

# A MODEL FOR ESTIMATING THE SCOPE OF THE PROJECT: A PILOT STUDY

**Ammar Odeh, Princess Sumaya University for Technology**  
**Ammar El-Hassan, Princess Sumaya University for Technology**  
**Ahmad Abushakra, Princess Sumaya University for Technology**  
**Ismail Keshta, AlMaarefa University**

## ABSTRACT

*In this paper, we have proposed a model for the implementation of SP 1.1, “Estimate the Scope of the Project” by utilizing the practitioners’ different experiences of and opinions on such implementation. The model has four elements, which are plan, create, review meeting, and rework/update—that help practitioners implement SP 1.1 effectively by providing advice. We have identified specific activities within each of these elements that need to be performed when estimating the scope of the project. The model’s initial evaluation revealed its ease of learning and using. It also satisfies stakeholders and can efficiently manage the process of estimating the scope of the project.*

**Keywords:** Software Processes Improvement, Project Scope Management, Capability Maturity Model Integration

## INTRODUCTION

The software products' quality is significantly influenced by how good the organization's software processes are for both maintenance and development. Therefore, software development currently faces a significant challenge: the offering of high-quality software product (Kitchenham & Pfleeger, 1996; Scacchi, 2001). The research conducted on software quality focuses on Software Process Improvements (SPI) (Noushin, 2003; García-Mireles, Moraga, García & Piattini, 2003).

SPI is an important aspect of software development process optimization because it effectively helps organizations customize their processes. The implementation of SPI is deeply determined by organizational characteristics and project factors (Nasir, Ahmad & Hassan, 2008; Mahmood, 2009; Cugola et al.; Keshta, Niazi & Alshayeb, 2018; Sommerville, 2004). Noted that SPI's prime goal is to improve not only the quality of the products but also the organization's productivity.

The authors of the following studies and believe that SPI is an important method for enhancing the quality of organizational software products, improving development speed, and reducing time and cost necessary to develop software products (Fox & Frakes, 1997).

In Niazi & Babar (2009), It is emphasized that one of the prime problems in the software industry environments is the design of a suitable SPI implementation plan, especially for small and medium-sized software development organizations. Therefore, various SPI model standards have been used to increase productivity and software quality (Iqbal, Ahmad, Nasir, Niazi, Shamshirband & Noor, 2015; Rahmani, Ashkan & Abdullah, 2016). Software development companies can use some well-known SPI modes such as CMMI. In Lee & Mei-Hui (2009), it is state that CMMI has been extensively studied. Therefore, a CMMI assessment can assist various types of software development organizations in improving the quality of their products (Moser, Winkler, Heindl & Biffel, 2011; O'Regan, 2011; Monteiro, Machado & Kazman, 2009).

CMMI assists software development companies in improving the quality of their software products, but so far, there are not many organizations that have decided to adopt CMMI. The identified reasons include the small size of the organization, the implementation of CMMI can take a long time, and high service costs Staples, (Niazi, Jeffery, Abrahams, Byatt & Murphy, 2009) ion takes much time. Moreover, Niazi et al., have stated that CMMI's limited success, along with the massive investment needed by companies, are the two primary reasons why a large proportion of software development organizations do not use CMMI. Additionally, Batista & Dias (2000), Highlighted that small software development organizations face more considerable difficulties with CMMI than large ones.

As a result, these small firms consider themselves to be far away from adopting different SPI models like CMMI, since the processes are laborious and costly Al-Tarawneh, Abdullah & Jasem (2013). Unlike large firms, small organizations cannot adopt CMMI as the costs can run into hundreds of thousands of dollars. Moreover, they do not have sufficient resources to make the necessary investments. Development of the models entails designing new process areas (PAS) for CMMI Level 2 that would support companies and enable them to adopt the CMMI Level 2 PAs speedily (Eileen, Brandon & Buteau, 2011; Vivatanavorasin, Prompoon & Surarerks, 2006; Chrissis, Konrad & Shrum, 2006 & Xu, Hu, Yu, Lv, Qu & Zhu, 2013).

Since there are no approaches to help give small and medium-sized software firms in the implementation of CMMI Level 2, more research is needed on how it can be implemented effectively. Software companies need to understand the best implementation process of CMMI Level 2 to improve productivity and the quality of their software (Keshta, Niazi & Alshayeb, 2017; Keshta, Niazi, & Alshayeb, 2018; & Keshta, 2019).

