

How the PMBOK Addresses Critical Success Factors for Software Projects: A Multi-round Delphi Study

Mohd Hairul Nizam Md Nasir^{1*}, Shamsul Sahibuddin², Rodina Ahmad¹, Shukor Sanim Mohd Fauzi³

¹ Faculty of Computer Science & Information Technology, University of Malaya, 50603 Kuala Lumpur, Malaysia

² Advanced Informatics Schools, Universiti Teknologi Malaysia, 54100 Kuala Lumpur, Malaysia.

³ Faculty of Computer & Mathematical Sciences, Universiti Teknologi Mara, Perlis Campus, 02600 Arau, Perlis, Malaysia.

* Corresponding author. Tel: +603 79676435; email: hairulnizam@um.edu.my

Manuscript submitted March 8, 2014; accepted July 8, 2015.

doi: 10.17706/jsw.10.11.1283-1300

Abstract: A four-round Delphi was conducted to determine the degree to which the PMBOK can address the identified critical success factors for software projects. Our findings show that the PMBOK provides a very effective framework for addressing only one critical success factor. Thirteen factors are addressed to a very good degree, five to a good degree, four to a fair degree and two are addressed to a limited degree. The experts highlighted that the PMBOK provides little operational guidance in its processes, and some customisations are required to manage software projects.

Key words: Delphi study, PMBOK, software project management, success factors.

1. Introduction

Project management has been a long-standing approach promoted by various bodies, intended to assist organisations, especially project managers, to manage project activities by applying a set of knowledge, skills, tools and techniques to meet user and stakeholder needs [1]. By using effective project management practices, it may assist organisations to better plan, manage, execute and control projects, thus resulting in better performance and productivity and contributing the success of software projects. Although it has been widely acknowledged that good project management cannot guarantee project success, poor project management usually results in project failure.

Various project management frameworks and methods have been developed covering all aspects of managing projects with the purpose of increasing the project success rate. The most prominent and internationally recognised project management framework is the Project Management Body of Knowledge (PMBOK) [1], which is now the de facto standard for project management, with widespread use in many non-IT and IT projects. Several published studies have reported a positive impact on the project outcome. To highlight a few of these, Phin *et al.* [2] reported that the PMBOK greatly assisted and guided them in planning and structuring the project as well as in defining the roles and responsibility of the stakeholders. Another large-scale project called Resistance Temperature Detectors Bypass Elimination (RTDBE), by ComEd, showed that by applying sets of PMBOK management principles, the RTDBE project was completed on time and within budget, without an adverse impact on the outage [3].

However, due to the nature of software itself which has been highlighted in [4], we argue that the PMBOK is not a perfect model to ensure the success of a software project. Our justification is that the PMBOK has

been claimed as being generic and descriptive [5]-[10], suitable for non-IT and IT projects, thus causing incorrect activities and processes to be applied to managing software projects. In addition, PMBOK suggests exhaustive list of good practices, in the form of tools and techniques that can be tailored and customized to specific needs. However, it can be confusing to choose which of these tools and techniques are suitable for use in managing software projects, especially for less experienced project managers. Thus, at the outset, we predicted that the PMBOK would contribute to a certain degree to addressing the identified critical success factors.

In our previous research, we have identified and determined to what degree does the Team Software Process (TSP) addresses the critical success factors for software projects [11]. Meanwhile, in this article we would like to determine to what degree the PMBOK addresses the critical success factors for software projects. Two research questions have motivated the investigation reported here:

- 1) Research Question 1 (RQ1): To what degree, as agreed upon by the experts, can the PMBOK address critical factors, given a particular set of critical success factors for software projects?
- 2) Research Question 2 (RQ2): What processes and/or knowledge areas, as outlined in the PMBOK, address each of the success factors, given a particular set of critical success factors for a software project?

The remainder of the paper is organised as follows. We first present a brief overview of the PMBOK in Section 2. Then, we present a rank-order of critical success factors for software projects in Section 3. In Section 4, our findings are presented and discussed. Section 5 presents our research limitations. Conclusions and suggested future research are highlighted in Section 6.

2. PMBOK

The PMBOK was launched in 1983 by the Project Management Institute (PMI), which is a global organisation that promotes the profession of project management, serving 408,465 members (as of January, 2013). PMI offers a certificate examination by which one can become a certified Project Management Professional (PMP). In the last ten years, the number of PMP credential holders has grown more than 1,300%, despite the current global economic recession [12], [13]. Project managers with the PMP credential indicate that they can direct and manage a project by following the processes and knowledge areas embodied in the PMBOK. All of these statistics reflect the increasing acceptance of the project management framework and standard, which are believed to have a substantial impact on project success.

The PMBOK is a collection of processes and knowledge areas that are generally accepted as the best practices within the discipline of project management. The PMBOK structures project management processes into five basic groups known as process groups: initiating, planning, executing, monitoring and controlling, and closing. These five basic process groups are executed in each project phase of the project lifecycle, and the application of these process groups is iterative, in such a way that the closing of a phase enables the initiation of the following one. Although the process groups and project phases are illustrated discretely, in an actual project there will be many overlaps and interactions that cannot be completely described here. Within each process group, the individual processes can be described by their inputs (e.g., requirement definition, software requirement specification), the tools and techniques that can be applied (e.g., cost estimation techniques, work-breakdown structure, object-oriented techniques) and the resulting outputs (e.g., development plan, quality plan, complete functional coding), all of which are known as software artefacts in a software project. The PMBOK relies heavily on processes and methods and places a strong focus on balancing scope, quality, time, cost, resources and risk for project success. These process groups also comprise 42 management processes that describe activities throughout a project's lifecycle. These management processes include the tools and techniques used for applying knowledge and skills

described in the nine knowledge areas, namely project integration management, project scope management, project time Management, project cost management, project quality management, project human resource management, project communication management, project risk management and project procurement management.

The PMBOK is the standard for managing projects across various industries, including software projects, and is mostly used in US-based organisations. The value of the PMP certification is demonstrated by its formal acceptance by several well-known organisations such as Accenture, IBM Corporation, Infosys Technologies Limited, Siemens, the National Aeronautics and Space Administration (NASA), Oracle and many others as PMI Global Corporate Council Members. These council members are aligned with PMI and committed to using the PMBOK for achieving a competitive advantage and a more effective overall organisation.

