International Journal of Information Management 34 (2014) 533–545

Contents lists available at ScienceDirect

International Journal of Information Management

journal homepage: www.elsevier.com/locate/ijinfomgt

Countering user risk in information system development projects

Yu Wen Hung [a], Shih-Chieh Hsu [a,*], Zhi-Yuan Su [b], Hsieh-Hong Huang [c]

[a] Department of Information Management, National Sun Yat-Sen University, No. 70, Lienhai Road, Kaohsiung 80424, Taiwan, ROC
[b] Department of Information Management, Chia-Nan University of Pharmacy and Science, No. 60, Erh-Jen Road, Sec. 1, Jen-Te, Tainan, Taiwan, ROC
[c] Department of Information Science and Management System, National Taitung University, No. 684, Sec. 1, Chunghua Road, Taitung, Taiwan, ROC

ARTICLE INFO

Article history:
Available online 20 March 2014

Keywords:
Information system development (ISD) project
User risk
Risk coping
Risk reduction

ABSTRACT

User related issues have long been broadly discussed in the information system development (ISD) project research area. In this study, we focus on user risk and identify two risk countering approaches to demonstrate how to deal with user risk and its negative impact on ISD projects. We hypothesize that (1) user risk has a negative impact on project performance, (2) users' bond with the project and the development team can help reduce user risk, and (3) developers' task knowledge and vertical coordination can ease the negative impact of user risk and increase project performance. A quantitative approach with survey data collected from 240 practitioners confirmed our hypotheses. In addition, we interviewed seven developers and three user representatives to complete our understanding of this issue. Implications for academia and practitioners are discussed at the end of this paper. Suggestions for future research directions are also provided.

© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

User related issues have long been discussed by different research streams in the information system development (ISD) project research area. For example, user participation research indicates that users should participate in the ISD process to prevent the developed system from deviating from users' actual needs (He & King, 2008; Ives & Olson, 1984). Control based studies have pointed out that users can serve as controllers to ensure the performance of the project and the quality of the developed system (Kirsch, Sambamurthy, Ko, & Purvis, 2002). By jointly reviewing the developed system, the impact of uncertainty can be eased (Hsu, Chan, Liu, & Chen, 2008). Agile software development research, which is growing in popularity and importance in practical and academic circles, emphasizes the importance of individuals, interactions, and user collaboration in software projects (Lee & Xia, 2010). Thus, users not only play an important role but also cannot be absent from the process of software development (Misra, Kumar, & Kumar, 2009; Rumpe & Schröder, 2002). Moreover, previous studies (e.g., Lin & Shao, 2000; Wagner & Piccoli, 2007) concluded that granting users the right to engage not only increases the quality of requirements determination but also raises the users' level of commitment and acceptance. However, it is noticeable that conclusions made by these studies rely on one basic assumption: users are willing to cooperate when needed. In fact, a lack of user support is often cited as one risk factor which amplifies the variation during the development process (Addison, 2003; Wallace, Keil, & Rai, 2004a, 2004b). Thus, in this paper, user risk reflects an unwillingness to cooperate, resistance to change, lack of commitment to the project, conflict between users, and negative attitudes toward the project. Various reasons may induce users not to cooperate with developers during the development process. Uncooperative users may even resist changes brought by the new system (Chatzoglou & Diamantidis, 2009; Kim & Kankanhalli, 2009; Lapointe & Rivard, 2005). Consequently, insufficient support from users makes project planning and control difficult and reduces the probability that the project goal can be achieved effectively and efficiently (Wallace et al., 2004a, 2004b). This suggests the need to consider user-related risks seriously and explore potential approaches to counter them. Therefore, this study aims to explore the means by which user risk and its negative impact on project performance can be eased.