The purpose of this paper is to implement one specific practice of the CMMI "Project Planning" (PP) PA. This paper's main contribution is to demonstrate how SP 1.1 of the PP PA of CMMI will be constructed. To help us accomplish these goals, we will be examining the following Research Questions (RQS):

- RQ1: How does one go about implementing SP 1.1 and estimating the project's scope?
- RQ2: How is the outcome of the proposed model for SP 1.1 perceived to be "easy to learn?"
- RQ3: Is the outcome of the proposed model for SP 1.1 useful?

In-depth interviews with practitioners in software development organizations are conducted with the specific purpose of: Obtaining an understanding of their SP 1.1 adaptation experiences. Recognizing their reservations about the implementation of SP 1.1 Investigating and Examining the various phases/steps required for SP 1.1 execution.

The remainder of the paper will be organized as follows: Section 2 will discuss the purpose of the research paper. Section 3 will discuss the research's context. Section 4 will describe the methodology used in the study. Section 5 will detail the development of our proposed model, and Section 6 will evaluate the model. Section 7 discusses the study's limitations. Finally, Section 8 will discuss the conclusion of the paper as well as any future research directions.

## **MOTIVATION**

The main reasons for selecting SP 1.1 are as follows: Crespo et al. show that the PPA helps develop and maintain any project plans appropriately as well as set the whole foundation for the project. Many studies, like Jarvis et al, have highlighted the critical role played by the PA in the project management process, emphasizing that enhancing PP is a vital tool that should apply to high-risk projects. Other studies have shown the problems that can occur from a lack of PP. In Hallows, (2015), the authors state that one of the leading causes of project failure is weak PP. The possibility of project failure rises considerably due to poor planning of a project. Even though a lack of planning can be a risk to software development projects, Niazi et al. revealed that only 8 out of 14 of the PP process

practices were of "medium" value to the study's participants. More attention, concentration, and consideration should be on PP. If this does not occur, many future projects will fail because of weak PP.

Scope estimation is critical to project planning because it has a direct impact on the project's budget and timeline. Despite this, it appears as though scope management is one of the most overlooked aspects of software project management. To define the project's scope, what the project can and cannot deliver needs to be understood Moustafaev (2014). The foundation of project scope management is the project scope definition process because it defines all the functions and works to be done if the product is to be a success Schwalbe (2015). Scope refers to every process that is involved in delivering all the work needed to provide the customer with a valued product Bingham (2010). Scope definition is a challenging but vital project management task, according to the Project Management Institute (PMI). The failure or success of a project is linked strongly to its scope definition (Dumont, Gibson & Fish, 1997; McQuighan & Hammell, 2011; Mirza, Pourzolfaghar & Shahnazari, 2013; Zahran, 1998).

## **BACKGROUND**

This section will provide background information, contextualize this research, and set the stage for the contribution of this research paper to SP 1.1 knowledge.

### **SPI**

With increased pressure on businesses, the majority of them have made customer satisfaction their motto (Paulk, Weber, Curtis & Chrissis, 1998; Pitterman, 2000). Software quality is critical, as it plays an increasing role in our daily lives (Yamamura, 1999); (Jiang, Klein, Hwang, Huang, & Hung). Despite the fact that several approaches exist for effectively addressing software quality issues, SPI is the standard method (Humphrey, 1995; Ville, 1999).

SPI enables software developers to position themselves optimally for initiating a process improvement program, which entails setting and defining attainable goals. In and (Chrissis, Konrad & Shrum, 2003), it is noted that "SPI entails analyzing existing processes and modifying them in order to improve product quality while reducing costs and development time."

### **Current Challenges of Software Process Improvement**

Due to the industry's current level of competition, many software organizations place a premium on attaining high maturity levels without enhancing their process capability. Measuring organizational capacity employing maturity levels is challenging, as the maturity framework does not examine all corporate practices in depth. The appraisal methods, as well as the CMMI framework, also need to be adjusted (Shih, Shaw, Fu & Cheng, 2013).

