

Defining The Detailed Scope: How And Where Do You Find Requirements?

By Jamal Moustafaev, MBA, MBA, PMP
President & CEO
Thinktank Consulting

Roses, Stars And The Sun



Edward's banner

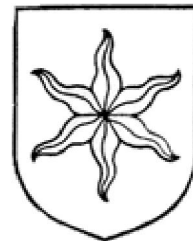
Medieval England. War of Roses. On April 13th, 1471 Yorkist troops under the command of the king Edward IV and Lancastrian forces led by Richard Neville, 16th Earl of Warwick and John de Vere, 13th Earl of Oxford met near the town of Barnet some twelve miles North of London. At four o'clock in the morning on April 14th the soldiers of both armies were awakened and started preparing for the decisive battle. Warwick's army heavily outnumbered Edward's, although sources differ on exact numbers. However according to some historians Lancastrian strength ranged from 10,000 to 30,000 men, with 7,000 to 15,000 on the Yorkist side.

The major problem that both armies had to deal with was a heavy fog that covered the entire battlefield thus making any kind of monitoring or communications with their units somewhat problematic.

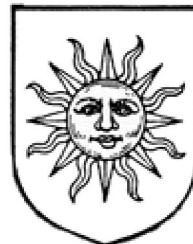
Medieval battles have never been known for their orderliness and organization; however because of the heavy fog, the confusion that existed on the field on that particular day dwarfed anything that was seen in military campaigns before and since that memorable day. In this mayhem at one point of time John de Vere's forces ended up behind their allies, a part of Lancastrian force led by Warwick's younger brother John Neville. Neville's regiments for reasons to be explained a bit later mistook their comrades-in-arms for enemies – Edward's reserve forces – and unleashed a volley of arrows on them. De Vere, in his turn quite logically assumed treachery and attacked Neville's troops ... The cries of treason quickly spread throughout the entire battlefield and as the fog started to dissipate, Edward saw the Lancastrian centre in disarray and sent in his reserves, hastening its collapse. One by one, first John Neville, then de Vere and finally Warwick were killed by Yorkists.

Some historians claim that as many as 6,500 Lancastrians perished in that engagement – a mind-boggling number of casualties by the 15th century standards. As for Edward, he retained his crown and ruled England for the next twelve years.

We now arrive at the most curious part of this story: how was it possible for Neville's forces to mistake their allies for their enemies? The answer lies in the fact that 15th century armies did not



Éstoile



Sun In Splendor

have uniforms; the only way to tell enemy from foe was by the heraldic banners each troop carried into the battle. And here is the kicker: Edward's coat of arms contained a depiction of the sun with several wavy rays – "Sun in Splendour", while John de Vere, had an "Étoile" (star with six wavy rays) on his crest. Unfortunately the bulk of John Neville's regiment consisted of illiterate peasants who had no training in heraldry whatsoever; at the time this discipline was typically reserved for someone belonging to the higher echelons of British society. It is not surprising, therefore that they confused these heraldic signs and reported back to Neville that "the enemy is behind us!"

The project management lesson of this story is that when running projects, especially large and complicated ones, no little detail of the project scope can be overlooked. Ignoring these "little and insignificant" details can lead to disastrous results like the one mentioned in the story above. Accordingly I would like to dedicate this article to the problems and issues of the detailed scope definition that has been somewhat overlooked in the current project management science.

Introduction: More Detailed Scope Definition

In the article titled "Kick-Starting Your Projects – What Should You Know About Defining Scope?" we have already discussed the elicitation of high-level scope requirements that would probably appear in the Project Charter or some other auxiliary scope document. At this stage however we are zeroing-in on the next step, the following iteration of scope definition that happens some time after the sign-off of the project charter but before the technical team members start working on the final designs, blueprints and bills of materials.

Eliciting Detailed Scope

Who Should You Talk To?