Since 1983, the PMBOK has been extensively revised and expanded, and in 1996, the first edition of “A Guide to the Project Management Body of Knowledge,” known as the PMBOK Guide, was released. So far, over three million copies of the PMBOK Guide have been circulated. The primary reference for PMBOK information, processes and knowledge in this study was “A Guide to the Project Management Body of Knowledge Fourth Edition,” the PMBOK Guide [1], which was released in 2008.

3. Critical Success Factors

In order to determine how PMBOK address the success factors for software projects, the set of success factors must first be determined. Critical success factors are factors that, if addressed appropriately, will significantly improve the chances of project success [14]. Therefore, to make success more likely, processes and activities should be chosen to address critical success factors. Although numerous research studies have been performed in the area of project management to identify factors that influence the success and/or failure of projects [15]-[18] these factors have usually been identified for projects in a range of industries, such as engineering, manufacturing, construction and training, rather than being focused on software development or IT projects. Dvir *et al.* [19] suggested that project success factors are not universal to all projects. Thus, the success factors identified in other industries cannot be used as valid factors for software projects. Therefore, in this study it was decided to used the rank order of success factors for software projects developed by Nasir and Sahibuddin [4]. The justification of selecting and using this set of success factor is that these factors were derived from an extensive literature review of established scientific research journals, well-known survey reports and articles written by experts and practitioners with wide experience in software-related industries between 1990 and 2010. Their most important characteristic is that these success factors are only applicable to software projects. The 26 success factors that were found to be related to project success are tabulated in Table 1.

This rank order of success factors for software projects served as a baseline to conduct a multi-round Delphi study with experts who have years of experience in software industries and an in-depth knowledge of TSP and PMBOK. The advantage of this approach is that a collective analysis of how TSP and PMBOK address these success factors by a group of experts is less likely to be influenced by biases such as limited knowledge and experience, which may be the case if a single individual performs the analysis.

4. How the PMBOK Addresses the Critical Success Factor for Software Projects

The Delphi method allowed us to capitalise on the varied experience and in-depth knowledge of the experts in providing complete knowledge about phenomena [20] through controlled feedback. We decided on the Delphi method for two reasons. First, prior research has not yielded a set of validated measures of the construct of interest, i.e., how the PMBOK addresses the critical success factors for software projects.

Second, we chose it because of its ability to achieve consensus, something that was lacking in field interview and case study methods. The Delphi method provided a good solution that allowed us to conduct our investigation with rigor and internal consistency.

Table 1. Critical Success Factors Identified in [22]

Rank No	Critical Success Factors
1	Clear requirements and specifications
2	Clear objectives and goals
3	Realistic schedule
4	Effective project management skills/methodologies (project manager)
5	Support from top management
6	User/client involvement
7	Effective communication and feedback
8	Realistic budget
9	Skilled and sufficient staff
10	Frozen requirement
11	Familiar with technology/development methodology
12	Proper planning
13	Appropriate development processes/methodologies (process)
14	Up-to-date progress reporting
15	Effective monitoring and control
16	Adequate resources
17	Good leadership
18	Risk management
19	Complexity, project size, duration, and number of organisations involved
20	Effective change and configuration management
21	Supporting tools and good infrastructure
22	Committed and motivated team
23	Good quality management
24	Clear assignment of roles and responsibilities
25	Good performance by vendors/contractors/consultants
26	End-user training provision

4.1. The Expert Profiles

To ensure the reliability of the experts' opinions, the following criteria were established and used to select the experts: 1) the expert must have at least 20 years of experience in the area of software industry, 2) the expert must have at least 15 years of experience in software project management, 3) the expert must possess PMI Project Management Professional certification and 4) the expert must have at least 20 publications related to software project management. The first two criteria ensure that the expert has a varied experience background, and the last two criteria ensure expertise and familiarity with the PMBOK and software project management. We also excluded experts with experience in non-software projects because our focus was on software project management. Other important criteria that we took into account were: 1) capacity and willingness of the experts to participate and 2) effort and time commitment to participate in a multi-round Delphi study [21].

We invited nine experts to participate in this research study. Three experts responded and stated their willingness and commitment to participate. The small sample size is due to limited expertise in our country and also to the difficulty of finding experts who could fulfil our criteria, especially that of effort and time commitment. However, we have high confidence in the quality of our experts. The profiles of the three experts, as shown in Table 2, indicate that all of the experts had impressive experience in the area of software project management and were well-qualified. According to Hakim [22], small samples can be used to develop and test explanations, particularly in the early stages of the work. For example, Lam et al. [23] used three experts to develop rules for a ceramic casting process, Nambisan *et al.* [24] recruited six experts to develop taxonomy of organisational mechanisms and Gustafson *et al.* [25] used four experts to estimate almanac events in their investigation of Delphi accuracy. We argue that the number of experts does not have much impact on the outcome of our research; however, the true experts in this field provide great insight in analysing, extracting and discussing all the features that are outlined in the PMBOK and map back to the identified success factors. Thus, we decided to utilise three experts in our research study. We believe that the involvement of experts of such high reputation and calibre gives weight and rigor to our results.

Table 2. Experts Profiles

Expert	Experience in software industries	Experience in software project management	PMP Certified	Software project management-related publications
Expert 1	21 years	15 years	Yes	23
Expert 2	32 years	28 years	Yes	24
Expert 3	26 years	23 years	Yes	22

4.2. Data Collection and Analysis Methods

A Delphi questionnaire was mailed to the experts to collect their input in this multi-round Delphi. We first requested the experts to review our generated list of critical success factors for software projects, as shown in Table 1. We provided a definition and description of each of the factors to ensure that they were all working from a common list of items with common definitions. The experts did not highlight any problems with the list of critical success factors given. We also asked the experts about their ability to respond to the questions, and we found that they felt qualified and able to respond to the questions. Prior to its mailing, the survey was pre-tested by five information science researchers for clarity and ease of understanding. No changes were found to be necessary.

