

Association for Information Systems AIS Electronic Library (AISeL)

AMCIS 2009 Doctoral Consortium

Americas Conference on Information Systems
(AMCIS)

2009

Information Systems Development Success: Perspective of Software Development Team Members

Lucian M. Zelazny
VA Tech, zelazny@txstate.edu

Follow this and additional works at: http://aisel.aisnet.org/amcis2009_dc

Recommended Citation

Zelazny, Lucian M., "Information Systems Development Success: Perspective of Software Development Team Members" (2009).
AMCIS 2009 Doctoral Consortium. 20.
http://aisel.aisnet.org/amcis2009_dc/20

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2009 Doctoral Consortium by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

Information Systems Development Success: Perspectives of Software Development Team Members

Lucian M. Zelazny

VA Tech

lzelazny@vt.edu

ABSTRACT

The traditional models of IS success measure success from the viewpoint of the system, users, and the organization. The system viewpoint is measured by information quality, system quality, and service quality; the users' viewpoint by user satisfaction, use, and individual net benefits; and the organization's viewpoint by organizational net benefits. This study adds the development team's viewpoint. I decompose system quality into its functional and non-functional components and combine them with new constructs to create Information System Development (ISD) success.

Like IS Success, ISD Success is a comprehensive model composed of multiple interrelated dimensions: practitioner satisfaction, project manager satisfaction, and the antecedents to these constructs, which include functional system quality, non-functional system quality, and process quality. Unlike the traditional models of IS success, ISD Success can be used to evaluate systems during the development cycle as well as on projects that never reach completion or are never used.

Keywords

Information Systems Development Success, IS Success, Systems Development.

INTRODUCTION

Information systems success has been an important and often researched dependent variable in IS literature. This is evidenced by the number of studies that have used information systems success or one of its components as the dependent variable. However, the definition of success is context dependent and most of the research to date has focused on measuring success from the perspective of the user and the organization (a proxy for upper management) using the IS. This limited perspective causes a problem which can be exemplified by the fable of the blind men and the elephant.

A wise man asked six blind men to determine what an elephant looks like. The man who felt the leg said that an elephant is like a pillar; the man who felt the tail said that an elephant is like a rope; the man who felt the trunk said an elephant is like a tree branch; the man who felt the ear said an elephant is like a hand fan; the man who felt the belly said an elephant is like a wall; and the man who felt the tusk said an elephant is like a pipe. The wise man tells them that even though they gave different answers they were all correct. An elephant is like all of these things because it has all the features mentioned and each man touched a different part of the elephant (Wikipedia, 2009).

This same issue occurs with information systems success. Once you select a perspective (e.g. a stakeholder) you can begin to get a picture of IS success but that picture is only a part of the whole. DeLone and McLean (2003) and Seddon (1997) both agree that setting the context to measure net benefits is necessary. Seddon (1999) states that "different measures are likely to be needed to assess the impact and effectiveness of a system for different groups of stakeholders." This research will focus on measuring success from the development team's perspective to add to the picture of the success of a system. The model of Information System Development (ISD) Success will be proposed and an instrument will be created to measure the constructs within the model. Before going into ISD Success we need to explore the predominant model of success, the DeLone and McLean IS Success Model.

The remainder of this paper is structured as follows; 1) A discussion of the DeLone and McLean IS Success Model, 2) a taxonomy for models of success of an IS, 3) the Information System Development Success model is proposed, and 4) hypotheses are made.

LITERATURE REVIEW AND MODEL DEVELOPMENT

The DeLone and McLean IS Success Model

Summary

In an effort to clarify the dependent variable in IS research, DeLone and McLean (1992) organized the measures of system success used in prior research into the comprehensive taxonomy known as the DeLone and McLean (henceforth D&M) Model of Information Systems (IS) Success. In the years subsequent to introduction of the D&M IS Success Model several articles challenged, critiqued, or extended the model (DeLone & McLean, 2003), the most notable being Seddon (1997). Seddon takes the position that combining a process model with a causal model creates confusion. He also believes that the model leads to three possible meanings for IS use; 1) use as a variable that proxies for benefits from use, 2) Use as a dependent variable in a variance model of future use, and 3) use as an event in a process leading to individual or organizational impact. DeLone and McLean updated their model (Figure 1) (DeLone & McLean, 2003) to address criticisms and suggested extensions of the original model.