A question that gets asked frequently, especially on large and complicated projects is: "Who should I include in scope discussions?" This is a very complicated matter since a good number of projects I have known suffered setbacks in the form of cost overruns and missed deadlines because the right group of people (or even single person) were not consulted properly at the scope definition stage.

The first collective group of primary stakeholders are the clients (aka "sponsor"), customers and the future users of the product or service you are trying to build.

Clients have the final say in the product scope discussions with respect to what the product does, how it does it and how sophisticated or simple it should be because they are the ones financing the project.

Customers are also important because they (hopefully!) will pay for your product and walk out of the stores with it under their arms. Or they alternatively may ignore it and decide to buy the competitor's product. Therefore it is very important that you understand their real needs.

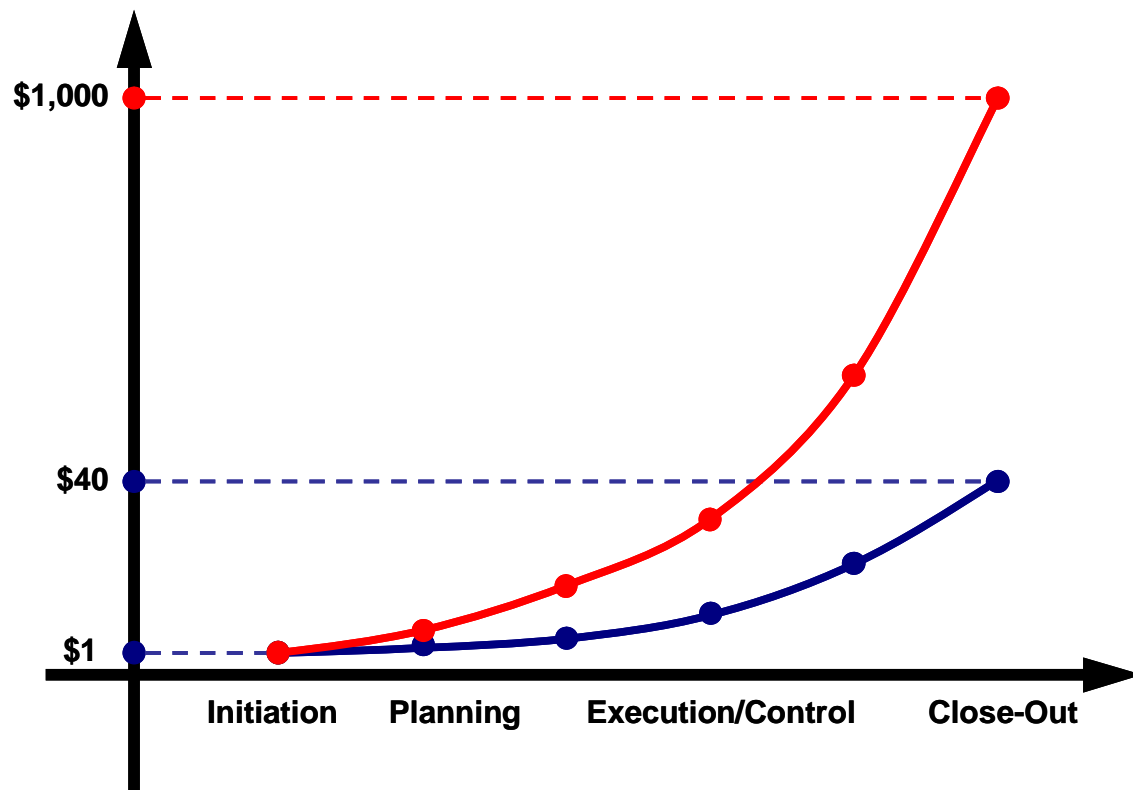
End users could be a different group from customers; i.e. purchasing can be done by one group of people and actual usage by another (think of millions of parents buying video games for their kids during the Christmas season). On a more serious note, it is typically the users of the product or service who possess the deep knowledge and expertise in the area to provide the project manager with real detailed requirements.