Over the last decade, models and standards for improving the software development process have grown dramatically. The development of these standards and models aided in the production of high-quality software, increased productivity, decreased costs, and saved time (Herbsleb & Goldenson, 1996). Similar advancements have not been made in adopting the models and standards. Which means the success of many SPI efforts has remained limited. About 67% of all the SPI management staff would like to have guidelines on effective implementation of SPI activities rather than receive advice on which of the SPI activities to adopt Leung. Even though implementing SPI is vital, the current research does not adequately address strategies for effective implementation of the SPI programs. Despite the presence of models or standards for implementing, the current strategies are not helpful (Hareton & Terence; (Wilkie, McFall & McCaffery, 2005).

A thorough review of the literature revealed that one of the SPI topics was missing. The focus of the discussion was on empirical studies that focus on the activities that are required rather than on how businesses can implement them. Recent research indicates that simply identifying the actions required to implement SPI is no longer sufficient. To effectively execute SPI programs in the future, organizations must first understand how to do so attention to the "how" is therefore critical.

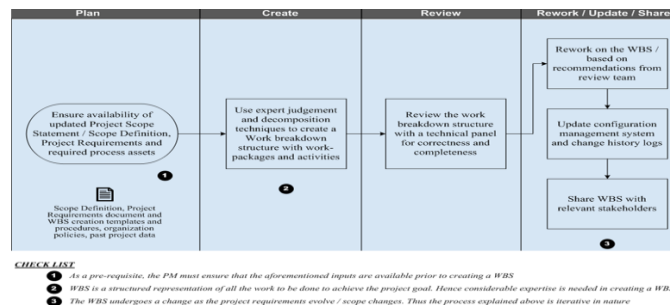
**Project Planning (PP)**

PP is a critical component of CMMI Maturity Level 2 project management. The primary objective of PP PA is "to develop and maintain plans that define project activities." PP PA is critical for efficient project management Lester, (Wilkie & McFall, 2010). The project plan must be followed throughout the duration of the project. Additionally, both generic and specific objectives must be met, and practices must be institutionalized and implemented.

**STUDY METHOD**

We conducted in-depth interview sessions with three professionals from three Saudi software development companies in order to ascertain the different stages and phases required to implement SP 1.1. The meeting took approximately one hour and were all taped as well as recorded. The interviews were then analyzed using the qualitative research method (Dasanayaka, 2008; Krippendorff, 1998).

The companies identified each interviewee as the most capable and skilled individual when it came to approximating the scope of the project. The interviewees were chosen as the most qualified individuals to respond to our queries about the process of approximating the scope of projects in their respective organizations. We consulted numerous research papers and journals to identify and create a list of appropriate and pertinent questions for the interview (Babar & Kitchen ham, 2009; Matthias, 2015). We determined the suitability of the questions by aligning them with the research project's goals. One of our colleagues, which facilitated us to make improvements, then reviewed the interview tool. Each of these organizations is a small business with fewer than 30 employees who are directly involved in software maintenance. Each organization has an active role in software development. Organization A gives support to its clients' IT project management through processes such as planning, initiation, control, execution, and closing. Additionally, the organization maintains a variety of administrative applications and provides support for the organization's current platforms. Organization B specializes in the development of software solutions for the hospitality and financial sectors. Additionally, the organization can repair industrial applications and assist with business and administrative tasks. Organization C is a software developer and architect with a skilled team of web designers and programmers on staff. The company provides clients with web solutions in the fields of design, as well as planning, managing, and implementing web content.



**FIGURE 1**  
**HIGH LEVEL MODEL FOR SP1.1**

Face-to-face interviews were conducted, while a questionnaire was used to ask a short series of questions. These questions resulted in the collection of sufficient data on the demographics of each company. Some other set of questions examined the processes used to determine the scope of the project.

During the interviews, a Dictaphone and notes were used to record all responses, and notes were taken during the interviews. Analysis of the response of each interviewee involved reading all the notes to consolidate the vital points made about the processes of estimating the project's scope. We listened to these tapes several times to ensure that nothing essential was missed out.