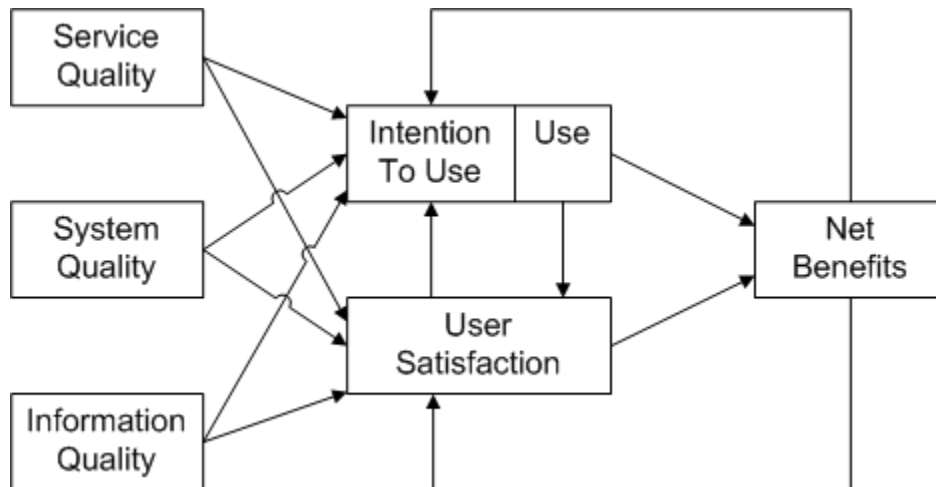


Figure 1. The Updated DeLone and McLean IS Success Model

Issues

Process vs. Causal Models: Seddon (1997) takes the position that combining a process model with a causal model leads to confusion. The result is that the boxes and arrows “have both a variance and an event-in-process interpretation” which leads to “a level of muddled thinking that is likely to be counter-productive for future IS research” (Seddon, 1997). On the other hand, DeLone and McLean take the position that it is appropriate to combine the two types of models and the combination adds to understanding. The theory behind development of the original model was based on a process understanding of information. However, “the application of our model to empirical research also requires a contextual variance specification of the model” (DeLone & McLean, 2003). The direction of the relationships between the dimensions is to ‘be determined’ by the individual researcher based on context.

For this research, I take the stance that ‘process’ is an inherent part of an information system and thus knowing the process adds to understanding. However, combining the models into one may not always be appropriate. Once a process construct is placed in-between a causal relationship or vice versa (a causal construct is placed in-between a process relationship) then separate models are necessary. Since a process model does not imply causality, it is hard to evaluate the effect (either positive or negative) when a process construct is inserted between causal constructs. I propose that a process model can be combined with a causal model as long as the process constructs match the causal constructs. Otherwise, each of the individual models will add understanding to the phenomenon of interest but they should be kept separate. Therefore, I agree with both Seddon and DeLone and McLean. Under certain circumstances combining a process model with a causal model is appropriate (usually at a higher level meta model) but when confusion arises (usually a more detailed model) the models should be separated but both presented to aid in understanding.

Perspective: The DeLone and McLean IS Success Model does a good job of defining and measuring IS success from the perspective of users and the organization (e.g. upper management). The user's perspective is measured by user satisfaction, intention to use/use, and net benefits (from the individual's perspective). The organization's perspective is measured by organizational net benefits. However, there are other stakeholders who play an important role in the life of an information system whose viewpoints should be included. "Project stakeholders are individuals and organizations who are actively involved in the project, or whose interests may be positively or negatively affected as a result of project execution or successful project completion" (Baccarini, 1999). They are the members of the development team, particularly the project manager and practitioners¹. Adding the viewpoints of these stakeholders will help to uncover a deeper understanding of the success of a system.

State: DeLone & McLean (2003) assert that the process understanding of IS Success has three stages; 1) creation of the system, 2) use of the system, and 3) consequences of system use. Based on the constructs contained within D&M's model, the system must first be used before net benefits can be realized. However, the first stage in their process is that the system must be built. Thus, in order to use the D&M IS Success model the system must first reach the state of being completed. As stated in the introduction, there are many systems that either get cancelled or are seriously over their time budgets. Recent research suggests that some of the stakeholders involved in the development of the IS may feel that the project was a success even though the project was cancelled or seriously over budget (Linberg, 1999). The D&M IS Success model cannot be used to evaluate these systems because they have yet to reach or will never reach completion and have never been used.

A Taxonomy for Models of Success of an Information System

Past research has identified that measuring software development (SD) project success using time, budget, and fulfillment of system requirements is insufficient (Atkinson, 1999). IS success should be viewed as a multi-dimensional construct (DeLone & McLean, 1992, 2003; Seddon, 1997). Ballantine et. al. (1996) break success into three levels: 1) development, 2) deployment, and 3) delivery. Saarinen (1996) defines success using four dimensions: 1) development process, 2) use process, 3) quality of the IS Product, and 4) impact of the IS on the organization. Categories were used by Thomas and Fernandez (2008) to define success: 1) project management, 2) technical, and 3) business. In order to determine the completeness of the models, I propose that models of information system success should account for three dimensions; 1) product, 2) process, and 3) people.