Other groups of stakeholders can include company management, subject matter experts and consultants, project team members, inspectors, legal experts, public opinion, government and last, but definitely not the least, adjacent departments within the organization.

Why Do We Neglect The Stakeholders?

I have seen it happen on numerous occasions - project manager and his team of technical experts deciding either independently or after being pressured by their management to ignore one or more stakeholder groups in their detailed scope definition efforts.

Exhibit 1



Some of the excuses mentioned by these teams were:

- *"We are afraid the stakeholder will find out too much about the problems we are having and the mistakes we made, rather than seek their help in overcoming the difficulties"*
- *"We are too busy to take time out to communicate and coordinate with our stakeholders"*
- *"We think/pretend it is harder when we involve the stakeholder"*
- *"We think we can do it without the stakeholder"*
- *"We believe stakeholder involvement costs too much money and time"*
- *"There are personality conflicts with key stakeholders"*
- *"Our own management won't let us"*
- *"We are already late with some of our deliverables! Talking to the stakeholder will waste more of a valuable project time"*

Note: in this particular discussion stakeholders include clients, customers, end-users, company management, subject matter experts, etc.

Project managers should be aware of these “excuses” to ignore the voice of the stakeholder and must be ready to defend their decisions to invest the necessary time and effort into building proper project scope. The value of investment in requirements can be clearly demonstrated by the study conducted by Barry Boehm to determine the relative costs of fixing an error at various stages of the project (see Exhibit 1). (For a detailed discussion of the “cost of mistake” concept see “Kick-Starting Your Projects – What Should You Know About Defining Scope”) The question I have been consistently asking my overly hasty team members, managers and other project stakeholders”

“Would you like to spend one hour discussing the scope with me now or would you rather spend between forty to one thousand man-hours fixing our scope omissions and defects closer to the end of the project?”

Types of Requirements

One of the popular ways of grouping requirements is by dividing them in *conscious*, *unconscious* and *undreamed-of* requirements (see Exhibit 2). Conscious requirements are typically the easiest to trawl for. These scope items are typically uppermost in customers’ minds, they are almost always indicative of something your customer is trying to create or improve. For example, a small business owner who is trying to increase her revenue stream by selling products on the Internet will undoubtedly mention a website with product catalogue, shopping baskets and ability to make payments using most popular credit cards. Therefore, it is very likely that these requirements would be the first ones mentioned by the stakeholders during one-on-one interviews with the project manager making such requirements fairly easy to catch.

On the other hand, unconscious requirements are a bit more difficult to extract because they are so common and familiar to the stakeholder that he frequently fails to mention them, assuming that everyone is aware them. Again, using to the small business owner example, including value-added tax into the final price of the product sold via Internet could be considered a common knowledge by the business owner, and yet a web developer may not possess sufficient understanding of accounting to add this requirement to the scope of the project. Unconscious requirements are one of the most difficult to elicit since they are frequently go unmentioned by the stakeholders. Only inquisitive and thoughtful questioning and follow up walkthroughs can unearth all of the unconscious scope items.

Undreamed-of scope items are the things that could be very useful to the customer, but for whatever reason – typically lack of technical expertise – she is not aware of their existence. Once more, retuning to the website example – the small business owner may not be aware that all credit card-related information must be encrypted and protected according to the industry standards. The burden of informing him about this regulation and thus adding an extra requirement lies on project manager’s shoulders.

Exhibit 2

Requirement Type	Explanation
Conscious Requirements	Uppermost in customers’ minds indicative of something your customer is trying to create or improve
Unconscious Requirements	So common and familiar to the stakeholder that

	he frequently fails to mention them
Undreamed-of Requirements	Things that could be very useful to the customer, but for whatever reason – typically lack of technical expertise – she is not aware of their existence

How To Ask Right Questions?

When developing a detailed scope document it is very important to ask the right questions to elicit all of the requirements from the customers. The problems of scope definition and documentation could be rooted in *unspecified information*, *unspecified comparison*, *generalization* or *universal qualifier* types of statements.

