Sharing Requirements for Fun and Profit

by Mickey Hadick

June 7, 2010

Requirements Management is an emerging field whose beginnings pre-date the modern computer era. Technical projects have always needed them in order to complete the solution. Like Satan himself, requirements have been published under a number of names but the essence is always the same: the best way for the technical wizard to assure the buyer that he will solve the intended problem is to write down what he plans on building, and to share his plan with the buyer. It is the beginnings of a technical contract, and a document that describes the project requirements is what convinces the buyer to part with his first few payments.

The recent RQNG article, "The Big Freakin' Requirements Document Must Die. Here's Why." by Chris Gurney, had one very brilliant point in that it is foolhardy to use a word processing tool, like Microsoft Word, to gather, manage, and engineer requirements collaboratively. Word offers very little of the functionality needed to create requirements beyond the outline mode and applying styles with shortcut keys. You have my sympathies if you are forced to develop requirements with a word processor.

Nevertheless, I believe the concept of a "document" should be preserved even as requirements management tools become mainstream. The one very nice concept that a "document" provides is that it provides a snapshot in time representing a version of a set of requirements. In our increasingly agile world with constantly shifting priorities and needs, it is important to respond to the users and change a technical project's course accordingly. It is also important to keep track of where you have been, and blazing a trail in the forest is best done with concrete markers. Respond to needs, of course, but track where you've been.

Some may bristle at the notion of creating a physical document when using a requirements management tool. I agree. My current tool, CaliberRM™, has a nice baseline feature that allows us to take a snapshot and preserve a set of requirements. I can move ahead and change anything I want, but a particular baseline stays in the same state like a museum piece behind glass.

Where our particular version of CaliberRM fails (we are trapped for the moment on an older version) is to share the requirements. The best technique is to create a Word document. It is awkward, at best, to review anything more than a few requirements with the tool, and for anything longer it's best to provide an MS Word collection of the requirements.

I have other reasons to defend the use of a standalone document. I believe it provides a motivation for users to read and review simply for its form. People are familiar with Word documents in a business setting. They know what to do with it just as you know what to do with a newspaper in a coffee shop, or a magazine at the dentist's office. If you send a Word document or a PDF to someone as an attachment

---

in e-mail, they click on it and read it. If your opening paragraph is compelling, they may just read the whole darned thing.

Requirements documents should only be large enough to convince the buyer that the planned solution will actually solve his problem, and should be no larger than that. If the problem is complex, the solution will likely be complex, as well. The complexity of the presentation of that solution, however, should be minimized. A table of contents is warranted, for instance, only if the document is larger than a few pages. I use the table of contents only to list the major sections--User Requirements, Functional Requirements, etc.-- but not the sub-sections and never the requirements. That would make it a cluttered list and not very useful to the reader. Even for documents of a dozen pages, I will omit the table of contents if the list of requirements are all focused on the details of an otherwise straight-forward solution.

I would not consider using Microsoft Word as a requirements management tool. I have done so in the past, and I do not want to go back to that place. In that regard, I most emphatically agree that the big, freaking, requirements document must die. A wiki hosted on a free server is a better tool than Word for collaboratively managing requirements. In fact, a MediaWiki based web server allows granular definition and editing of portions of a web page using the outlining feature built into its text-based editor, thanks to it markdown syntax.

The key is to create content first and think about how best to deliver that content later. There are a number of tools in the marketplace, some free, some by subscription, some expensive, and any of them are better than a word processor. CaliberRM has the feature to pull the requirements into a Word document; it's clumsy, but it works. MediaWiki has an option to display a web page in a ready-to-print layout, and that could be delivered for consumption or copied to a more formal document if necessary.

For the requirements to be useful in helping solve the problem, it's important to involve the client (the person with the business problem) in the requirements development process from the beginning and with every successive draft of the requirements. To enhance their pleasure in the exercise, which I believe increases engagement in the requirements process, make the requirements document fit the client's preference for viewing. If they are old school and not yet comfortable with computers, place a printed document on their desk; if they like Word or PDF documents, then attach it to an e-mail and encourage them to have a nice day in the signature line of that e-mail; and if you are so fortunate as to have a connected, savvy, and enlightened client, then refer them to a link to your web-based requirements management tool (even if it's just a wiki) and ask them to review the web page and make comments.

Reports of the death of physical requirements documents may be premature. I don't wish to see Word used any more than it has to be. Word processing has its place, but developing requirements is not that place. The requirements process also has its place, and that is to be part of the process to solve a business problem. Accommodating the quirky needs of your client in order to reach consensus on requirements is also part of that process, so encourage them to be enlightened about collaboration, but if they need to see a document then by all means give them a document.

Mickey Hadick is a Business Analyst with 25 years of experience working in I.T., and has been involved in pre-sales support, programming, analysis, administration, and project management. He blogs about a variety of topics at http://www.mickeyhadick.com, and you can connect with him professionally on LinkedIn at http://www.linkedin.com/in/mickeyhadick.