"A system is an array of components that work together to achieve a common goal, or multiple goals, by accepting input, processing it, and producing output in an organized manner" (Oz, 2009). In this context, inputs come into the system (e.g. design documents), a process is performed (e.g. programming), people (e.g. programmers) perform the process, and the output is an artifact (e.g. the executable program). The artifact can be measured against the goals of the project (i.e. speed, accuracy, functionality, benefits, etc.) to determine success on this dimension. Process refers to the way we get to the end product (i.e. the artifact). The perspective for measurement is provided by the people dimension. Davis (1995) believes that there is a duality between product and process. "You can never derive or understand the full artifact, its context, use, meaning, and worth if you view it as only a process or only a product" (Davis, 1995). Baccarini (1999) states that the combination of project management success (process) and product success needs to be used to get the complete picture of project success (e.g. project success = project management success + product success).

Under this taxonomy, the IS success model does not show a complete picture of IS Success. The IS Success model measures the product through the System Quality and Information Quality constructs. However, it only measures benefits from the perspective of two stakeholders (e.g. users and the organization) and is lacking in that it only partially covers the process stages. Therefore, I propose the information systems development (ISD) success model.

Information System Development (ISD) Success

ISD Success measures the perception² of the result of a software development effort from the perspective of the software development team. The perception of success is judged on both the product (artifact) created as well as the process undergone to create it. Like IS Success, ISD Success is a comprehensive model composed of multiple

¹ As used here, practitioners include system architects, data base designers, analysts, programmers, etc...

² Many researchers distrust perceptual measures (Seddon, 1997). However, I take the position that once a human being is involved in passing judgment that judgment is merely his perception of things. Even if quantitative factors are used as input, once a human is involved in the evaluation the result is a perception.

interrelated dimensions: practitioner satisfaction, project manager satisfaction, and the antecedents to these constructs, which include functional product quality, non-functional product quality, and process quality. Both functional product quality and non-functional product quality are influenced by process quality. The model of ISD Success can be found in Figure 2. A separate model from D&M's is proposed in order to keep each model parsimonious and increase clarity by not allowing the models to cross process stages.