Unspecified Information - very frequently these statements are somewhat counterintuitive to the average person, since we tend to ignore or just blindly accept the underlying assumptions buried in them. For example a simple statement “We get the sales reports” should right away raise at least the following questions from an experienced project manager (see Exhibit 4 for more examples):

- Who “we”
- What do you mean by “get”?
- What is a “sales report”?

Unspecified Comparison – when the stakeholder uses words like “better”, “faster”, etc. The questions that should right away start popping up in the project manager’s head once he hears a statement like “The new container terminal should be better” are:

- “Better” compared to what?
- “Better” in which way?
- Who decided that it should be better?

Generalization – typically occurs when you hear words like “must” or “can’t”. For instance, the customer may state something to the effect of, “The new store must be located at the intersection of two major streets in the Oakmount neighbourhood”. The questions asked by the project manager should be:

- Why must you do it specifically that way? Why should the store be located at the intersection? Why major streets? If you are looking for high-traffic areas, have you considered other locations, like malls or shopping plazas?
- What happens if you can’t find a space for rent at the intersection of two major roads? Will you abandon the idea of opening the new store? Will you postpone it? What kind of drop in revenue are you anticipating?

Exhibit 3

Type of Statement	Sample Statement	Sample Questions
Unspecified Information	“We get the sales reports”	“Is it just you or your entire department?” “Or just specific employees within your

		<p>team?"</p> <p>"Are other departments privy to these reports?"</p> <p>"If yes, then who in other departments gets them?"</p> <p>"What do you mean by "get"?"</p> <p>"Are the reports printed out and mailed to you or are they faxed?"</p> <p>"Do you receive them by e-mail?"</p> <p>"In what format?"</p> <p>"Do you have a software package that produces them automatically?"</p> <p>What is a sales report?</p> <p>What kind of information is contained there?</p> <p>Do you just have one type of report or several?</p>
Unspecified Comparison	"The new container terminal should be better"	<p>"Better compared to what?"</p> <p>"To the existing terminals at your port or your competitors?"</p> <p>"Or the best in the world?"</p> <p>"Better in which way?"</p> <p>"In terms of square footage, overall container capacity, logistics, usage of computer technologies, security?"</p> <p>"Who decided that it should be better?"</p> <p>"What is the reasoning behind this decision to increase capacity, improve logistics or security?"</p>
Generalization	"The new store must be located at the intersection of two major streets in the Oakmount neighbourhood"	<p>"Why must you do it specifically that way? Why should the store be located at the intersection?"</p> <p>"Why major streets?"</p> <p>"If you are looking for high-traffic areas, have you considered other locations, like malls or shopping plazas?"</p> <p>"What happens if you can't find a space for rent at the intersection of two major roads?"</p> <p>"Will you abandon the idea of opening the new store?"</p> <p>"Will you postpone it?"</p> <p>"What kind of drop in revenue are you anticipating?"</p>
Universal Qualifiers	"The terminal gate shall always be operated from the main control room"	<p>"Is it really never or does it happen sometimes?"</p> <p>"Is it really always or are there some exceptions?"</p>

		<p>“What happens if the main control room is blocked?”</p> <p>“What should the employees do in case of emergency (e.g. fire)?”</p>
--	--	--

Universal Qualifiers – you come across them in statements containing words like “never” or “always”. The questions one should be asking in this case are:

- Is it really never or does it happen sometimes?
- Is it really always or are there some exceptions?

Which Technique Is The Best?

There has been some confusion with respect to which scope definition techniques should be used on various types of projects. In IT and software development we more or less know that functional and non-functional requirements and full-scale use cases typically fall into the category of larger, traditional waterfall projects, while user stories tend to get utilized on more agile, smaller engagements. Despite that I have been questioned by a large number of my colleagues and students from non-IT industries regarding the validity of various requirements trawling techniques on different kinds of projects.