To answer this question, we proposed a model to demonstrate how different approaches can be used to counter user risk through incorporating the reduction and coping concept in uncertainty and risk management literature (Field, Ritzman, Safizadeh, & Downing, 2006; Hsu, Lin, Cheng, & Linden, 2012). While risk reduction refers to ways to bring down the level of risk directly so as to minimize its impact on performance, risk coping refers to attempts to reduce the negative effects of risk without addressing the risk directly. In this study, we proposed that, *first*, users' bonds with

* Corresponding author. Tel.: +886 7 5252000x4732.
E-mail addresses: d004020004@student.nsysu.edu.tw (Y.W. Hung), jackshsu@mis.nsysu.edu.tw (S.-C. Hsu), szj974@gmail.com (Z.-Y. Su), kory@nttu.edu.tw (H.-H. Huang).

http://dx.doi.org/10.1016/j.ijinfomgt.2014.02.003
0268-4012/© 2014 Elsevier Ltd. All rights reserved.

the project and the development team can help reduce the level of risk. Users are more willing to engage in the development process when they are bonded with the project strongly. *Second*, we proposed the moderating role of developers' business knowledge and vertical coordination on the relationship between user risk and project performance. The concept of vertical coordination can serve as actions undertaken through project managers, functional managers, or even higher level steering committees. The moderating effects highlight the important concept that the negative impact of user risk on project performance can be suppressed when these two conditions exist. Furthermore, in order to complete our understanding of this issue, we adopted a two-phase approach. In the first phase, we adopted survey data to quantitatively test the relationship between user risk and project performance, the effects of users' bonds with the project and the development team on user risk, and the impacts of vertical coordination and developers' task knowledge on project performance. In the second phase, we used qualitative data from interviews with user representatives and project developers to have further insights.

By determining how user risk and its negative impact on project performance can be reduced, this study contributes to information system development research by highlighting the ways to strengthen users' bond with developers and project teams to ensure that user risk and its negative impact can be eased. The remainder of the paper proceeds as follows. We first review the impact of user risk on project performance and develop the concept of the users' bond with the project and the development team. Based on risk reduction and coping concepts, we then hypothesize approaches to reduce user risk and mitigate its negative impact on project performance. In Section 3, methods to collect required data to examine our hypotheses are introduced. Section 4 presents the data analysis result and discussion, accordingly. Lastly, conclusions are made and suggestions are provided.

2. Literature review and hypotheses development

2.1. User risk and project performance

Wallace et al. (2004a, 2004b) defined software project risk as *a set of factors or conditions that can pose a serious threat to the successful completion of a software project* and identified six risk factors, including team risk, organizational environment risk, requirements risk, planning and control risk, user risk, and complexity risk. Furthermore, Alhawari, Karadsheh, Nehari Talet, and Mansour (2012) noted that risk is a barrier to success and may lead to a negative impact on the project goal. To avoid project failure, managers should identify and control those risk factors which may lead to cost and schedule overruns, unmet user requirements, and the inability to deliver business value. Among those identified risk factors, user risk drew our attention. User risk reflects an unwillingness to cooperate, resistance to change, lack of commitment to the project, conflict between users, and negative attitudes toward the project. User risk is a critical problem because users should be involved in the ISD process. For example, from knowledge integration perspective, Grant (1996) pointed out that knowledge resides within the individuals in an organization and certain integration is needed to generate effective cross-domain solution. In ISD, since developers have technical knowledge for carrying out the final system and users possess business knowledge to guide the direction of development, effective knowledge integration is needed for effective system development (Mitchell, 2006). However, boundaries between users and developers may limit the effectiveness of knowledge integration between these two parties (Carlile, 2002, 2004). Empirical studies indicated that user risk may lead to lower system quality (Barki, Rivard, & Talbot, 2001; Choe, 1998; Hunton & Price, 1997; Hwang & Thorn, 1999), and unwillingness to use the system (Baroudi, Olson, & Ives, 1986; Choe, 1996, 1998; Gallivan & Keil, 2003; Guimaraes & Igbaria, 1997; Kim & Lee, 1986; Mak, Schmitt, & Lyytinen, 1997). Wallace et al. (2004a, 2004b) further argued that, as a component of social subsystem risk, user risk leads to technical subsystem risk. Through technical subsystem risk and project management risk, the final process and product performance are undermined. On the other hand, the agile software development approach is getting popular recently. It is noticeable that users cannot be excluded from the development process, given the nature of some agile approaches (Misra et al., 2009). Specifically, users need to collaborate with developers intensively during the development process to verify the developed function for every short period. If cooperative support from users is lacking, the agile approach must not be adopted or the outcome is doomed to fail. Therefore, project managers should minimize user risk so as to reduce the variation during the ISD process, which then leads to better project outcomes.