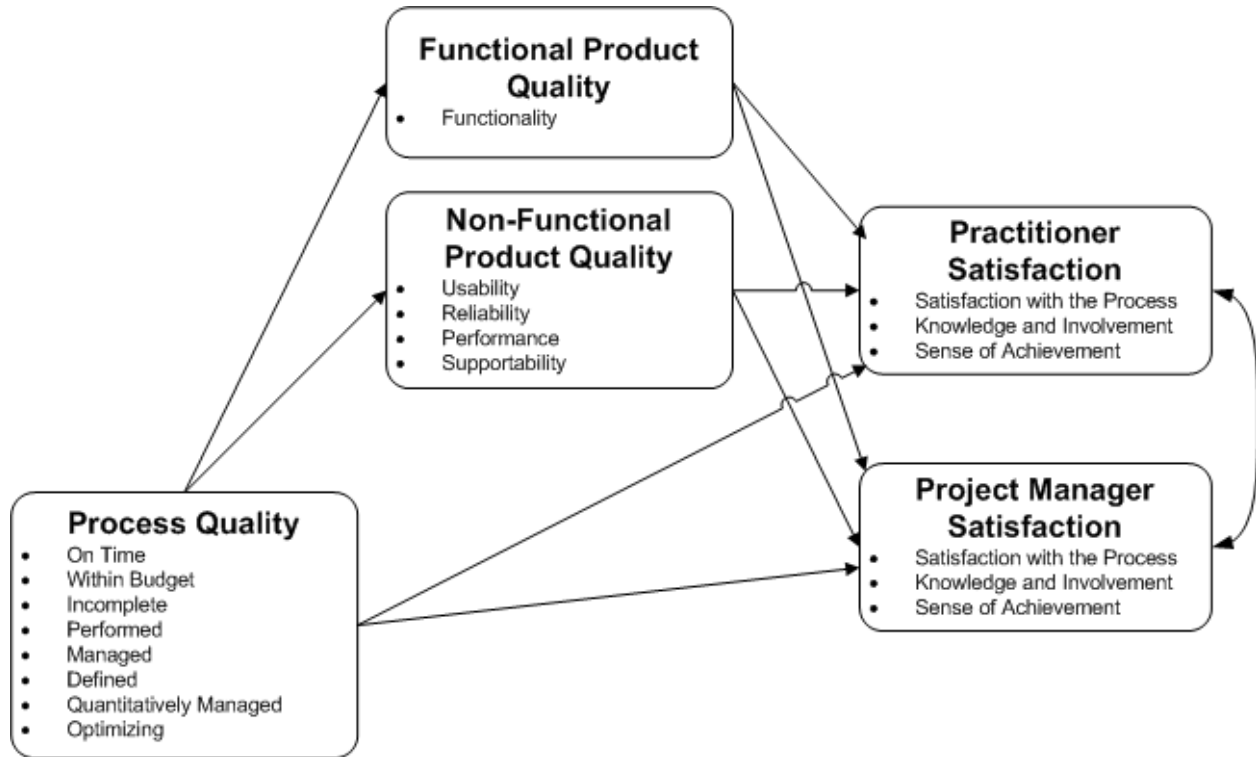


Figure 2. Information System Development (ISD) Success

People

This research increases our understanding of IS project success by evaluating success through the eyes of two stakeholders who play a major role in the development of information systems; practitioners and the project manager. The practitioner is an important stakeholder because he is the one that actually creates the IS and associated artifacts used by the end users. The project manager is important since they direct the process of completing the system and oversee the work of the practitioners.

Practitioner satisfaction includes the perceptions of software developers (including programmers), data base developers, systems analysts, etc. (Procaccino, Verner, Shelfer, & Gefen, 2005). Proxies for practitioner satisfaction and project manager satisfaction include satisfaction with the process, knowledge and involvement, and a sense of achievement (Jiang, Klein, & Discenza, 2002; Procaccino & Verner, 2002; Verner & Evanco, 2003). Research on project manager satisfaction has been lacking. Procaccino and Verner (2006) compared managers perceptions of project success with practitioners. They found that managers and practitioners had consistent perceptions of work related items of success. The work related items would fall under the satisfaction constructs in the ISD Success mode. Their project related items would fall under product quality and process quality in the ISD Success model. There were no differences between managers and practitioners for the project related items that would fall under product quality. Differences in importance and rankings occur with the project items that could serve as a proxy for process quality. Procaccino and Verner state that the similarities may be due to the high percentage of managers who had development experience. A contribution of this research will include a more detailed analysis of practitioner and project manager satisfaction.

Quality

The SWEBOK describes the use of the term software (product) quality as:

Over the years, authors and organizations have defined the term “quality” differently. To Phil Crosby (Cro79), it was “conformance to user requirements.” Watts Humphrey (Hum89) refers to it as “achieving excellent levels of fitness for use,” while IBM coined the phrase “market-driven quality,” which is based on achieving total customer satisfaction. The Baldrige criteria for organizational quality (NIST03) use a similar phrase, “customer-driven quality,” and include customer satisfaction as a major consideration. More recently, quality has been defined in (ISO9001-00) as “the degree to which a set of inherent *characteristics* fulfills *requirements*.”

The ISO defines quality in ISO 8402-1986 as “The totality of features and characteristics of a product or service that bear on its ability to satisfy stated and implied needs” (Oskarsson & Glass, 1996). They also define a quality system as “The organizational structure, responsibilities, procedures, processes and resources for implementing quality management” (Oskarsson & Glass, 1996). Pressman (1997) stresses three important points in defining software quality;

1. “Software requirements are the foundations from which quality is measured. Lack of conformance to requirements is lack of quality.
2. Specified standards define a set of development criteria that guide the manner in which software is engineered. If the criteria are not followed, lack of quality will almost surely result.
3. There is a set of *implicit requirements* that often goes unmentioned (e.g., the desire for good maintainability). If software conforms to its explicit requirements, but fails to meet implicit requirements, software quality is suspect.”

All of these definitions include two sides to quality, product quality and process quality.

Product Quality

Factor	Description
Functionality	Is assessed by evaluating the feature set and capabilities of the program, the generality of the functions that are delivered, and the security of the overall system.
Usability	Is assessed by considering human factors, overall aesthetics, consistency, and documentation.
Reliability	Is evaluated by measuring the frequency and severity of failure, the accuracy of output results, the mean time between failures (MTBF), the ability to recover from failure, and the predictability of the program.
Performance	Is measured by processing speed, response time, resource consumption, throughput, and efficiency.
Supportability	Combines the ability to extend the program (extensibility), adaptability, and serviceability (these three attributes represent a more common term – maintainability), as well as testability, compatibility, configurability [the ability to organize and control elements of the software configuration], the ease with which a system can be installed, and the ease with which problems can be localized.

