The Moderating Effect of User Involvement on IS Success

Debbie Tesch
Xavier University
tesch@xavier.edu

Gary Klein, Richard Discenza, Ben Martz
{gklein, rdiscenz, wmartz}@uccs.edu

Abstract

User participation is an accepted practice for reducing risks during system development. Little is known with regard to which specific risks are mitigated and which user activities are most important. This report describes preliminary results that divide risks into technical expertise and more general process expertise to determine if early partnering or ongoing activities moderate risk more fully.

1. Introduction

As new technologies continue to become a significant factor in organizations, the growth of new-start application software development projects has soared in the last decade. Yet the success rate for such projects is unacceptably low. Software project failures cost companies millions of dollars each year and often prevent key business objectives from being met. The Standish Group's most recent "CHAOS" study update revealed that only 29 percent of development efforts were completed on time and on budget. Another 18 percent were canceled before completion of the development cycle. The remaining 53 percent were completed over-budget, behind schedule, and contained fewer functions than originally specified. These numbers are up from a decade ago [The Standish Group 2004]. Software project risk management advocates allege that identifying and analyzing components necessary for success can lead to actions that reduce the chance of failure.

The purpose of this study is to understand the role of general and technical expertise factors in the successful development of IS projects. Additionally, we examine the moderating effects of user partnering and user hands-on activities on general and technical expertise respectively. We pose the following research questions: *How do user partnering and user hands-on activities interact with general system development expertise and technical expertise to explain the varying degrees of software project success? Does system development (general)* *expertise positively impact project success? Does technical expertise positively impact project success? Does the degree of user involvement (partnering) influence the relationship between general expertise and the success of the project? Does the degree of active user participation (hands-on activities) influence the relationship between technical expertise and the success of the project?* Answers to these questions may allow managers to plan interventions that serve to mitigate user-related risks.

2. Background

The literature indicates that user support and participation is important [Barki 1989], but do not analyze the various practices that can serve to mitigate risks. Creating a positive environment in which to develop the systems can be accomplished early in the life cycle. It should be expected that the timing of such partnering activities would serve to alleviate risks associated with expertise risks that can be addressed at the commencement of the project [Jiang 2000b, 2002a, 2002b]. Unexpected technical risks due to a lack of expertise, however, are best addressed in an ongoing relationship that may best be served by including users in the design process [Lynch 2004; Henderson 1992]. This two pronged approach is illustrated in Figure 1.

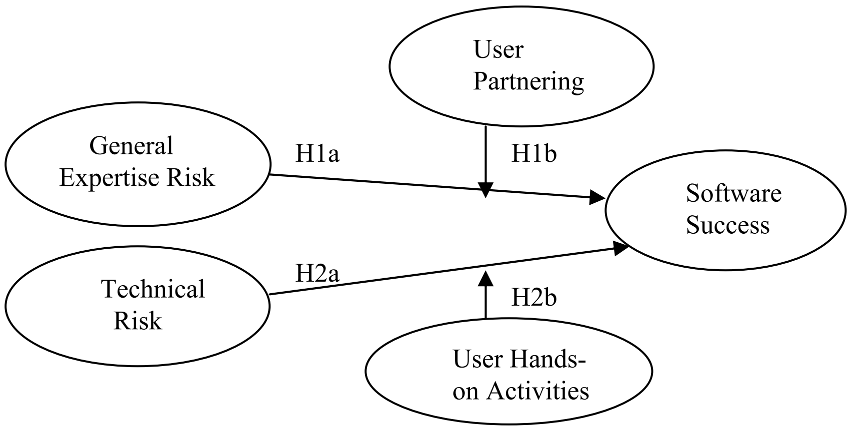

Figure 1. Research Model.

Project success is the ultimate goal and is associated with the degree of general and technical expertise available on the project. Likewise, the effects of user partnering and user hands-on activities may respectively moderate the relationship between general and technical expertise.

User Partnering (Involvement): User partnering describes activities to build a relationship between the IS staff and users to foster the sharing of responsibilities and expertise [Jiang, 2002b; Kirsch 1996; Robey 1996]. Partnering in general requires activities be conducted by the principal stakeholders in the project and include all development team members. Core partnering activities include 1) a team building session: key people meet to build a collaborative relationship, 2) conflict identification: key people identify potential conflict/problem areas, 3) problem solving process establishment: a documented process is put in place for joint resolution of problems, 4) formulate a charter/agreement that states shared objectives and responsibilities, and 5) provide for continuous improvement [Larson 1997]. Since our interest is limited to the impact of partnering on user issues, we use the term "user partnering".

User Hands-on Activities: On-going activities that involve users are illustrated by many control techniques and often incorporated into prototyping methodologies. Such involvement includes active involvement in the design of the system features as well as approval of features as the system advances. This serves to clarify any problems and remove uncertainty from the technical developers with respect to satisfactory outcomes. This is part of user involvement that has been extensively studied for a number of years in the field [Ives 1984].