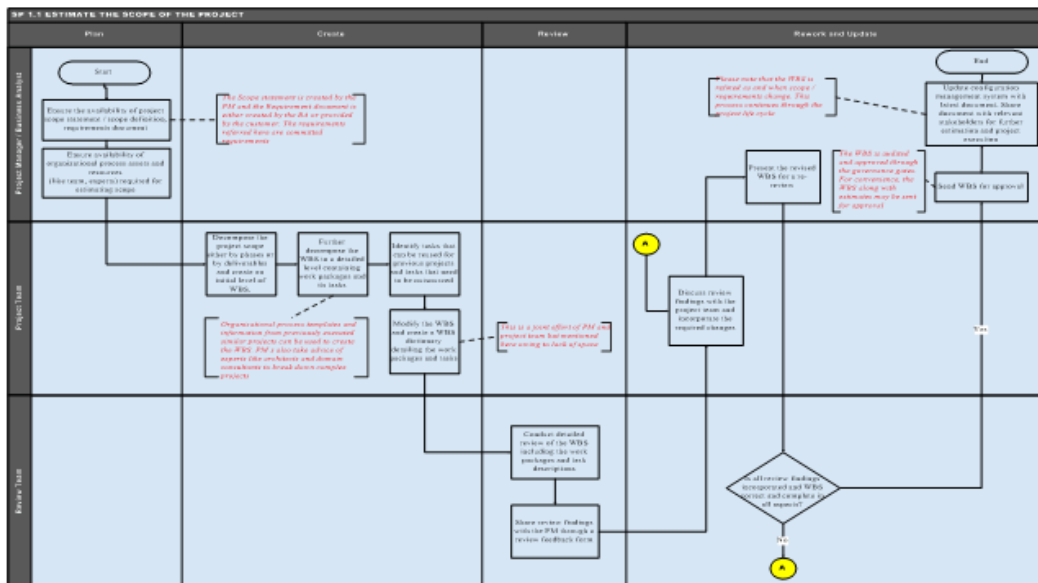
### PROPOSED MODEL

These are the phases of implementation of SP1.1: "plan," "create," "review meeting," and "update/rework." We will explain both of the high-level and comprehensive flow of the model we have suggested in this section.

#### High Level Flow

This section explains the high-level workflow (see Figure 1) for the process of approximating the scope of the project. The project coordinator should use the scope definition/statement, the software requirements document, and process assets such as templates, guidance, management practices, and previous project data to create the work breakdown structure during the planning phase. The process of developing a Work Breakdown Structure (WBS) entails breaking the larger scope down into smaller achievable units known as work packages and documenting them in a WBS template.

The review team then needs to share and verifies the WBS in preparation for the review phase. After that, the review team provides official reviews of bugs and explanations. The "rework and update" phase updates the document in response to the comments.



**FIGURE 2**  
**THE PROPOSED DETAILED FLOW WORKFLOW MODEL FOR SP1.1–**  
**“ESTIMATE THE SCOPE OF THE PROJECT”**

## Detailed Flow

This section details the process of estimating the scope of a project (see Figure 2). Each of these stages is associated with specific activities. The flow chart below illustrates the detailed process for estimating the project's scope. Each stage is described in detail in the subsequent subsections.

### Stage 1: Plan

Scope estimation is the process of determining all the work that require completion to achieve the project's end objectives. The document that provides a bird's-eye view of the work to be done to achieve the project scope is the WBS, created as part of the scope estimation process and describing the work involved in creating the project deliverables in a hierarchical/structured manner. A PM executes the necessary critical tasks during the planning phase such as:

- Determine the availability of data for previous projects and organizational policies governing the procurement of external services (contracting/subcontracting).
- Ascertain that all participants in the scope estimation process have a comprehensive understanding of the project's scope and requirements.

Not obtaining the desired resources or the non-availability of process assets or polices (like those that govern procurement) could be significant risks in many small and medium-sized organizations. The PM should thus identify such risks upfront and bring it to the notice of the relevant stakeholders/project sponsors for further action.

### Stage 2: Define

The first step is to decompose the broad scope into a first-level WBS either by work products (like requirement specs, design specs, codes, test case reports, etc.) or by phases (requirements, design, development, testing, implementation, etc.) or any other approach determined by the PM.

The second step is to further decompose the first-level WBS into the lowest units, known as work packages. A work package is the smallest unit of a WBS that can be used for estimation and execution. For example, under requirements management, documenting the requirements into a formal project requirement document could be a work package. Creating a requirement study questionnaire could be another. These packages can be easily accomplished, logically assigned to a resource, and tracked to closure. Similarly, designing database tables and creating external interfaces could be distinct work packages.