Table 1. FURPS (Pressman, 1997)

Product quality is measuring the quality of the output of an iteration of the SDLC or any phase of the SDLC. The output can include programs, modules, diagrams, documentation, or specifications. There are a number of models, or suggested ‘checklists’, which attempt to measure the quality of software; McCall (1976), Boehm (1978), FURPS (1987), ISO 9126 (1991), Dromey (1996), Systemic (2003), and PQM (2007) (Yahaya, Deraman, & Hamdan, 2008). FURPS was developed by Hewlett-Packard (HP) in the mid 80’s to measure the quality of their software. The acronym FURPS stands for the individual software quality factors measured by HP; Functionality, Usability, Reliability, Performance, and Supportability. Two steps are involved in implementing FURPS; 1) setting priorities and 2) measuring quality attributes. Priorities need to be established since oftentimes tradeoffs are made between quality attributes. “For example, adding a new function might improve functionality but decrease performance,

usability, and/or reliability” (Grady & Caswell, 1987). Measureable goals are set after the priorities are decided upon. A more detailed description of the quality factors can be found in Table 1.

I have chosen to use the definitions of the factors from the FURPS model to guide the characteristics found in the product quality constructs of ISD Success. One of the reasons the FURPS model was chosen was the ease with which it can be split into functional (F) and non-functional (URPS) components. The other reason is that it is both a parsimonious model but complete in factors based on the factor descriptions. A criticism of FURPS presented by Ortega, Perez, and Rojas (2003) is that it fails to measure the artifacts (product) portability. The portability factor in ISO 9126 covers adaptability, installability, conformance, and replaceability. I do not consider this a disadvantage since some of these attributes are covered under the Supportability factor and today’s 3GL programming languages, such as Java and the .net languages, theoretically have portability built into them. In the model, functional product quality represents the system requirements which are domain specific. These will typically be the functional requirements (features and capabilities) the system must perform. The non-functional product quality construct represents ‘other’ requirements of the product such as speed, scale, reliability, and maintainability. This construct will be measured by adherence to the Usability, Reliability, Performance, and Supportability factors from FURPS.

Process Quality

The software development field has been enamored with the idea of improving the quality of software by improving the process of creating software. Evidence of this is seen by the evolution of the different software development methodologies. Structured systems tried to make the process more like an engineering process that was measured and documented. Object oriented systems combined data with process to increase the idea of information sharing and reuse. The hot methodologies of today are the Agile methodologies which try to deal with changing and evolving requirements. All of these methodologies have been used to try to improve the ‘process’ of building software.

At a higher level, there are two overarching frameworks which are believed to help improve one’s ‘process’; ISO 9000 and CMMi. The Capability Maturity Model Integration (CMMI) which is the integration of three previous models. The CMM was originally developed for the Department of Defense (DOD) by the Software Engineering Institute (SEI) located at the Carnegie Mellon Institute. The current version, CMMI v1.2, was released in August 2006 and is a framework used to “help organizations improve their development and maintenance processes for both products and services” (Software Engineering Institute, 2006). The CMMI consists of three constellations which are “a collection of CMMI components that are used to build models, training materials, and appraisal documents” (Software Engineering Institute, 2006). The constellation applicable here is the CMMI for Development (CMMI-DEV). Each constellation is composed of a set of process areas (PAs). A PA is “a cluster of related best practices that when implemented collectively satisfies a set of goals considered important for making significant improvement in that area” (Mutafelija & Stromberg, 2008). CMMI for Development v1.2 is composed of 22 PAs which are specific to individual constellations.

There are two approaches which can be used to apply CMMs, continuous and staged representations. The staged representation organizes PAs into maturity levels which can be obtained by an organization. There are five maturity levels an organization can reach; 1) Initial, 2) Managed, 3) Defined, 4) Quantitatively Managed, and 5) Optimizing. A maturity level is obtained by satisfying the requirements for the PAs contained at a specific level.

The continuous representation allows an organization to select a process area, or group of process areas, and focus on improving the processes related to it. The four categories of areas include; 1) Process Management, 2) Project Management, 3) Engineering, and 4) Support. Maturity of a process is designated through six capability levels; 0) Incomplete, 1) Performed, 2) Managed, 3) Defined, 4) Quantitatively Managed, and 5) Optimizing. The capability levels represent a continuous representation of a company’s process improvement along the process areas (See Figure 3 for an idea of the concept). Table 2 provides a listing of CMMI for Development’s 22 PAs across maturity levels and capability levels.