General expertise: Lack of expertise variables are described in [Barki 1993] along five dimensions: development expertise in the team, expertise with the application, expertise with the task, lack of general expertise, and lack of user experience and support. General expertise dimensions considered in this study include the lack of general expertise, lack of development expertise, and lack of task expertise or expertise with application operations [Henderson 1992; Charette 1989].

Technical expertise: Technical expertise considered in this study is adapted from [Nidumolu 1995] and considers two key elements of technical expertise in a software development project: requirements analyzability and technological analyzability. Requirements analyzability refers to the expertise required to elicit requirements from the user. Often considered the most important capability in the software development process, the requirements analyzability component of technical expertise has the greatest impact on future stages of development. When users and analysts fail to establish a clear sense of project requirements, requirements are often ambiguous or incomplete. Technological analyzability refers to the ability to convert requirements specifications to software. Projects must ensure appropriate introduction of advanced technologies in order to avoid risks associated with technological uncertainty [Nidumolu 1995]. [McFarlan 1981] considers an organizations's experience with technology an important component in avoiding project uncertainty's associated with technological innovation.

Software Success: Project performance is viewed differently by the various stakeholders in a system development effort as well as by researchers in information systems [Henderson 1992; Nidumolu 1995]. Often described in terms of two key aspects, process performance, describes how well the software development process was undertaken and product performance describes the actual performance of the system. Product performance was initially described along three dimensions: operational efficiency of software, which describes the technical performance of the software; responsiveness of software, which describes how well the software meets the users' needs; and flexibility of software, which describes the software's ability to adapt to changing business needs [Nidumolu 1995, 1996]. Consistent with product performance objectives, [Henderson 1992] define project performance using software engineering issues of efficiency, effectiveness, and timeliness. Efficiency is often considered to be smooth team operations and adherence to allocated resources – time and cost. Effectiveness is considered to be the quality of work produced and meeting project and user objectives. The effectiveness component of the software engineering perspective corresponds to the product performance dimensions as described in [Nidumolu 1995].

More recently [Procaccino 2005] found that practitioners consider software projects successful if they provide intrinsic, internally motivating work to produce software systems that easy to use and meet customer/user needs. Their study pointed out that developers incorporate a sense a sense of achievement as they demonstrate that a good job results in professional growth in that they learned something new and the product is well accepted by those who must use it. These findings are in general agreement with earlier research regarding items that respondents considered less important. Software developers, in general, are more interested in the work they do than that of being promoted into a more managerial oriented position [Proaccino 2002; McConnell 1996; Couger 1998].

The research model thus provides four testable hypotheses:

H1a: General expertise has a positive relationship with software success.

H1b: User partnering (i.e. user involvement) moderates the relationship between general expertise and software success; the relationship becomes more positive as user partnering increases.

H2a: Technical expertise has a positive relationship with software success.

H2b: User hands-on activities moderate the relationship between technical expertise and software success; the relationship becomes more positive as user hands-on activities increase.

3. Research Methodology

3.1. Sample

Questionnaires were mailed to 2500 randomly selected Information Systems Special Interest Group (ISSIG) members of the Project Management Institute (PMI) in the U.S. PMI is a not-for-profit project management professional association with over 125,000 members worldwide. PMI provides global leadership in project management and their published body of knowledge is a recognized, global standard for the practice of project management. Members of

PMI represent a cross-section of managerial positions. Requested responses were to be based on respondents' most recent project experience. A postage paid envelope was enclosed with each questionnaire. Confidentiality of responses was assured.

A total of 171 questionnaires were returned for a response rate of about 7 percent. A summary of the demographic characteristics of the sample is presented in Table 1. Approximately one-fourth of the respondents were IS managers within their firms. The other three-fourths of the respondents were IS project leaders and professionals. Over 70% of participants either held or were pursuing PMP (project management professional) status.

Table 1. Demographics.

Gender:		**Team size on IS project:**	
Male:	67%	<= 7 members	36%
Female:	32%	8-15 members	39%
		16-25 members	16%
Position:		26 or more members	8%
IS managers:	26%		
IS Project Leaders:	60%	**Full-time employees:**	
IS Professionals:	12%	<=10	3%
		11-50	20%
PMP Status:		51-100	19%
Certified	46%	101-500	28%
Pursuing certification	25%	>500	29%
Intend to pursue certification	17%		
Not certified	12%	**Project duration:**	
		<1 year	51%
Industry type:		1-2 years	40%
Service	77%	2-3 years	5%
Manufacturing	19%	3-5 years	2%
Education	2%		

3.2. Measures

All measures were taken from existing scales in the literature. General Expertise and technical expertise are from [Barki 1993; Nidumolu 1995]. The user partnering activity measure is adopted from [Larson 1997]. The User Hands-On Construct is from [Jiang 2000b]. Software Success is measured by product performance as described in [Nidumolu 1995]. Cronbach alpha reliability coefficients for the scales used in this study are all greater than .80 indicating an acceptable level of reliability for social science studies [Nunnally 1978]. All items may be found in these original sources. Descriptive statistics are in Table 2.