The third step involves subdivision of the work into specific tasks. For example, in PRD development, the tasks could be to create the various sections of the document using various input sources, review the document, etc. Next, these work packages are analyzed to determine in-house activities, which can be reused (like reusable code components), and what needs to be outsourced. Finally, a WBS dictionary is created with descriptions of the work packages and associated tasks and task attributes. The WBS has metadata for the work packages and it is codified (codes are assigned to each of the nodes) for easy reference, tracking, and maintenance.

### Stage 3: Review Meeting

This stage forms the crux of the whole process, and the feedback reaches the PM. The entire review team gathers at a common location along with the project representatives, including the PM and lead (if required). The review lead presents the review findings to the entire team. Stakeholders then discuss the findings in detail, and the PM/project team members provide their viewpoints against the feedback or observations. The feedback could be in the form of the following:

- Issue — bugs in the eye of the reviewer that need fixing and would affect the end product
  - Observation — a possible improvement that may not affect the end product
- After the discussion, a status is marked against each feedback point as follows:
- Agree —the PM/project team accepts the feedback and will take action to correct them.
  - Hold/Pending —Items are still under debate and need some more introspection. Unless the arguments appear valid from both ends and further input is required to agree, putting a review comment on hold is not advisable. The review lead takes this decision and advice on a timeline when the item should be closed.
  - Dropped — Occurs after withdrawal of a feedback originally considered an issue that needs a further explanation from the PM/project team.
  - Obsolete —this status is marked against feedback that may not be relevant to the context and creep in unintentionally. Even duplicate feedback could be pushed in this category, citing reasons within the comments columns.

#### **Stage 4: Rework/Update**

This stage requires careful consideration to ensure that no comments are overlooked and that the work product is designed to align with the customer needs following response incorporation. Failure to achieve this stage makes the review process ineffective. Once the feedback is incorporated, the review team once again reviews the document and signs off to mark a satisfactory completion of the review process. Every member of the review team must sign off post verification. Regular follow-ups by the PM/review lead to ensure timely incorporation of feedback is again a key aspect in ensuring timely completion of the process. After modifying and validating the document, the work products and all accompanying documentation are addressed for final approval.

### **MODEL EVALUATION**

#### **Expert Panel Review Process**

This approach applies to product evaluation. The evaluation involves experts with adequate experience and knowledge in the specific area (Nan, Hallb & Barker, 2008).

When a company is evaluating the software's quality, expert opinion is generally most valued and recognized. As a result, members of the software development community prefer to offer more credence and weight to research that employs the expert opinion technique.

Initial evaluation of the model involved an expert review, a process used to seek the opinions of two SPI experts about both the "user satisfaction" and "ease of learning" of the requirements change model proposed (Rosqvist, Koskelo & Harju, 2003). One of the SPI experts was highly experienced in the field of software project management, having worked in the area for 25 years. The second expert had 16 years' experience in both software project management and SPI-related activities. A questionnaire was designed to seek SPI experts' opinions about the requirements change model. Some of these questions were taken from (Keshta et al) Error! Reference source not found. And then tailored to fit the goals of the study. The researchers were asked to evaluate these questions against "user satisfaction" and "ease of learning" (Kitchenham, Pfleeger, Mccoll & Eagan, 2002).

- Ease of Learning: The experts rated the model as being both clear and easy to understand and did not think that a lot of prior knowledge of SPI was required to follow our model. Additionally, they said the division of the model had helped them into four core sections. This feedback showed that the model was both comprehensive and concise.

- User Satisfaction: The experts thought that the model would generally be useful in the software industry. Both of them felt that the model was clear and could estimate the scope of the project adequately. Neither of them thought the model was missing essential parts, and they felt that other firms might need to adapt the model to fulfill their specific requirements.

## LIMITATIONS

This research has several distinct limitations noted as follows: The expertise and experiences of the expert reviewers may impose constraints on the study findings. We have confidence in the feedback received, however, because the expert reviewers have sufficient knowledge of SPI and RE. The researcher made no recommendations regarding the expert reviewers (Davis, Bagozzi & Warshaw, 1989).