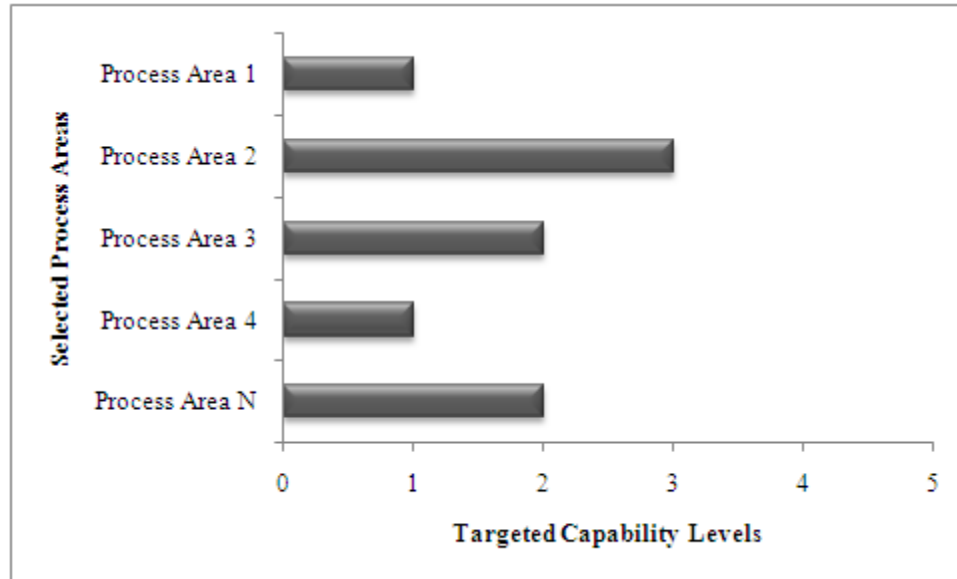


Figure 3. A Continuous Target Profile (Software Engineering Institute, 2006)

ISO 9000 and CMMI are competing but complementary frameworks. Usually the choice between frameworks is made based on the organizations target market. One of the main differences between ISO 9000 and CMMI is that CMMI provides a measure of capability level compared to ISO 9000 where you are either compliant or not. Research has shown that obtaining ISO 9000 certification would put you approximately at a CMMI level of 2 with some of the PA's for level 3 completed. Companies at a CMMI level of 2 will fulfill most of the requirements for ISO 9000 with some additional requirements to complete. The main thing to keep in mind is that both are overarching frameworks used to improve your process. They describe *what* you need to have but they do not tell you *how* to get there.

Process Quality will be a measure of how good the processes used by the company are. The areas to focus on for measurement will come from the Iron Triangle and CMMI. On time and within budget, which come from the Iron Triangle, will be used to proxy for the quality of the process. Projects which are completed on time and within budget will be seen as having an advanced process. The CMMI's continuous representation will also be used to measure process quality. Particularly the process management process area category. Projects with a higher capability level will be seen as having a better process than projects with lower capability levels. Process Quality impacts both functional product quality and non-functional product quality as well as project manager satisfaction.

Hypotheses

The hypotheses for the model of ISD Success are as follows:

- H1: Process quality will be positively associated with functional product quality.
- H2: Process quality will be positively associated with non-functional product quality.
- H3: Process quality will be positively associated with practitioner satisfaction.
- H4: Process quality will be positively associated with project manager satisfaction.
- H5: Functional product quality will be positively associated with practitioner satisfaction.
- H6: Functional product quality will be positively associated with project manager satisfaction.
- H7: Non-functional product quality will be positively associated with practitioner satisfaction.
- H8: Non-functional product quality will be positively associated with project manager satisfaction.
- H9: Practitioner satisfaction and project manager satisfaction will be correlated.

Process Area	Category				Maturity Level
	Process Management	Project Management	Engineering	Support	
Causal Analysis and Resolution (CAR)				X	5
Configuration Management (CM)				X	2
Decision Analysis and Resolution (DAR)				X	3
Integrated Project Management (IPM)		X			3
Measurement and Analysis (MA)				X	2
Organizational Innovation and Deployment (OID)	X				5
Organizational Process Definition (OPD)	X				3
Organizational Process Focus (OPF)	X				3
Organizational Process Performance (OPP)	X				4
Organizational Training (OT)	X				3
Product Integration (PI)			X		3
Project Monitoring and Control (PMC)		X			2
Project Planning (PP)		X			2
Process and Product Quality Assurance (PPQA)				X	2
Quantitative Project Management (QPM)		X			4
Requirements Definition (RD)			X		3
Requirements Management (REQM)			X		2
Risk Management (RSKM)		X			3
Supplier Agreement Management (SAM)		X			2
Technical Solution (TS)			X		3
Validation (VAL)			X		3
Verification (VER)			X		3