In round 1, the experts were asked to rate how the PMBOK addressed the critical success factors for software projects and to provide descriptions to justify their rating. We also asked the experts to specifically state the PMBOK management processes and/or knowledge areas in their description so that every critical success factor was clearly addressed by the management processes and/or knowledge area.

A six-point classification scale was implemented as follows:

- 1) Best Practice-5 : The PMBOK provides a very effective framework for addressing the critical success factors and has a direct impact on the software project's success.
- 2) Very Good-4 : The PMBOK provides a very good framework for addressing the critical success factors and has a significant impact on the software project's success, but may not be the most effective way of doing things.
- 3) Good-3 : The PMBOK provides a good framework for addressing the critical success factors, but there are minor missing processes that may impact the software project's success.
- 4) Fair-2 : The PMBOK provides a framework that addresses the critical success factors to a reasonable degree, but there are several missing processes and/or incorrect settings of priorities that impact the software

project's success.

- 5) Weak-1 : The PMBOK provides a framework that address the critical success factors to a limited degree but does not cover everything that is required.
- 6) Not addressed-0 : The PMBOK does not provide any way to address the critical success factors.

Their responses were reviewed, consolidated and disseminated back anonymously to them in the subsequent round. In the following round, we asked the experts to confirm that their ratings and descriptions were consistent with their previous responses. To achieve consensus, experts were asked to revise, correct, add to and eventually validate their earlier input after reviewing the feedback and comments of the others experts. We measured the degree of consensus among the experts using Cohen's kappa coefficient (k) for each round between each pair of experts. This coefficient reflects the extent to which the observed consensus between experts is superior to that obtained by chance [26]. In our study, we iterated this multi-round process until we reached the kappa value of 0.7 with $p < 0.001$, indicating an acceptable level of consensus. The kappa value of 0.85 indicates almost perfect agreement [27], but a kappa of 0.7 or more is usually considered to be an acceptable level of agreement. We can be confident of the reliability of our output by confirming a high level of agreement. This approach is consistent with the basis that the number of rounds is somewhat flexible and the Delphi iteration process stops when a reasonable level of consensus is reached. We also used standard deviation to observe agreement among the experts for each of the critical success factors throughout the round. A low standard deviation indicates that the ratings tend to be very close with each other, whereas a high standard deviation indicates that the ratings are spread out over a large range of values. During the final round, we presented the final findings to the experts and asked them to review and finalize their ratings and descriptions. All the experts were agreed with the final findings, and no changes were found necessary.

4.3. Findings

A four-round Delphi process was used to achieve consensus among the experts as well as to finalise the findings. Table 3 reports the kappa value for each Delphi round. A kappa value from 0.40 to 0.59 is considered moderate, 0.60 to 0.79 substantial, and 0.80 outstanding [27]. After we finalised our findings, the degree of consensus between expert 1 and expert 2 achieved outstanding level ($k = 0.944$) with $p < 0.001$, while the degrees of consensus between expert 1 and expert 3 as well as between expert 2 and expert 3 were at the outstanding level, i.e., 0.888 and 0.945, respectively, with $p < 0.001$. Even though we have reached kappa value of 0.7 with $p < 0.001$ in round 2, we decided to proceed with the next round since there are 2 results indicated high standard deviation.

Table 3. Kappa Values for Each of the Delphi Rounds

Experts	Round 1	Round 2	Round 3	Round 4
Expert 1 vs. Expert 2	$k=0.799$	$k=0.946$	$k=0.944$	$k=0.944$
Expert 1 vs. Expert 3	$k=0.632$	$k=0.890$	$k=0.888$	$k=0.944$
Expert 2 vs. Expert 3	$k=0.606$	$k=0.839$	$k=0.945$	$k=0.945$

Table 4 reports our findings for the final round for each of the critical success factors. In terms of agreement among the experts, the average standard deviation in round 1 was 0.672 with a total standard deviation of 17.465; this was reduced to 0.155 with a total of standard deviation of 4.041 in round 2. Round 3 provided better agreement among the experts, where the average standard deviation was further reduced to 0.111 with a total of standard deviation of only 2.885. These figures remained stable until the final round. The results show that the agreement among experts for each critical factor is quite high, namely, the standard deviation for each was equal to or less than 0.577 and standard deviation for over 80% of the

factors was 0.000. The results also suggest that 21 of the 26 critical factors gained an outright consensus from the experts. The remaining 5 critical factors showed slight differences with a standard deviation of 0.577.

Table 4. Experts' Ratings of How PMBOK Addresses Critical Success Factors: Final Round

No.	Critical Success Factors	Expert Ratings on PMBOK			Std. Deviation
		E1	E2	E3	
1	Clear requirements and specifications	4	4	4	0.000
2	Clear objectives and goals	3	3	2	0.577
3	Realistic schedule	3	3	3	0.000
4	Effective project management skills/methodologies (project manager)	4	4	4	0.000
5	Support from top management	2	2	2	0.000
6	User/client involvement	2	2	2	0.000
7	Effective communication and feedback	4	4	4	0.000
8	Realistic budget	4	3	4	0.577
9	Skilled and sufficient staff	4	4	4	0.000
10	Frozen requirement	3	3	3	0.000
11	Familiar with technology/development methodology	2	2	2	0.000
12	Proper planning	4	4	4	0.000
13	Appropriate development processes/methodologies (process)	2	2	2	0.000
14	Up-to-date progress reporting	4	4	4	0.000
15	Effective monitoring and control	4	4	4	0.000
16	Adequate resources	4	4	4	0.000
17	Good leadership	1	1	1	0.000
18	Risk management	4	4	4	0.000
19	Complexity, project size, duration, and number of organisations involved	4	4	3	0.577
20	Effective change and configuration management	5	5	4	0.577
21	Supporting tools and good infrastructure	1	1	1	0.000
22	Committed and motivated team	3	3	3	0.000
23	Good quality management	4	4	3	0.577
24	Clear assignment of roles and responsibilities	3	3	3	0.000
25	Good performance by vendors/contractors/consultants	4	4	4	0.000
26	End-user training provision	0	0	0	0.000

As shown in Table 5, it appeared that the experts agreed that the PMBOK provides a very effective framework for addressing only 1 (3.85%) critical success factor, i.e., 'effective change and configuration management'. It was also found that the experts agreed that the PMBOK provides a very good framework for addressing 13 critical success factors (50.00%) where the greatest concentration of critical success factors lies. Our findings suggested that 5 critical success factors (19.23%) are addressed by the PMBOK at a 'Good' level, 4 out of 26 critical success factors (15.38%) at a 'Fair' level, and only 2 critical success factors (7.69%) are addressed to a limited degree. Only 1 critical success factor (3.85%) is not addressed by the PMBOK, i.e., 'end-user training provision'. Tables IV and V answered our second formulated research question (RQ2).