In order to answer these questions, it is useful to divide the projects into three broad categories:

- *Rabbit projects* are small and fairly simple engagements with a collocated, tightly knit team and a small number of customers who are willing to invest a lot of their time in the scope definition stage. The techniques that work well on such projects are *brainstorming* and *creativity workshops* because of the proximity of the participants and their willingness to participate in scope definition process.

Exhibit 4

Technique	Description	Rabbit	Horse	Elephant
Apprenticing	Spend time working with an expert	★	★★	★★★★
Interviewing	One-on-one discussion with a user or expert	★★★★	★★★★	★★★★
Essence	Finds the real problem first	★★★★	★★★★	★★★★
Brainstorming	Facilitates creativity and invention	★★★★	★★	★★
Creativity Workshops	Unites the team to discover innovative requirements	★★★★	★★★★	★★

Note: Three stars implies great fit, two stars - adequate fit, one star - bad fit

- *Elephant projects* are as a rule, very large and sophisticated endeavours with budgets in tens or hundreds of millions of dollars, like launch of a space shuttle, development of a new type of passenger airplane or construction of a new port terminal. Project teams on such projects are large and very likely to be spread out at several locations - sometimes on different continents separated by thousands of miles. There are a lot of stakeholders who more often than not do not have much time at their disposal to invest in requirements trawling. As a result *apprenticing*, i.e. spending time watching the stakeholders perform their daily duties, especially in order to understand business processes can be of great value.
- *Horse projects* are the “something-in-between” endeavours that retain the characteristics of both “Rabbit” and “Elephant” projects. They could be, for example, of medium size but have a distributed group of stakeholders. My personal observations actually confirm that a vast percentage of projects will most likely fall into the “Horse” category since it is very difficult to find a project that possesses all of the characteristics of the “Rabbit” or “Elephant”. As a result of such blend, most of the techniques mentioned in the table rate as “great” or “satisfactory” fits for the “Horse” projects.

The interesting aspect of this table is that apparently techniques like *interviewing* and *searching for the essence* (i.e. trying to identify the real problem behind every scope item) are absolutely universal no matter what project you are working on.

What Is The Criteria For Quality Scope Items?

So, now that your scope document is complete what guidelines should you follow to ensure that requirements recorded are of good quality and can be validated by the project stakeholders and at the same time easily understood by all the technical team members – programmer, architects, engineers, etc. Below you will find a list of attributes of good project requirements (see Exhibit 6):

Exhibit 5

Attribute	Explanation
Complete	Has all the details of every scope item been covered? (e.g. all possible scenarios, alternatives and exceptions)
Consistent	Can the requirement be met without conflicting with other requirements? If not, the requirement should be removed or revised
Traceable	Is the origin of the detailed scope item known? Can it be traced directly to the high-level product or service feature recorded in the project charter?
Concise	Is the requirement stated simply and clearly? Can it be easily understood by both the customers and technical team members?
Design Free	The detailed scope item should be stating what must be done without indicating how. Did we make sure that the

	technical aspects of the detailed product design are left to the technical experts?
Standard Constructs	Requirements are stated as imperative needs using “shall”. Avoid using words “should” or “may”; they create a sense of false priorities.
Unique Identifier	Has the detailed scope item been uniquely identified? For example, “R1”, “R2”, “R3”, etc.
Prioritized	Has the requirement been assigned a priority relative to other requirements in the document? For example “Must Have”, “Should Have” and “Nice To Have”
Rationale for each requirement	Has the rationale for each scope item been clearly identified and agreed upon by all the relevant stakeholders? In other words, “do we really need this?”

Introducing Measurability

Let me start this section of the article with the following example. Pretend that you are a custom furniture builder and a client shows up at your workshop citing that he needs a dinner table “of an adequate size for his family”. Without any further discussions and requirements elicitation how easy do you think it would be to fulfill this order? Compare this requirement with the following one, “The table should be 6 feet long, 4 feet wide and 3 feet high”. Do you think this request is easier to implement?