Table 2: Process area across categories and maturity levels (Mutafelija & Stromberg, 2008)

METHODOLOGY

Overview

This study will be conducted in two steps, interviews and surveys. The interviews will be conducted using semi-structured interviewing techniques with software developers and project managers. The objective of the interviews is to confirm and refine the hypothesized factors for practitioner and project manager satisfaction. This step is deemed to be necessary due to the lack of prior research in this area. An instrument will be developed using the data collected from the interviews and information gathered during the literature review. The detailed procedures used to create and validate the instrument will be presented the next section. The second step in this research will be conducted through surveys. Potential participants will be contacted through personal contacts, relationships with

corporations, and identified through professional groups. Participants will be separated into practitioners and project managers based on their response to demographic questions. The model will be tested using Partial Least Squares (PLS) regression. This analysis technique is chosen since PLS is recommended for theory-building and it handles both reflexive and formative indicators (Gefen, Straub, & Boudreau, 2000).

Instrument Development and Validation

Since ISD success is a proposed model there are no existing instruments for its measurement. Therefore, one of the contributions of this research will be to create and validate an instrument. I will follow the three steps as outlined by Aladwani and Palvia (2002); 1) conceptualization, 2) design, and 3) normalization. Following these steps will address the content validity, construct validity and reliability analyses of the instrument (Straub, 1989).

Conceptualization

During the conceptualization stage the constructs of interests are defined and a list of candidate items is generated. The results of the semi-structured interviews will be used to propose/confirm the factors contained in practitioner satisfaction and project manager satisfaction. The FURPS framework will form the basis for the factors for functional product quality and non-functional product quality constructs. The factors for process quality will be formed from a combination of items from the iron triangle (on time and within budget) and the CMMI framework. Candidate items will be created from the above mentioned definitions by myself and several other committee members (all of which have experience working on software development projects in industry). Clarifications to items will be made and repetitive items will be combined. The conceptualization stage will address content validity.

Design

The design phase involves the process of refining the candidate items from the conceptualization phase, determining the question types and sequence, and pilot testing the proposed instrument. After the questions are refined, they will be configured into questionnaire format. The alpha questionnaire will then be given to a representative sample and data collected. Reliability coefficients using Cronbach's alpha will be calculated. The data will be screened using Churchill's recommendations it see if reliability levels can be improved (Aladwani & Palvia, 2002; Churchill, 1979). Items will be removed and the reliability screening will continue until the reliability levels reach an acceptable level. The resulting instrument will then be tested for dimensionality through factor analysis. The results will be analyzed and items will be removed if their loadings do not meet a predecided cutoff or if they loaded on more than one factor. This procedure of factor analysis and removal of items will continue until a meaningful factor structure is reached. The resulting factors will then be analyzed against the factors from the literature review and semi-structured interviews. The end result of this stage will be the beta questionnaire. This stage will address construct validity and reliability.

Normalization

To verify and validate the structure from the design phase, another independent sample of practitioners and project managers will be selected. The data collected will undergo reliability analysis using Cronbach's alpha and dimensionality tests using factor analysis. Comparisons to the alpha tests structure will be made. Convergent and discriminate validity will be tested through the use of the multitrait-multimethod matrix (Aladwani & Palvia, 2002).

CONCLUSION

The results of this study will help to uncover the relationships of the different quality components of ISD Success and the importance of the quality components to the different members of the development team. Hypotheses 1 and 2 will determine if having a higher level of process quality leads to a higher quality product. I predict that there will be a significantly positive relationship between process quality and product quality (both functional and non-functional). Comparing the results of hypotheses 3-8 will determine which types of quality the different members of the development team prefer. I predict that practitioners will believe that product quality is more important for success (factor loading for H3 < H5 and H7) and project managers will believe that process quality is more important (factor loading for H4 > H6 and H8).

This research will make contributions to both researchers and practitioners. For researchers, this study will demonstrate the importance of taking a multi-stakeholder point of view when determining systems development success. An instrument will be developed and validated to measure ISD Success. A validated instrument will allow future research to consistently measure ISD success so that the results of multiple studies can be accurately compared. For practitioners, this study will not only identify how the development team defines success but it will uncover the relationships between the identified variables. Knowing the development team's definition of success

and the relationships found within will aid in the selection, measurement, and achievement of goals for the development team and the organization.