Table 5. Degree to Which PMBOK Addresses the Critical Success Factors in Software Projects

Classification Degree/Level	Critical Success Factors
Best Practice-5 : The PMBOK provides a very effective framework for addressing the critical factor and has a direct impact on the software project's success	<ul style="list-style-type: none"> • Effective change and configuration management
Very Good-4 : The PMBOK provides a very good framework for addressing the critical factor and has a significant impact on the software project's success, but it may not be the most effective way of doing things	<ul style="list-style-type: none"> • Clear requirements and specifications • Effective project management skills/methodologies (project manager) • Effective communication and feedback • Realistic budget • Skilled and sufficient staff • Proper planning • Up-to-date progress reporting • Effective monitoring and control • Adequate resources • Risk management • Complexity, project size, duration, and number of organisations involved • Good quality management • Good performance by vendors/contractors/consultants
Good-3 : The PMBOK provides a good framework for addressing the critical factor, but there are minor missing activities that may impact the software project's success	<ul style="list-style-type: none"> • Clear objectives and goals • Realistic schedule • Frozen requirement • Committed and motivated team • Clear assignment of roles and responsibilities
Fair-2 : The PMBOK provides a framework that addresses the critical factor to a reasonable degree, but there are several missing activities and/or incorrect settings of priorities that impact the software project's success	<ul style="list-style-type: none"> • Support from top management • User/client involvement • Familiar with technology/development methodology • Appropriate development processes/methodologies (process)
Weak-1: The PMBOK provides a framework that addresses the critical factor to a limited degree but does not cover everything that is required	<ul style="list-style-type: none"> • Good leadership • Supporting tools & good infrastructure
Not addressed-0 : The PMBOK does not provide any way to address the critical factor	<ul style="list-style-type: none"> • End-user training provision

Table 6. Summary of Experts Opinions on How the PMBOK Addresses the Critical Success Factors

Critical Success Factor	Experts' Opinions	Coverage Level	PMBOK Processes/ Knowledge Areas
1. Clear requirements and specifications	PMBOK 2008 has been dramatically enhanced to take full advantage of requirement-gathering techniques. Eight tools and techniques are suggested by the PMBOK for requirement gathering, but it has nothing to say about how to define requirements and produce clear requirements and specifications.	Very Good (4)	5.1 Collect Requirements 5.2 Define Scope
2. Clear objectives and goals	Project goals and constraints are provided to the project manager by sponsoring management without involving team members; thus, no clear communication method is defined for team members to know and understand the objectives and goals. Although the PMBOK makes frequent reference to aligning processes with the project objectives, it gives little guidance and provides few mechanisms for how to achieve clear objectives and goals.	Good (3)	Both processes that fall under the Initiating Process Group

3. Realistic schedule	<p>PMBOK uses formal processes for estimating and scheduling that are capable of producing a realistic schedule. However, nothing specific is included in the process to address political pressure to produce shorter schedules during private negotiations between the project manager and sponsoring management. This can be a much bigger problem than the lack of capability of the estimating process. Furthermore, there is an absence of operational guidance about how to develop and maintain realistic schedules.</p>	Good (3)	<p>6.1 Define Activities 6.2 Sequence Activities 6.3 Estimate Activity Resources 6.4 Estimate Activity Duration 6.5 Develop Schedule</p>
4. Effective project management skills/ methodologies (project manager)	<p>A project manager with a PMP credential guarantees that he or she is familiar with the best practices and effective processes for project management to the extent that has been defined in the PMBOK. Although operational guidance is seldom provided, the project manager role is well defined, and project management activities are covered from many perspectives. PMP certification is at best a necessary but not sufficient condition. It is no guarantee that the PMP-certified manager has the necessary skills or is aware of the required methods to manage a software project.</p>	Very Good (4)	Not Applicable
5. Support from top management	<p>The identification of top management is documented in the Stakeholder Register; and top management influence, interest and degree of involvement are documented in the Stakeholder Management Strategy plan. By actively managing the expectations from top management, the PMBOK may increase project success by influencing their expectations, addressing their concerns and resolving their issues. However, it does not really have specific mechanisms to ensure alignment and on-going support from top management throughout the project.</p>	Reasonable (2)	<p>10.1 Identify Stakeholders 10.4 Manage Stakeholder Expectations</p>
6. User/client involvement	<p>The identification of user and client is documented in the Stakeholder Register; and their influence, interest, and degree of involvement are documented in the Stakeholder Management Strategy plan. Customer involvement is expected during project initiation, requirement definition, risk identification, validation, phase exit reviews, scope management and project closeout activities. However, operational guidance for user/client involvement is not provided.</p>	Reasonable (2)	<p>4.6 Close Project or Phase 10.1 Identify Stakeholders 10.4 Manage Stakeholder Expectations</p>
7. Effective communication and feedback	<p>The PMBOK offers a rich set of processes to ensure effective communication to all of the identified stakeholders and recommends that communication activity be considered from several potential perspectives. The PMBOK calls for regular status meetings but provides very little guidance on content or making them effective, especially in the operational communication process for teams.</p>	Very Good (4)	<p>All 5 processes that fall under Project Communications Management Area</p>
8. Realistic budget	<p>Although an entire chapter in the PMBOK is dedicated to estimating and managing costs and budget, there is a lack of operational guidance for software project cost and budget estimation.</p>	Very Good (4)	<p>7.1 Estimate Cost 7.2 Determine Budget</p>

