is what I call "concurrent modeling." This approach is different from concurrent component engineering, or concurrency, commonly used in developing e-solutions. Concurrent modeling suggests that by modeling business data, business processes, system processes through use case modeling, and prototyping, requirements quickly surface. In addition, each type of modeling effort supports the other modeling efforts and forces analysts to ask their customers the kinds of pertinent questions that drive out hidden requirements.

While modeling requirements with use cases does not provide a complete solution, it does provide one important dimension of requirements-gathering—that of describing the conversation between a system and its user(s). Although this system is usually automated (such as an Order system), it can be a system in its broadest sense, comprised of methods, procedures, forms, and automated systems. Another way to think of a use case is that it describes what a system does in response to a request or a trigger. If I want my car to stop, I need to have a 'conversation' with it and make that request. Today I need to talk my car's language; I cannot simply shout 'STOP!' I need to use an interface to state my request. That interface is the brake. Hopefully the car responds by slowing down and coming to a halt.

Use cases have their roots in Information Engineering, which twenty years ago had such proponents as Ed Yourdon and James Martin. What we then called event or process modeling, we now call use case modeling. 'External agents,' became what we now call 'actors.' Data flowing into and out of the system and processes within the system we now think of in terms of interfaces. We now call the hierarchy of system processes use cases.

Many practitioners confuse this "system" process model with a "business" process model, or process map. Both use cases and process maps describe processes. The former describes the reaction of a system to an external trigger. Process maps also describe processes, but from the perspective of who does what in the organization. In both cases the process is described with one verb and one noun, known as an action and an object, and in both instances inputs are transformed into outputs.

An example of a process is 'Check Inventory.' From a business process perspective, checking inventory might include a business person examining actual items, perhaps taking a physical count of the inventory, scanning cartons or items, and comparing the items on-hand to a report. In use cases modeling 'Check Inventory' might describe how an Order system (actor) queries the Inventory Management System to see if the requested items are in stock and then reserves the items, all without human interaction.

One of the most commonly asked questions from my students is why they need to do modeling other than use case modeling. I answer as follows:

- Data models provide what information appears on user interfaces (UI's), for both data entry screens, as well as reports. It also provides many of the business rules, e.g., whether or not customers are required to have accounts.
- Process maps provide UI navigation, which should follow the business processes (hopefully improved or reengineered). These maps also drive out business rules, e.g., we process deposits before withdrawals.
- Prototyping, which is derived from both the data and the process models, allows for early feedback and helps drive out additional requirements.
- Use case models not only show system interfaces, but can lead to a description of edits and messages and testing scripts. They also become the basis for software program design.

It's no wonder, then, that one of the most common complaints from my students is that they have done a thorough job of use case modeling and yet requirements still surface throughout and after the project!

So to help achieve success on software projects, focus not just on the project effort, but on the product requirements. And remember that concurrent modeling will reduce the risk of schedule and budget overruns, since hidden requirements will surface sooner in the development process, reducing the cost of rework during the entire project and, importantly, after the project is implemented.

About Watermark Learning

Watermark Learning helps improve project success with outstanding project management and business analysis training and mentoring. We foster results through our unique blend of industry best practices, a practical approach, and an engaging delivery. We convey retainable real-world skills, to motivate and enhance staff performance, adding up to enduring results.

With our academic partner, Auburn University, Watermark Learning provides Masters Certificate Programs to help organizations be more productive, and assist individuals in their professional growth. Watermark is a PMI Global Registered Education Provider, and an IIBA Endorsed Education Provider.

Contact us at **800-646-9362** or at **www.watermarklearning.com**.

Watermark Learning
ENHANCED PERFORMANCE. ENDURING RESULTS.