## CONCLUSION AND FUTURE WORK

To address RQ1, a model for estimating the project's scope has been developed based on the processes of such estimation. The interview sessions we performed with three industry members provided us with some fascinating insights into the processes they use to determine the scope of the project. We used an expert review process to conduct an initial assessment of our proposed model to address RQ2 and RQ3. After consulting with two SPI experts regarding the proposed model for SP 1.1's "user satisfaction" and "ease of learning," these experts rated our model as both clear and simple to understand (Davis, 1989). Additionally, these experts believed that the model would be beneficial in the software development as a whole, but that some firms may need to modify it to meet their unique requirements (Regnell, Runeson & Thelin, 2000).

## REFERENCES

- Al-Tarawneh, M.Y., Abdullah, M.S., & Jasem, A. (2013). "Software Development Process Improvement Framework (SDPIF) for Small Software Development Firms (SSDFs)." *International Journal of Computer Science Issues (IJCSI)*, 10(1).
- Babar, M.A., & Kitchenham, B. (2007). "Assessment of a framework for comparing software architecture analysis methods." *11th International Conference on Evaluation and Assessment in Software Engineering (EASE)*.
- Batista, J., & Dias, D.F. (2000). "Software process improvement in a very small team: A case with CMM". *Software Process-Improvement and Practice*, 5, 243-250.
- Bingham, E. (2010). "Development of the Project Definition Rating Index (PDRI) for infrastructure projects," Ph.D. dissertation, Arizona State Univ., Phoenix, AZ, USA.
- Chrissis, M., Konrad, M., & Shrum, S. (2003). *CMMI Guidelines for Process Integration and Product Improvement*. Addison-Wesley, Reading.
- Chrissis, M.B., Konrad, M., & Shrum, S. (2006). *CMMI guidelines for process integration and product improvement, (2nd Edition)*. Pearson Education.
- Crespo, D., & Ruiz, M. (2012). "Decision making support in CMMI process areas using multiparadigm simulation modeling." *Proceedings of the 2012 Winter Simulation Conference (WSC)*, 1-12.
- Cugola, G., & Ghezzi, C. (1998). "Software processes: A retrospective and a path to the future." *Software Process: Improvement and Practice*, 4(3), 101-123.
- Dasanayaka, S. (2008). "SMEs in a globalized world: A brief note on basic profiles of Pakistan's small and medium scale enterprises and possible research directions". *Business Review*, 3(1), 69-77.
- Davis, F.D. (1989). "Perceived usefulness, perceived ease of use and user acceptance of information technology". *MIS Quarterly*, 13(3), 319-340.
- Davis, F.D., Bagozzi, R.P., & Warshaw, P.R. (1989). "User acceptance of computer technology: A comparison of two theoretical models". *Management Science*, 35, 982-1003.
- Day, B., Ke-Zun, S.C., Lovelock, L., & Lutteroth, C. (2009). "Climbing the ladder: CMMI level 3," *2009 IEEE International Enterprise Distributed Object Computing Conference*, 97-106.
- Dumont, P.R., Gibson, G.E., Jr., & Fish, J.R. (1997). "Scope management using project definition rating index." *Journal of Management in Engineering*, 13(5), 54-60. 1997.