9. Skilled and sufficient staff	Skill and sufficiency of resources are addressed during the Develop Project Management Plan process. Also, the dedicated chapter entitled "Project Human Resource Management" deals with this factor, which covers technical competency, interpersonal skills, the need for training, resource availability and performance assessment. The PMBOK does not even mention taking organization attrition rate into consideration in its discussion of staffing. Furthermore, the issue on political pressure from sponsoring management to keep staff size down, particularly in contracting environments, is not addressed by the PMBOK.	Very Good (4)	4.2 Develop Project Management Plan All 4 processes that fall under Project Human Resource Management Area
10. Frozen requirement	Requirement change is unavoidable during project execution, and therefore the ideal state of a Frozen Requirement is essentially never achieved. However, the PMBOK does recommend baselining and then actively managing changes to requirements.	Good (3)	5.2 Define Scope 5.4 Verify Scope 5.5 Control Scope
11. Familiar with technology/development methodology	The PMBOK does not address software development methodology. It addresses familiarity with the technology in the Develop Human Resource Plan, where skills needed by the project are identified and then acquired.	Reasonable (2)	9.1 Develop Human Resource Plan
12. Proper planning	The processes outlined are comprehensive and cover the aspects of scope, time, cost, quality, communications, risks, human resource and procurement. All of these processes are managed by a project manager who produces a plan based on input from the team leader, and a copy of the approved plan is provided to team members. A key component is missing, i.e., members' involvement in producing the plan.	Very Good (4)	All 20 processes that fall under the Planning Process Group
13. Appropriate development processes/methodologies (process)	The PMBOK consists of a collection of management processes that are generally accepted as best practice within the discipline of project management. It includes phases common to most processes, such as planning and validation. However, there are unique differences in terms of processes and methodologies in software projects that require some customisation of PMBOK management processes.	Reasonable (2)	Not Applicable
14. Up-to-date progress reporting	Report Performance contains information such as current project status, accomplishments for the reporting period, upcoming activities, forecasts, issues and risks. It ensures that the collection and distribution of project performance reports are done in a timely manner. However, the frequency of status reports is left unspecified.	Very Good (4)	10.5 Report Performance
15. Effective monitoring and control	The Monitoring and Controlling process group comprehensively covers the aspects of scope, time, cost, quality, communications, risks and procurement. Yet, the PMBOK should include quantitative approaches to integrating defect management into Monitoring and Control	Very Good (4)	All 9 processes that fall under the Monitoring and Controlling Process Group

provides an opportunity to get ahead of problems by managing the cause rather than trying to correct the negative effect.

16. Adequate resources	Estimate Activity Resources determines the type and quantity of resources required. However, no operational guidance is provided. It is weak on commitment discipline, making it somewhat more likely that a manager would commit to performing a project with inadequate resources due to organisational political pressure.	Very Good (4)	6.3 Estimate Activity Resources
17. Good leadership	The PMBOK has a paragraph on leadership, distinguishing between leadership skills and management techniques. However, there is no guidance or training provided to the project manager or senior management on how to be an effective leader other than a reference. Compared with other project management qualities, leadership does not get much attention in the PMBOK.	Limited (1)	Not Applicable
18. Risk management	The PMBOK has a specific knowledge area on risk management that addresses risk identification, qualitative and quantitative risk analysis, planning risk responses, and monitoring and controlling risks. However, the PMBOK's processes do not address to a great deal the roles and responsibilities of team members relative to risk management.	Very Good (4)	All 6 processes that fall under the Project Risk Management Area
19. Complexity, project size, duration, and number of organisations involved	Most complex and large-scale software projects involve great numbers of contractors and vendors, and one of the most crucial issues is procurement and subcontracts. The PMBOK does often mention how the appropriate choice of phases, processes and techniques varies depending on the complexity and size of the project.	Very Good (4)	All 4 processes that fall under the Project Procurement Management Area
20. Effective change and configuration management	The Change Management Plan and Configuration Management Plan are produced during the development of the Project Management Plan. Meanwhile, during Perform Integrated Change Control, there is a review of all change requests, in which changes to project-related deliverables and documents are approved and managed. Every change request will be either approved or rejected by an authorised party such as the Change Control Board (CCB) or customer representatives. The PMBOK suggests that a configuration management system with integrated change control provides a standardised, effective and efficient way to manage changes.	Best Practise (5)	4.2 Develop Project Management Plan 4.5 Perform Integrated Change Control
21. Supporting tools & good infrastructure	The PMBOK points to consideration of Enterprise Environmental Factors (existing facilities and equipment) for every management process, even though they are not listed as inputs in the process specification. The Estimate Costs process also mentions consideration of infrastructure and material costs. However, it does not get much attention in the PMBOK.	Limited (1)	7.1 Estimate Cost

22. Committed and motivated team	Develop Project Team improves team interaction and coordination, hence increasing team productivity and project performance. However, the coverage is superficial at best, with almost no operational guidance provided.	Good (3)	9.3 Develop Project Team
23. Good quality management	The management processes cover many aspects of quality, including quality management and quality assurance (audit), but the PMBOK does not provide actual operational definitions.	Very Good (4)	All 3 processes that fall under the Project Quality Management Area
24. Clear assignment of roles and responsibilities	Develop Human Resource Plan identifies and documents project roles and responsibilities required to complete the project. Three tools and techniques are suggested by the PMBOK to produce the Human Resource Plan, but it discussed the identification of roles and responsibilities at a meta level as an activity.	Good (3)	9.1 Develop Human Resource Plan
25. Good performance by vendors/contractors /consultants	Selection of contractors is based on pre-established criteria before determining who is qualified to carry out the project. During the Administer Procurements process, the contractor's ability to meet the procurement and contractual requirements is ensured. The performances of contractors are reviewed, evaluated and reported so their work performances can be rated and measured. This will determine whether the contractors or vendors have the ability to continue to perform work on the current project or similar work on a future project. For large project, it should have a responsible engineer resident at the subcontractor's sight to handle technical coordination issues. This was not even mentioned as a consideration in PMBOK	Very Good (4)	All 4 processes that fall under the Project Procurement Management Area
26. End-user training provision	The PMBOK does not address this CSF to any degree.	Not Addressed (0)	Not Applicable