Project management literature has classified project outcomes into two types: product and process (Nidumolu, 1996; Wallace & Keil, 2004). Product performance is defined as the extent to which the designed system contains adequate design or can reflect users' needs (Wallace et al., 2004a, 2004b). To have high product performance, the project team should possess mature analyzing knowledge and methodological tools to capture users' actual requirements (Faraj & Sproull, 2000; Nidumolu, 1995). However, in addition to the availability of required analyzing knowledge and methodological tools, users play an important role in helping clarifying the actual requirements. They should engage in the early stages to verify the design or join the review meeting in the later stages to ensure that the quality of the developed system can really meet their needs (Hsu, Lin, Zheng, & Hung, 2012). One of the most direct consequences of lack of support from users is that actual user requirements are difficult to obtain and, therefore, the developed system cannot support the operational needs of business.

H1a. User risk is negatively associated with product performance.

The lack of user support may also lead to low quality process performance. Process performance refers to the extent to which a project team achieves predefined project goals within budget and on schedule (Schwalbe, 2002). Many projects cannot adhere to predefined schedules or budgets because development teams are unable to determine the actual requirements or fail to identify serious fatal designs in the early stages. It is not uncommon that a flawed and inappropriate system is first presented to users in the late stages of system development. The cost for reworking flaws found in the later stages is much higher than for those found in the early stages (Boehm & Turner, 2003). The most direct consequence is a delayed schedule or budget overrun. Therefore, flaws should be identified and corrected as early as possible. Users should serve as a control tool to monitor system development and identify potential flaws from the business perspective (Kirsch et al., 2002). The project is, therefore, able to adhere to the predefined schedule and budget. Therefore, we hypothesize that:

H1b. User risk is negatively associated with process performance.

Although user risk may cause variation in the development process and undermine final performance, it is not totally uncontrollable. In this study, we propose that the impact of user risk on project outcomes can be managed through two ways: reducing the level of risk (reduction) and mitigating its impact on performance (coping). These two concepts were proposed by Field et al. (2006) to address the mitigation of the negative impacts resulted from operational uncertainty in service area. They argued that uncertainty-countering approaches should be classified into reduction and coping two types. *Reduction* refers to ways to bring down the level of uncertainty directly so as to minimize its impact

on performance whereas *coping* refers to attempts to reduce the negative effects of uncertainty without addressing the uncertainty directly. Hsu, Lin, Cheng, et al. (2012) and Hsu, Lin, Zheng, et al. (2012) adopted these two concepts to illustrate the possible ways to deal with the problem of incorrect requirements in information system development project. They further extended this research stream by showing that (1) the direct effects (from reduction on risk) represents that those approaches can reduce the level of risk directly and (2) the moderating effects (on the relationship between risk and performance) reflects the idea that "coping negative impacts resulted from risk without addressing on risk directly". In this study, we followed this research stream and examine approaches that can be used to counter user risk from both coping and reduction perspectives.

2.2. Risk coping: developers' business knowledge and vertical coordination

As indicated above, risk coping emphasizes easing the impact of risk on performance without making attempts to reduce risk itself (Field et al., 2006). This implies that the project team can reduce the negative impact resulting from user risk when it does take place. In this study, we proposed that the negative impact of user risk may be eased under two conditions: when developers possess sufficient business knowledge and when developers are able to communicate with managers through vertical channels.

An information system development project is knowledge intensive and the final outcome is determined by the extent to which required knowledge resources are available. In addition to developers' technical knowledge, business knowledge is indispensable (Ko, Kirsch, & King, 2005). It is well recognized that the major benefit of having users in the development process is to increase the availability of business knowledge (Tesch, Sobol, Klein, & Jiang, 2009). However, it is noticeable that developers can also be the source of business knowledge. Developers' business knowledge refers to the extent to which developers understand the business operation and application domain of new system. When developers are more familiar with the organizational culture and management style, the negative impacts resulting from uncooperative users can be reduced. Developers' business knowledge not only allows developers to communicate with users more effectively but also allows the project team to carry out an appropriate design with reduced levels of support from users. When adequate requirements are captured, the chance that an appropriate system can be developed consequently is high. In addition, without the need for rework, the project team can better achieve the predefined goal within budget and on schedule. Therefore, we hypothesize that:

H2a. Developers' business knowledge is positively associated with project process performance.