- Dutra, E., & Santos, G. (2015). Software process improvement implementation risks: *A qualitative study based on software development maturity models implementations in Brazil*. In *International Conference on Product-Focused Software Process Improvement*, 2, 43-60.
- Eileen, F.L., Brandon, S., & Buteau, S. (2011). *Introduction to CMMI for services: Guidelines for superior service*, (2nd Edition), Published by Addison-Wesley Professional. Part of the SEI Series in Software Engineering series, March.
- Fox, C., & Frakes, W. (1997). "The quality approach: Is it delivering?" *Communications of the ACM*, 40(6), 25-29.
- García-Mireles, G.A., Moraga, M.A., García, F., & Piattini, M. (2003). "The influence of process quality on product usability: A systematic review." *CLEI Electronic Journal*, 16(2), 6-6.
- Hallows, J. (2015). *Information systems project management: How to deliver function and value in information technology projects*. New York, NY, USA: American Management Association.
- Hareton, K.N.L., & Terence, C.F.Y. (n.d). "A process framework for small projects". *Software Process-Improvement, And Practice*, 6, 67-83.
- Herbsleb, J.D., & Goldenson, D.R. (1996). "A systematic survey of CMM experience and results". *Proceedings of the 18th International Conference on Software Engineering (ICSE-18)*.
- Humphrey, W.S. (1995). *A discipline for software engineering*. Addison Wesley.
- Iqbal, J., Ahmad, R.B., Nasir, M.H.N.M., Niazi, M., Shamshirband, S., & Noor, M.A. (2015). Software SMEs' unofficial readiness for CMMI-based software process improvement". *Software Quality Journal*, 1-27.
- Jarvis, A., & Crandall, V. (1995). *Inroads to software quality: "How to" guide and toolkit*. Prentice-Hall, Inc.; 1997 Apr 1. Standish-Group Chaos, *the Standish Group Report*. Standish Group International.
- Jiang, J., Klein, G., Hwang, H.G., Huang, J., & Hung, S. (n.d). An exploration of the relationship between software development process maturity and project performance". *Information and Management*, 41, 279-288.
- Keshta, I. (2019). "A model for defining project lifecycle phases: Implementation of CMMI level 2 specific practice." *Journal of King Saud University-Computer and Information Sciences*.
- Keshta, I., Niazi, M., & Alshayeb, M. (2017). "Towards implementation of requirements management specific practices (SP1.3 and SP1.4) for Saudi Arabian small and medium sized software development organizations." *In IEEE Access*, 5, 24162-24183.
- Keshta, I., Niazi, M., & Alshayeb, M. (2018). "Towards implementation of process and product quality assurance process area for Saudi Arabian small and medium sized software development organizations." *In IEEE Access*, 6(4), 1643-41675.
- Keshta, I., Niazi, M., & Alshayeb, M. (2018). "Towards implementation of process and product quality assurance process area for Saudi Arabian small and medium sized software development organizations." *In IEEE Access*, 6, 41643-41675.
- Kitchenham, B., & Pfleeger, S. (1996). "Software Quality: The Elusive Target," *IEEE Software*, 13(1), 12-21.
- Kitchenham, B., Pfleeger, S.L., Mccoll, B., & Eagan, S. (2002). "An empirical study of maintenance and development estimation accuracy". *Journal of Systems and Software*, 64(1), 57-77.
- Krippendorff, K. (1998). *Content analysis: An introduction to its methodologies*. Sage, London.
- Lee, C.S., & Mei-Hui, W. (2009). "Ontology-based computational intelligent multi-agent and its application to CMMI assessment." *Applied Intelligence*, 30(3), 203-219.
- Lepmets, M. (2010). "Which process model practices support project success?" *Proceedings of the 17th European Conference on Communications in Computer and Information Science*.
- Lester, N.G., Wilkie, F.G.D., & McFall, M.P. (2010). "Investigating the role of CMMI with expanding company size for small- to medium-sized enterprises". *Journal of Software: Evolution and Process*, 22(1), 17-31.
- Leung, H. (n.d). "Slow change of information system development practice", *Software Quality Journal*, 8(3). 97-210.
- Mahmood, N. (2009). "Software process improvement implementation: Avoiding critical barriers." *CROSSTALK. The Journal of Defense Software Engineering*, 22(1), 24-27.
- Mahmood, N. (2015). "A comparative study of software process improvement implementation success factors." *Journal of Software: Evolution and Process*, 27(9), 700-722.
- Matthias, R. (2015). *Tools & Techniques Expert Opinion*.
- McQuighan, J.M., & Hammell, R.J. (2011). "Scope as a leading indicator for managing software development." (SERA), 235-241.
- Mirza, M.N., Pourzolfaghar, Z., & Shahnazari, M. (2013). "Significance of scope in project success." *Procedia Technology*, 9, 722-729.
- Monteiro, P., Machado, R.J., & Kazman, R. (2009). "Inception of software validation and verification practices within CMMI Level 2." *2009 Fourth International Conference on Software Engineering Advances*, 536-541.
- Moser, T., Winkler, D., Heindl, M., & Biffel, S. (2011). "Requirements management with semantic technology: An empirical study on automated requirements categorization and conflict analysis." *In Advanced Information Systems Engineering*. Springer, Berlin Heidelberg.
- Moustafaev, J. (2014). *Project scope management: A practical guide to requirements for engineering, product, construction, IT and enterprise projects*. Boca Raton, FL, USA: CRC Press.