Table 6 summarises the opinions of the experts on how the PMBOK addresses the critical success factors for software projects. To answer RQ2, Table 6 also provides a mapping between PMBOK processes and/or knowledge areas and all the critical success factors. This mapping is useful because we can identify which PMBOK process and/or knowledge area addresses each of the critical success factors. As the PMBOK Guide itself does not identify the relative importance of each of the processes and knowledge areas, this mapping can help to identify the most important processes and knowledge areas that should be given more attention based on the rank-order of critical success factors. For example, the project manager should give higher priority to the 'Collect Requirements' and 'Define Scope' processes compared with the 'Develop Project Team' processes because both 'Collect Requirements' and 'Define Scope' processes address higher-ranked critical success factors, which have a higher impact on project success. An in-depth analysis should be conducted to determine the relative importance of the PMBOK processes and knowledge areas based on the rank-order of critical success factors.

The summary of expert opinions in Table 6 shows that, from an expert's perspective, the PMBOK provides little or no operational guidance in its processes that are aimed at addressing 12 of the critical success factors. These 12 critical success factors are clear objectives and goals, realistic schedule, effective project management skills/methodologies (project manager), support from top management, user/client

involvement, effective communication and feedback, realistic budget, adequate resources, good leadership, committed and motivated team, good quality management and clear assignment of roles and responsibilities. In the case of clear objectives and goals, for example, the PMBOK states that, as part of the initiating processes, clear descriptions of the project objectives are developed, and it includes the tasks of defining and refining the objectives as part of the planning process group. The PMBOK does mention measurable project objectives and related success criteria as an example of an item that could appear in the project charter. It mentions quality objectives as a part of project quality management and establishing goals as one of the tasks in a team building activity. While it makes frequent reference to aligning things with the project objectives, it gives little guidance and identifies few mechanisms for how to achieve clear objectives and goals.

Other than that, the experts also highlighted that some of the processes in the PMBOK need to be modified for use with a software project. Three critical success factors that relate to this issue, as identified by the experts, are clear requirements and specifications, familiarity with technology/development methodology and appropriate development processes methodologies (process). Although the processes cover many aspects of project management, there are unique differences in terms of processes and methodologies in software projects that require some customisation of PMBOK management processes. However, it must be noted that if the organisation needs to customise the PMBOK framework, this can take substantial resources, especially time, effort and money, which should be taken into consideration. All the issues highlighted reflect the finding that the management processes outlined in PMBOK are generic and descriptive, which is consistent with the claim in previous studies [5]-[10].

We are also interested in pointing out the expert views on the twelfth critical success factor, i.e., proper planning. PMBOK addresses this critical success factor in the planning process group, which is the only group that spans all nine knowledge areas, covering 20 well-defined management processes along with recommended tools and techniques. The processes outlined are comprehensive and cover the aspects of scope, time, costs, quality, communication, risk, resources and procurement. However, in normal practice, all these planning processes are handled and managed by the project manager, who produces a plan based on input from the team leader, and a copy of the approved plan is provided to team members. The experts highlighted the need for the team members' involvement in producing and maintaining a plan. The main reason is that the project managers do not know enough about the work to make detailed plans, and the management needs to trust the developers (team members) to produce plans [28].

An interesting point that is worth considering, as highlighted by one of the experts, is that a framework or a model should get wide acceptance by the customer community. If a framework or a model cannot gain traction with the user community, it cannot have much impact, regardless of how capable or perfect it is. This has always been the strength of the PMBOK.

5. Research Limitations

The critical success factors identified in this research study were extracted from multiple empirical data and expert views from eight countries (Finland, the United States, the United Kingdom, Hong Kong, Singapore, Belgium, Australia, and Canada) concerning small to large software projects in various domains; however, the findings are applicable only to software projects. Although a sample of eight countries is small and generalisability to the entire software engineering community worldwide is problematic, we have high confidence in our research findings because most of the articles included were taken from established scientific research journals and had a minimum of 11 citations, and a few of them were from well-known survey reports and journal articles written by experts and practitioners who had a wide range of experience in software-related industries from 1990 until 2010. Often, the factors reported in books are

based on the previous work of others and do not cover the latest research findings; therefore, we did not consider books in this study. We also decided not to include conference and workshop proceedings because it is hard to determine the quality of the articles in such publications. Note that this research study was not intended to localise the findings; thus, we considered it irrelevant to conduct an empirical study in any particular country.

As with any Delphi-type technique, this research was limited by the fact that it employed only three experts. While experts were chosen for their vast experience in managing software projects and in-depth knowledge of PMBOK, we can make no claim about the representativeness of our sample set. The experts were not randomly chosen, but their selection was based on the quality and reliability of the sources based on criteria that had been set. The profiles of the three experts, as shown in Table 2, indicate that all of the experts had impressive experience in the area of software project management and that they were well-qualified. With our careful design and execution of the Delphi study, we have high confidence in the quality of the experts and the opinions they contributed. Despite the aforementioned limitations, we believe that the results have both research and practical implications.

6. Conclusion

We have reported on our extensive literature survey of critical success factors that impact software projects. In this research study, 43 articles were found to make significant contributions that could be analysed to develop a list of critical factors that specifically affect the success of software projects. These 43 articles consist of 9 published sets of empirical data from case studies, 29 published empirical data sets from surveys and 5 articles written by experts and practitioners from 1990-2010. The method of content analysis was adopted in this study, rather than the data extraction method or the frequency analysis method alone, because some of the factors described by the authors in the articles were not explicitly clear and required careful reading, understanding and interpretation to produce accurate findings.