If the above exercise was a bit too easy, let’s complicate things. How about this scope item:

“The new container terminal shall be of a sufficient square footage to accommodate load requirements in peak times”

I can’t speak for the reader’s experiences, but I have come across statements similar to the one just mentioned on numerous occasions while working as a process improvement consultant and peer reviewer. Interestingly enough, for some reason most of the people I have questioned find statement about the terminal much more “digestible” than the statement about the dinner table. I am not sure why this happens (this could probably be explained much better by psychologists, than by a project management expert) but it appears that the concept of dinner table is somewhat more tangible and familiar to the average person than an enormous area designed to house the containers. Maybe we tend to be more fastidious when it comes to familiar, palpable objects and concepts?

Now, compare the old terminal requirement to the following statement:

“The new container terminal shall have a capacity of at least 600,000 TEUs (twenty-foot equivalent units)”

If you were an architect or an engineer, which requirement do you think would have been more helpful if you were about to start working on a terminal blueprint?

Introducing measurability to scope items is one of the key factors in improving the quality of requirements and avoiding future misunderstandings (i.e. rework and budget overruns) as the project progresses from the planning to the execution and closeout stages.

Having said that, imposing measurability on scope is one of the most stressful exercises the project manager can go through during the scoping stage. Firstly one has to discover all of the words and phrases in the scope definition that need to be made measurable – this, by the way, is typically the easier of two steps. Secondly, the project manager has to sit down with the stakeholders and compel them to provide real numbers to replace vague wordings in the scope statements. Customers are usually reluctant to do that fairly early in the project because they either don't know the measures required and/or are unwilling to accept the responsibility for providing them. In other words, their line of thought can go something like this, “what happens if I claim 600,000 TEUs and it turns out that in the peak times the demand reaches 850,000 TEUs?”

Also, providing answers to such questions typically require a lot of additional work and research which in my experience has sometimes been discouraged by senior management as “wasting time” or “delaying the start of an important project”.

How does one impose measurability on scope items? One of the easiest ways of doing that is by asking the following question:

“What would you consider to be a failure to meet this requirement?”

Let's examine this method on an example. This conversation took place between me and a project manager working for a very large retail chain. He was responsible for one of the multiple “New Store Opening” projects and I was assigned to peer review his project documentation.

Me: *“You need to impose some kind of measurability on your deliverables”*

PM: *“Our deliverables can not have measurability associated with them. It is all-or-nothing approach. For example, if the store is supposed to have thirty point-of-sale (POS) stations, it should have all thirty stations installed and working for the opening!”*

Me: *“OK, but what happens if twenty nine out of thirty stations are working and one is defective? Do you think the company will say “no” to hundreds of thousands of dollars in revenue and postpone the opening of the store by even one day?”*

PM: *“Well in this case, obviously we will still go ahead with the opening”*

Me: *“And if twenty eight out of thirty stations are working?”*

PM: *“We would still go ahead!”*

Me: *“And what happens if only one out of thirty POS stations is working and the remaining twenty nine are broken?”*

PM: *“Well, of course we will have to postpone the opening of the store then!”*

Me: “Here is what you need to do; talk to the customer and try to determine the threshold for the number of non-working POS stations. That number is your measurability parameter!”

How Do You Know When To Stop?



The caveats of not spending enough time defining the detailed scope of the project have been discussed in project management literature a lot and have also been confirmed by numerous studies by various researches including Standish Group (Chaos Report), Project Management Institute (PMI) and others. However a question diametrically opposed to the issue above is, “how does one know when he is done with building the scope definition?” Below you will find some of the indicators that it is time to wrap up the requirements stage of the project and move on to the final stages of the planning phase:

- The stakeholders can't think of any other requirements – you notice that the stakeholders can't think of any scope items no matter how many different types of questions you are asking
- The stakeholders repeat issues that have already been covered – you discover that the conversation starts going in circles and the same requirements get mentioned again and again
- The stakeholders request the requirements that are all out of scope – Project Charter should

have listed all of the high-level features; if the scope was limited to just a three-bedroom home, the requests for pool and landscaped yard should be out of scope.