Table 2: Descriptive statistics.

	User partnering (UP)	User hands-on (UH)	General risk (GRISK)	Technical risk (TRISK)	System success (SS)
Mean	2.88	3.21	3.32	3.25	3.27
Std Dev	1.05	1.11	1.06	0.88	.75
Correlations					
UP	.37				
GRISK	.41	.18			
TRISK	.50	.30	.61		
SS	.38	.35	.51	.53	

3.3. Statistical Analysis

Hierarchical regression analysis is used to test the hypotheses. In hierarchical regression, the order for which the predictor variables enter the regression is determined by the researcher based on logical or theoretical considerations. For hypothesis 1, regarding the relationship between general expertise risk and software success and the moderating relationship of user partnering on general expertise risk, technical expertise risk, user hands-on activities, and their interaction are entered as explanatory factors in the first stage. General expertise risk is entered in the second stage to assess its relationship with software success after controlling for technical expertise risk variables. Hypothesis 1a is tested by examining the significance of the increase in variance explained by adding general expertise risk. The user partnering variable is entered in the third stage, and the interaction of general expertise risk and user partnering is entered in the fourth stage. Hypothesis 1b is tested by examining the significance of the increase in variance explained by adding the user partnering moderating factor. If the moderating effect is as hypothesized, the interaction should be significant and positive. A similar hierarchical regression analysis is undertaken to test hypotheses 2a and 2b.

4. Results

H1a asserts that general expertise has a positive relationship with software success. As shown in Table 3, after controlling for the effects of technical expertise, user hands-on activities and their interaction, the introduction of general expertise yields a R^2 increase of .053. The general expertise coefficient is .343 with a standard error of .094, a t-value of 3.7 and significance of .001. Thus, H1a is supported. The Stage 3 regression results show that the introduction of user partnering moderates the relationship between general expertise and software success yielding an R2 change of .015. The general expertise coefficient in stage 3 is .303 with a standing error of .095, a t-value of 3.193, and significance of .002. However, the introduction of the interaction term does not improve the model, thus H1b is not supported.

Table 3. Hierarchical Regression Results for Hypothesis 1.

Model	R	R Square	Adjusted R Square	R Square Change	F Change	Sig. F Change
1	.586	.343	.330	.343	27.00	.000
2	.629	.396	.380	.053	13.40	.000
3	.641	.411	.392	.015	3.99	.048
4	.642	.412	.389	.001	.257	.613
1. Dependent variable: Software Success 2. Independent variables included: Stage 1: Technical Risk + User Hands-on + Interaction Stage 2: Stage 1 variables + General Risk Stage 3: Stage 2 variables + User Partnering Stage 4: Stage 3 variables + Interaction of General Risk and User Partnering						

Table 4. Hierarchical Regression Results for Hypothesis 2.

Model	R	R Square	Adjusted R Square	R Square Change	F Change	Sig. F Change
1	.561	.314	.301	.314	23.70	.000
2	.607	.368	.352	.054	13.16	.000
3	.620	.385	.365	.016	4.08	.045
4	.642	.412	.389	.027	7.05	.009
1. Dependent variable: Software Success 2. Independent variables included: Stage 1: General Risk + User Partnering + Interaction Stage 2: Stage 1 variables + Technical Risk Stage 3: Stage 2 variables + User Hands-On Stage 4: Stage 3 variables + Interaction of Technical Risk and User Hands-On						

Results for technical expertise and user hands-on effects on software success are shown in Table 4. H2a asserts that technical expertise has a positive relationship with software success. As shown in Table 4, after controlling for the effects of general expertise, user partnering and their interaction, the introduction of technical expertise in stage 2 yields a significant R^2 increase of .054. The technical expertise coefficient is .26 with a standard error of .072, a t-value of 3.6 and significance of .001. Thus, H2a is supported. The Stage 3 regression results in show that the introduction of user hands-on activities moderates the relationship between technical expertise and software success yielding an R^2 change of .016. The technical expertise coefficient in stage 3 is .232 with a standard error of .072, a t-value of 3.214, and significance of .002. When the interaction effect is added in stage 4, the R2 change is .027, the technical expertise coefficient in stage 4 is .646 with a standard error of .171, a t-value of 3.773 significant at .000. These findings indicate that H2b is supported.