Based on this set of critical success factors, a four-round Delphi study was conducted to determine the degree to which PMBOK can address all the identified factors. Our results show that the experts agreed that PMBOK provides a very effective framework for addressing only 1 critical success factor out of 26 (3.85%), i.e., 'effective change and configuration management'. It was also found that the experts agreed that PMBOK provides a very good framework for addressing 13 critical success factors (50.00%) where the greatest concentration of critical success factors lies. Our findings suggest that 5 critical success factors (19.23%) are addressed by PMBOK at a 'Good' level, 4 critical success factors (15.38%) at a 'Fair' level, and only 2 critical success factors (7.69%) are addressed to a limited degree. Only 1 critical success factor (3.85%) is not addressed by PMBOK, i.e., 'end-user training provision'. From an expert's perspective, PMBOK provides little or no operational guidance in its processes for addressing 12 out of 26 critical success factors. The experts also highlighted that although PMBOK processes cover many aspects of project management, unique differences in processes and methodologies exist in software projects that require some customisation of PMBOK management processes. Our data also provide a useful mapping between PMBOK processes and/or knowledge areas and the critical success factors they address. Throughout the round, we noticed that Delphi study provided good commentary and discussion channel. Although many of the same issues emerge, it was clear that the experts often focused on different angle when discussing the same critical success factors.

This research focused on the PMBOK fourth edition, as it is the latest version released by PMI as of 2008 and is internationally recognised. Other frameworks and methodologies such as Project In Controlled Environment 2 (PRINCE2) and Agile Project Management could also have been examined. A similar approach could also be used in fields other than software project management, for instance, in the field of

software process improvement standards, such as the Capability Maturity Model Integration (CMM-I), International Organization for Standardization (ISO) 15504, International Organization for Standardization (ISO) 9000 or Team Software Process, or in well-known software development processes such as agile processes, Rational Unified Process (RUP), for determining how these frameworks and methods address all these critical factors.

Each framework or methodology on its own could not perfectly address all the identified critical success factors. By blending a project management framework and software process improvement, or other excellent software development processes, we believe one can address all the critical success factors to a better degree. Based on our initial research study, for example, it was found that PMBOK and Team Software Process (TSP) each contribute to a certain degree to addressing the identified critical success factors. To highlight 'realistic schedule' as an example, PMBOK uses formal processes for project estimating and scheduling that are capable of producing a realistic schedule when suitable historical data are available. However, nothing specific is included in the process to address political pressure to produce lower cost estimates and shorter schedules during the negotiations of scope, budget and delivery date that are private between the project manager and sponsoring management. This gap can be a much bigger problem than a lack of capability of the estimating process. TSP, in contrast, uses a highly capable estimation process based on historical data. The entire team produces and reviews the plan together, identifies any proposed exceptions to management goals and constraints, and participates in the out-brief to sponsoring management. This process helps to catch omissions and inconsistencies that could result in an unrealistic plan. Any negotiations concerning scope, budget, staffing, for example, are performed in public as part of the out-brief meeting. Team and management commitments are documented in the minutes of the meeting. This element is explicitly included to prevent the project manager from privately acquiescing to unrealistic goals or constraints under pressure from management or the client. We can integrate these two models to complement each other and to more effectively address the critical success factors for software projects. This approach is supported by several research efforts that seek integration between different areas, such as traditional project management and agile project management [29], CMMI for Development (CMMI-Dev) and PMBOK [30], PMBOK and Rational Unified Process [31], Agile and PRINCE [32], CMMI and PMBOK [33] and many more, to ensure better control in managing software projects.

It is our hope that the findings reported here will complement existing research in the area of software project management and will be investigated more deeply. The findings provide an indication as to what extent PMBOK addresses the critical success factors for software projects.

Acknowledgment

This research was supported financially by the University of Malaya Grant (no. RG316-14AFR). We thank the reviewers and associate editor for their comments which improved this manuscript.

References

- [1] Adler, M., & Ziglio, E. (1996). *Gazing into the Oracle: The Delphi Method and its Application to Social Policy and Public Health*. London: Jessica Kingsley Publishers.
- [2] Belassi, W., & Tukel, O. I. (1996). A new framework for determining critical success/failure factors in projects. *Int. J. Project Manage.*, 14(3), 141-151.
- [3] Bently, C. (2006). PRINCE2-project in controlled environments. *Frameworks for IT Management*, J. V. Bon and T. Verheijen, Eds. Netherlands: Van Haren Publishing.
- [4] Berzisa, S., & Grabis, J. (2008). An approach for implementation of project management information systems. *Proceedings of the 17th International Conference on Information Systems Development (ISD)*

2008).