- Nan, M., Hallb, T., & Barker, T. (2008). "Using an expert panel to empirically validate a requirements engineering mediation model". *In Proc. IADIS Int. Conf. ICT*. 99-107.
- Nasir, M.H.N.M., Ahmad, R., & Hassan, N.H. (2008). "Resistance factors in the implementation of software process improvement project." *In Information Technology, International Symposium on*, 4, 1-10. IEEE.
- Niazi, M., & Babar, M. (2009). "Identifying high perceived value practices of CMMI level 2: An empirical study". *Information and Software Technology Journal*, 51(8), 1231-124.
- Niazi, M., & Babar, M.A. (2009). "De-motivators for software process improvement: An Analysis of Vietnamese Practitioners' Views". *International Conference on Product Focused Software Process Improvement PROFES*, LNCS 4589, 118-131.
- Niazi, M., Wilson, D., & Zowghi, D. (2006). "Critical success factors for software process improvement: An empirical study". *Software Process Improvement and Practice Journal*, 11(2), 193-211.
- Noushin, A. (2003). "The impact of software process improvement on quality: In theory and practice." *Information & Management*, 40(7), 677-690.
- O'Regan, G. (2011). *Introduction to software process improvement, undergraduate topics in computer science*. Springer-Verlag, London Limited.
- Paulk, M., Weber, C., Curtis, B., & Chrissis, M. (1998). *A high maturity example: Space shuttle onboard software, in the capability maturity model: Guidelines for improving software process*. Addison-Wesley.
- Paulk, M.C., Humphrey, W.S., & Pandelios, G.J. (1992). "Software process assessments: Issues and lessons learned". *In Proceedings of ISQE92*, 4B41-4B513.
- Pitterman, B. (2000). *Telcordia technologies: The journey to high maturity*. IEEE Software.
- Rahmani, H., Ashkan, S., & Abdullah, K. (2016). "CIP-UQIM: A unified model for quality improvement in software SME's based on CMMI level 2 and 3." *Information and Software Technology*, 71, 27-57, 2016.
- Regnell, B., Runeson, P., & Thelin, T. (2000). "Are the perspectives really different-further experimentation on scenario-based reading of requirements." *Empirical Software Engineering*, 5(4), 331-356.
- Rosqvist, T., Koskelo, M., & Harju, H. (2003). "Software quality evaluation based on expert judgement". *Software Quality Journal*, 11(1), 39-55.
- Scacchi, W. (2001). "Process models in software engineering." *Encyclopedia of Software Engineering*, 993-1005, December.
- Schwalbe, K. (2015). *Information technology project management*. Boston, MA, USA: Cengage Learn.
- SEI Report, (2010). "CMMI for development version 1.3 CMMI-DEV V1. 3.
- SEI. (2004). *Process maturity profile*. Software Engineering Institute Carnegie Mellon University.
- Shih, S.P., Shaw, R.S., Fu, T.Y.F., & Cheng, C.P. (2013). "A systematic study of change management during CMMI implementation: A modified activity theory perspective". *Project Management Journal*.
- Small-Medium Enterprises in Saudi Arabia Report, (2016). Jeddah Chamber.
- Sommerville, I. (2004). *Software Engineering, (Eighth edition)*. Pearson Education.
- Staples, M., Niazi, M., Jeffery, R., Abrahams, A., Byatt, P., & Murphy, R. (2009). "An exploratory study of why organizations do not adopt CMMI", *Journal of Systems and Software*, 80(6), 883-895.
- Ville, S. (1999). *Software Engineering, (9th Edition)*. Addison-Wesley.
- Vivatanavorasin, C., Prompoon, N., & Surarerks, A. (2006). "A process model design and tool development for supplier agreement management of CMMI: Capability Level 2." *2006 13th Asia Pacific Software Engineering Conference (APSEC'06)*, 385-392.
- Wilkie, F.G., McFall, D., & McCaffery, F. (2005). "An evaluation of CMMI process areas for small to medium-sized software development organizations." *Software Process: Improvement and Practice*, 10(2), 189-201.
- Xu, G., Hu, P.H., Yu, J., Lv, P., Qu & Zhu, M. (2013). "Supporting flexibility of the CMMI process framework with a multi-layered process model." *2013 10th Web Information System and Application Conference*, 409-414.
- Yamamura, G. (1999). *Software process satisfied employees*.
- Zahran, S. (1998). *Software process improvement Practical guidelines for business success*. UK, Addison-Wesley.