- Newly proposed requirements are all low priority – the stakeholders keep mentioning scope items to include in the project but agree that these are low-priority items
- The users are proposing features that should be included in one of the next phases – you notice that the conversations with the stakeholders start drifting towards the future phases rather than current “release” of the project

Article Summary

When selecting the sources of project scope make sure to consider all relevant people, especially your clients, customers and future users of the product. Having said that, it is imperative to, at least, inquire if other stakeholder groups, like your company management, subject matter experts, project team members, company lawyers, and the “adjacent” departments within the organization have anything to say about the project scope.

There could be some pressure from your clients, your project team or even from you managers to “speed things up” and “avoid wasting time” by forgoing conversations with some of the stakeholders. Avoid such shortcuts at all costs! Always remember that one-dollar mistake or omission made during the requirements stage can become a thousand dollar defect at the end of the project.

Keep in mind the types of requirements that exist and make an extra effort to capture the “unconscious” requirements, since they frequently go unmentioned by the customer.

Employ the entire array of various kinds of questions at your disposal. It will take some time to know which question should be asked at what time, but eventually it will become your second nature. Also, don’t be intimidated to ask “politically incorrect” questions like “Why do you think this way is better than the others?” or “When you say ‘never’ do you really mean ‘never’, or are there some exceptions?”

Know what requirements trawling techniques are appropriate for the type of the project you are working on. Also keep in mind that one can never go wrong with one-one interviews and trying to find the problem first rather than blindly accept the solution imposed by the customer - the “search for the essence” technique.

When documenting and especially when reviewing your scope documents, remember the quality attributes of good requirements and the concept of measurability.

Bibliography

1. “Software Requirements, Second Edition (Pro-Best Practices)” by Karl E. Wiegers
2. “Effective Requirements Practices” (Addison-Wesley Information Technology Series) by Ralph R. Young
3. “Exploring Requirements: Quality Before Design” by Donald C. Gause and Gerald M. Weinberg
4. “Requirements by Collaboration: Workshops for Defining Needs” by Ellen Gottesdiener (Kindle Edition - Feb 17, 2009)
5. Mastering the Requirements Process (2nd Edition) (Hardcover) by Suzanne Robertson and James C. Robertson
6. “Project Management Body of Knowledge (PMBOK)” by Project Management Institute (PMI)
7. Carpenter, Christine (2002) [1997]. The Wars of the Roses: Politics and the Constitution in England, c. 1437–1509. Cambridge Medieval Textbooks. New York: Cambridge University Press. ISBN 0-52131-874-2
8. Hicks, Michael (1995). "The Sources". in Pollard, Anthony. The Wars of the Roses. Problems in Focus. London: MacMillan Press. ISBN 0-333-60166-1
9. Ross, Charles (1999) [1981]. Richard III. Yale English Monarchs. New Haven, Connecticut; and London: Yale University Press. ISBN 0-30007-979-6

About Thinktank Consulting

Founded in 2003 by Jamal Moustafaev Thinktank Consulting is an industry leader in providing management consulting services to North American companies and government organizations. Our portfolio of services includes:

- Project Management and Requirements Engineering (Business Analysis) consulting and outsourcing
- Process improvement programmes & audits including:
 - Business Process Re-engineering
 - Requirements Engineering
 - Estimation
 - Project Management
 - Portfolio Management
- Training
 - *“Practical Portfolio Management - Selecting & Managing The Right Projects”*

- *“Successful Hands-On Management of IT and Software Projects”*
- *“Successful Hands-On Management of Modern-Day Projects”*
- *“From Waterfall to Agile - Practical Requirements Engineering”*

For further information, please contact:

Jamal Moustafaev BBA, MBA, PMP
President and CEO
Thinktank Consulting Inc.
Phone – 778-995-4396
E-mail – info@thinktankconsulting.ca
Website – www.thinktankconsulting.ca