- [5] Callegari, D., & Bastos, R. (2007). Project management and software development processes: integrating RUP and PMBOK. *Proceedings of the Int. Conf. Systems Engineering and Modeling* (pp. 1-8).
- [6] Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educ. Psychol. Meas.*, 20(1), 37-46.
- [7] Donica, R. A., & Belair, R. (1997). Application of PMBOK principles for RTDBE project. *Proceedings of the Project Management Institute 28th Annual Seminars and Symposium*.
- [8] Santos, F. R. S. D., & Cabral, S. (2008). FMEA and PMBOK applied to project risk management. *J. Inf. Syst. Technol Manage.*, 5(2), 347-364.
- [9] Dvir, D., Lipovetsky, S., Shenhar, A., & Tishler, A. (1998). In search of project classification: A non-universal approach to project success factors. *Res. Policy*, 27(9), 915-935.
- [10] Ehsan, N., Malik, O. A., Shabbir, F., Mirza, E., & Bhatti, M. W. (2010). Comparative study for PMBOK & CMMI frameworks and identifying possibilities for integrating ITIL for addressing needs of IT service industry. *Proceedings of the IEEE International Conference on Management of Innovation and Technology (ICMIT)*, Singapore (pp. 113-116).
- [11] Gustafson, D. H., Shukla, R. K., Delbecq, A., & Walster, G. W. (1973). A comparison study of differences in subjective likelihood estimates made by individuals, interacting groups, Delphi groups and nominal groups. *Organ. Behav. Hum. Perform.*, 9(2), 280-291.
- [12] Hakim, C. (1987). Research design: strategies and choices in the design of social research. *Contemporary Social Research*, M. Bulmer, Ed. London: Routledge.
- [13] Hass, K. B. (2007). The blending of traditional and agile project management. *PM World Today*, 9(5), 1-8.
- [14] Humphrey, W. S. (2005). Why big software project fail: The 12 key questions. *Crosstalk: J. Defense Software Eng.*, 3, 25-29.
- [15] Iijima, T. (2007). IT industry standards: which ones to use?. *J. Corp. Account. Financ.*, 18(5), 51-56.
- [16] Jenkins, M. (2005). Combining multiple models to develop a software project management methodology. *Proceedings Int. Conf. Software Engineering and Applications*.
- [17] Lam, S. S. Y., Petri, K. L., & Smith, A. E. (2000). Prediction and optimization of a ceramic casting process using a hierarchical hybrid system of neural networks and fuzzy logic. *IIE Trans.*, 32(1), 83-92.
- [18] Landis, J., & Koch, G. (1977). Measurement of observer agreement for categorical data. *Biometrics*, 33, 159-174.
- [19] Lloyd, S., & Simpson, A. (2005). Project management in multi-disciplinary collaborative research. *Proceedings of the Professional Communication Conference* (pp. 602-611).
- [20] Nambisan, S., Agarwal, R., & Tanniru, M. (1999). Organisational mechanisms for enhancing user innovation in information technology. *MIS Quart.*, 23, 365-395.
- [21] Nasir, M. H. N. M., & Sahibuddin, S. (2011). Addressing a critical success factor for software projects: A multi-round Delphi study of TSP. *Intl. J. Phys.*, 6(4), 1213-1232.
- [22] Nasir, M. H. N. M., & Sahibuddin, S. (2011). Critical success factors for software projects: A comparative study. *Sci. Res. Essays.*, 6(10), 2174-2186.
- [23] Nawrocki, J., Olek, L., Jasinski, M., Paliswiat, B., Walter, B., Piertrzak, B. et al. (2006). Balancing agility and discipline with XPrince. *Lecture Notes in Computer Science*, 3943, 266-277.
- [24] Phin, L. N., Zamani, A. Z., & Kheng, K. S. (2005). Experience of using PMBOK in the development of the Malaysia's multimedia super corridor (MSC) integration architecture framework for large scale government applications. *Proceedings of PMI Global Progress 2005-Asia Pacific*.
- [25] Pinto, J. K., & Mantel, S. J. (1990). The causes of project failure. *IEEE Trans. Eng. Manage.*, 34(7), 305-327.

- [26] Pinto, J. K., & Rouhiainen, P. J. (2001). *Building Customer-Based Project Organizations*. New York: John Wiley and Sons.
- [27] Project management institute. (2010). PMI fact file. *PMI Today*, 16, 2010.
- [28] Project management institute. (2009). The growing gap between project manager supply and demand. *PMI Today Supplement*, 2.
- [29] Project management institute. (2008). *PMBOK: A Guide to the Project Management Body of Knowledge*. Pennsylvania, USA: Project Management Institute.
- [30] Skulmoski, G. J., Hartman, F. T., & Krahn, J. (2007). The delphi method for graduate research. *J. Inf. Technol. Educ.*, 6, 1-21.
- [31] Tukel, O. I., & Rom, W. O. (2001). An empirical investigation of project evaluation criteria. *Int. J. Oper. Prod. Manage.*, 21(3), 400-416.
- [32] Wangenheim, C. G. V., Silva, D. A. D., Buglione, L., Scheidt, R., & Prikladnicki, R. (2008). Best practice fusion of CMMI-DEV v1.2 (PP, PMC, SAM) and PMBOK 2008. *Inform. Software Technol.*, 52(7), 749-757.
- [33] White, D., & Fortune, J. (2002). Current practice in project management – An empirical study. *Int. J. Project Manage.*, 20(1), 1-11.



Mohd Hairul Nizam Md. Nasir received his bachelor of computer science specializing in software engineering and master of computer science from University of Malaya in 2003 and 2005, respectively. He completed his PhD in computer science from Universiti Teknologi Malaysia, Malaysia in 2014.

He is currently serving as a lecturer in the Department of Software Engineering, Faculty of Computer Science and Information Technology, University of Malaya. His area of specialization includes software process, software process improvement, software project management. He is also a member of IEEE Computer Society and Association for Computing Machinery, Project Management Institute. As an active researcher, he has run several research projects and produced significant research output. He has published more than 70 scientific publications and won several medals at national and international exhibition.



Shamsul Sahibuddin received his BSc in Computer Science from Western Michigan University, USA, the MSc in computer science from Central Michigan University, USA, and the PhD in computer science from Aston University, Birmingham, UK.

Currently, he is a dean of Advanced Informatics School, Universiti Teknologi Malaysia. He is a member of the program committee for Asia-Pacific Conference in Software Engineering since 2003 and a member of ACM. His fields of specialization are computer supported cooperative work, computer network, and software quality.



Rodina Binti Ahmad received her bachelor degree from University of Hartford, Western Hartford in USA; the master degree from Rensselaer Polytechnic Institute, USA and completed her PhD in the area of information systems management from National University of Malaysia, Kuala Lumpur, Malaysia in 2006.

Currently, she is an associate professor at Software Engineering Department, University of Malaya, Kuala Lumpur, Malaysia. She has been a lecturer and researcher for more than twenty years. Her research interests include software process improvement, requirements engineering and organizational analysis.



Shukor Sanim Mohd Fauzi received his master degree specializing in software engineering from Universiti Teknologi Malaysia. He also hold a BSc. information systems engineering and diploma in computer science from Universiti Teknologi Mara, Malaysia.

Currently he is serving as a senior lecturer in the Department of Computer Science, Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Perlis Campus. His area of specialization includes software quality, software process improvement, software process assessment, software engineering, software configuration management, software project management and also software testing. He is one of the technical committee in software engineering for standard and industrial research institute of Malaysia. Besides, he also is a member of ACM, IEEE, SPI MSC Malaysia and a member of Malaysia Software Engineering Interest Group. He has published more than 40 scientific publications which includes journals, conference proceedings and technical reports in international and national level. He also has won several medal at national and international exhibition and he is currently pursuing his PhD in software engineering at the University of New South Wales, Australia and he is a member of NICTA's Managing Complexity research group.