5. Discussion

The intent of this paper is to understand the role of general and technical expertise characteristics considered important to project management professionals on the successful development of IS projects. Using an experimental research design, we are able to examine the interactions between these factors and the moderating effects of user partnering and user hands-on activities factors. In a preliminary analysis, we find that technical expertise interacts with user hands-on activities to improve the successful implementation of software projects, while general expertise exhibits a main effect on software success with no interaction with user partnering.

The model employed could be effective for researchers in determining whether specific processes can alleviate specific risk categories. Practitioners need to be aware that common understandings of risk mitigation may not hold and that simply installing risk mitigation frameworks for user involvement may not be effective in reducing the risks associated with general process expertise, while continuous participation by users does seem to reduce the impact of technical risks to system success.

6. References

[Barki 1989] Barki, H. & Hartwick, J. Rethinking the concept of user involvement. *MIS Quarterly*, 13, 1 (1989), 53-63.

[Barki 1993] Barki, H., Rivard, S. & Talbot, J. Toward an assessment of software development risk. *Journal of Management Information Systems*, 10, 2 (1993), 203-223.

[Charette 1989] Charette, R.N. *Software Engineering Risk Analysis and Management.* Multiscience Press Inc., New York, 1989.

[Couger 1988] Couger, J.D., Motivators vs. demotivators in the IS environment. *Journal of Systems Management,* 39, (1988), 36-41.

[Henderson 1992] Henderson, J.C. & Lee, S. Managing I/S design teams: A control theories perspective. *Management Science*, 38(6), 1992, 757-777.

[Ives 1984] Ives B. & Olson, M. H. User involvement and MIS success: A review of research," *Management Science*, 30, 5, (1984), 586-603.

[Jiang 2000a] Jiang, J.J., Klein, G., Chen, H.G., & Lin, L. Reducing user-related risks during and prior to system development. *International Journal of Project Management,* 20, 7, (2000), 507-515.

[Jiang 2000b] Jiang, J. J., Muhanna, W. A. & Klein, G. User resistance and strategies for promoting acceptance across system types. *Information & Management*, 37, (2000), 25-36.

[Jiang 2002a] Jiang, J.J., Chen, E., & Klein, G. The importance of building a foundation for user involvement in information systems projects. *Project Management Journal,* 33,1, (2002), 20-26.

[Jiang 2002b] Jiang, J.J., Klein, G., & Discenza, R. Project partnering impact on an information system project, project team and project manager. *European Journal of Information Systems,* 11, (2002), 86-97.

[Kirsch 1996] Kirsch, L.J. & Beath, C.M. The enactments and consequences of token, shared, and compliant participation in information systems development. *Accounting, Management and Information Technologies*, 6, 4 (1996), 221-254.

[Larson 1997] Larson, E. W. Partnering on projects: A study of the relationship between partnering activities and project success. *IEEE Transactions on Engineering Management*, EM-44, 2 (1997), 188-195.

[Lawrence 1967] Lawrence, P. & Lorsch, J. *Organization and Environment*. Cambridge: Harvard University Press, 1967.

[Lynch 2004] Lynch, T., & Gregor, S. User participation in decision support systems development: influencing system outcomes. *European Journal of Information Systems,* 13, (2004), 286-301.

[McConnell 1996] McConnell, S. *Rapid Development.* Microsoft Press, Redmond, WA (1996).

[McFarlin 1981] McFarlan, F. W. Portfolio approach to information systems. *Harvard Business Review*, 59, 5 (1981), 142-150.

[Nidumolu 1995] Nidumolu, S. The effect of coordination and uncertainty on software project performance: Residual performance risk as an intervening variable. *Information System Research*, 6, 3 (1995), 191-219.

[Nidumolu 1996] Nidumolu, S. A comparison of the structural contingency and risk-based perspectives on coordination in software-development process. *Journal of Management Information Systems*, 13, 2(1996), 77-113.

[Nunnally 1978] Nunnally, J.C. *Psychometric Theory*. New York: McGraw-Hill, 1978.

[Procaccina 2002] Procaccina, J.D., Verner, J.M., Software practitioner's perception of project success: a pilot study. *International Journal of Computers, The Internet and Management* 10, (2002), 20-30.

[Procaccina 2005] Procaccina, J.D., Verner, J.M., Shelfer, K.M., & Gefen, D. What do software practitioners really think about project success: an exploratory study. *Journal of Systems and Software*, 78, 2, (2005), 194-203.

[Robey 1996] Robey, D. & Newman, M. Sequential patterns in information systems development: An application of a social process model. *ACM Transactions on Information Systems*, 14, 1(1996), 30-63

[Standish Group International 2004] Standish Group International *CHAOS: Project Failure and Success Research Report* (www.standishgroup.com), Dennis, MA: Standish Group International, Inc. (2004).