REFERENCES

1. Aladwani, A. M., & Palvia, P. C. (2002). Developing and validating an instrument for measuring user-perceived web quality. *Information & Management*, 39(6), 467-476.
2. Atkinson, R. (1999). Project management: cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria. *International Journal of Project Management*, 17(6), 337-342.
3. Baccarini, D. (1999). The logical framework method for defining project success. *Project Management Journal*, 30(4), 25.
4. Ballantine, J., Bonner, M., Levy, M., Martin, A., & et al. (1996). The 3-D model of information systems success: The search for the dependent variable continues. *Information Resources Management Journal*, 9(4), 5.
5. Churchill, G. A., Jr. (1979). A Paradigm for Developing Better Measures of Marketing Constructs. *Journal of Marketing Research*, 16(1), 64-73.
6. Davis, M. J. (1995). Process and Product: Dichotomy or Duality? *SIGSOFT Software Engineering Notes*, 20(2), 17-18.
7. DeLone, W. H., & McLean, E. R. (1992). Information Systems Success: The Quest for the Dependent Variable. *Information Systems Research*, 3(1), 60-95.
8. DeLone, W. H., & McLean, E. R. (2003). The DeLone and McLean Model of Information Systems Success: A Ten-Year Update. *J. Manage. Inf. Syst.*, 19(4), 9-30.
9. Gefen, D., Straub, D. W., & Boudreau, M.-C. (2000). Structural Equation Modeling and Regression: Guidelines for Research Practice. *Communications of the Association for Information Systems*, 4(7), 1-57.
10. Grady, R. B., & Caswell, D. L. (1987). *Software Metrics: Establishing A Company-Wide Program*. Englewood Cliffs, NJ: Prentice-Hall Inc.
11. Jiang, J. J., Klein, G., & Discenza, R. (2002). Perception differences of software success: provider and user views of system metrics. *Journal of Systems and Software*, 63(1), 17-27.
12. Linberg, K. R. (1999). Software developer perceptions about software project failure: a case study. *Journal of Systems and Software*, 49(2-3), 177-192.
13. Mutafelija, B., & Stromberg, H. (2008). *Process Improvement with CMMI v1.2 and ISO Standards*. Boca Raton, FL: Auerbach Publications.
14. Ortega, M., Pérez, M., & Rojas, T. (2003). Construction of a Systemic Quality Model for Evaluating a Software Product. *Software Quality Journal*, 11(3), 219-242.
15. Oskarsson, O., & Glass, R. L. (1996). *An ISO 9000 Approach To Building Quality Software*. Upper Saddle River, NY: Prentice Hall PTR.
16. Oz, E. (2009). *Management Information Systems* (6 ed.). Boston, MA: Thomson Course Technology.
17. Pressman, R. S. (1997). *Software Engineering: A Practitioner's Approach* (4th ed.). New York: McGraw-Hill.
18. Procaccino, J. D., & Verner, J. M. (2002). Software practitioner's perception of project success: a pilot study. *International Journal of Computers. The Internet and Management*, 10(1), 20-30.
19. Procaccino, J. D., & Verner, J. M. (2006). Software project managers and project success: An exploratory study. *Journal of Systems and Software*, 79(11), 1541-1551.
20. Procaccino, J. D., Verner, J. M., Shelfer, K. M., & Gefen, D. (2005). What do software practitioners really think about project success: an exploratory study. *Journal of Systems and Software*, 78(2), 194-203.
21. Saarinen, T. (1996). An expanded instrument for evaluating information system success. *Information & Management*, 31(2), 103-118.
22. Seddon, P. B. (1997). A Respecification and Extension of the DeLone and McLean Model of IS Success. *Information Systems Research*, 8(3), 240-253.
23. Seddon, P. B., Staples, S., Patnayakuni, R., & Bowtell, M. (1999). Dimensions of information systems success. *Communications of the AIS*, 2(3es), 5.

24. Software Engineering Institute (2006). CMU/SEI-2006-TR008, *CMMI for Development, Version 1.2*. Pittsburgh, PA: Carnegie Mellon University.
25. Straub, D. W. (1989). Validating Instruments in MIS Research. *MIS Quarterly*, 13(2), 147-169.
26. Thomas, G., & Fernández, W. (2008). Success in IT projects: A matter of definition? *International Journal of Project Management, In Press, Corrected Proof*.
27. Verner, J. M., & Evanco, W. M. (2003). An Investigation into Software Development Process Knowledge. In A. Aurum, R. Jeffery, C. Wohlin & M. Handzic (Eds.), *Managing Software Engineering Knowledge* (pp. 29-47): Springer-Verlag.
28. Wikipedia (2009). Blind men and an elephant Retrieved April 13, 2009, from http://en.wikipedia.org/wiki/Blind_Men_and_an_Elephant
29. Yahaya, J. H., Deraman, A., & Hamdan, A. R. (2008). Software Quality from Behavioural and Human Perspectives. *International Journal of Computer Science and Network Security*, 8(8).