H2b. Developers' business knowledge is positively associated with project product performance.

As indicated above, system development requires both business and technical domain knowledge in the process of transforming business requirements into system design. This implies high knowledge interdependencies among various stakeholders. In order to effectively manage interdependences, coordination between sources of domain knowledge is needed (Andres & Zmud, 2002; Thompson, 1967; Ven, Delbecq, & Koenig, 1976). Nidumolu (1996) stated two major types of coordination between different knowledge sources: horizontal and vertical coordination. By viewing users and developers as major sources of business and technical knowledge, horizontal coordination represents direct interaction activities between these two parties through mutual adjustments and communications, via personal or group means. On the other hand, vertical coordination can be viewed as actions undertaken through vertical means. In most organizations, vertical coordination is through project managers, functional managers, or even higher level steering committees.

Coordination through higher level managers helps reduce project uncertainty through clarifying mission and scope (Parolia, Goodman, Li, & Jiang, 2007). With a clear goal and scope, reasonable planning and control can then be done and process performance tends to be high. Vertical coordination also can help the project team to obtain required resources. With adequate resources (e.g., advanced hardware and technical support), the developed system tends to be of higher quality. Vertical coordination also serves as one type of control mechanism. It provides a means by which the project team can receive feedback about project performance from functional or higher management (Ouchi & Maguire, 1975). With this feedback, required adjustments can be made to get the deviated development work back on the right track. Prior empirical results indicated that vertical coordination can benefit project performance through reducing residual performance risk (Nidumolu, 1996). Therefore, we hypothesize that:

H3a. Vertical coordination is positively associated with project process performance.

H3b. Vertical coordination is positively associated with project product performance.

One major benefit of having users in the development process is to clarify the actual functions which are needed to support business operations. Although a lack of user support reduces the opportunities to capture the real requirements, its impact can be minimized when developers themselves are able to clarify the actual needs. This relies on the business knowledge that developers possess. We argue that, even though support from the user side is insufficient, developers can still design and develop a system that fits business needs if they know business operations processes well, are able to obtain resources, and can remove barriers. Consequently, higher project performance can still be achieved. From the product perspective, without knowledge support from users, better system design can still be derived if developers themselves have sufficient business knowledge. On the other hand, from the process perspective, the project team can still execute and control the project well, even when resources from users are low, if developers themselves possess those resources. Hence, we hypothesize that developers' business knowledge can suppress the negative impact of user risk on project performance. Therefore, we hypothesize that:

H4a. The impact of user risk on project process performance is less when developers have high business knowledge.

H4b. The impact of user risk on project product performance is less when developers have high business knowledge.

We proposed that the project team or manager should adopt vertical coordination to reduce the unwanted effect arising from user risk. For example, when low motivated users are selected as representatives, direct communication between the project and functional managers in the steering committee meeting can help clarify the goal, scope and requirements of the target system. Having higher management involved also allows political barriers (or conflicts between users and developers) to be resolved. Functional managers can also act to increase the motivation of current representatives or even simply replace current representatives with more suitable ones. As an outcome, project execution can be more efficient and the developed system can better fit users' needs. Furthermore, the project team can also obtain slack resources through interacting with higher management directly (Galbraith, 1977; Thompson, 1967). Those resources can be used to cope with

variation resulting from resistant users. Therefore, higher performance can be expected.

H5a. The impact of user risk on project process performance is less when a vertical coordination mechanism is present.

H5b. The impact of user risk on project product performance is less when a vertical coordination mechanism is present.

2.3. Risk reduction: users' bond with the project and the development team

As has been noted above, vertical coordination and developers' task knowledge can lower the negative impacts of user risk on project performance; however, the negative impacts arising from user risk are not likely to be totally eliminated. This suggests a need to minimize risk itself to avoid any unwanted consequences. We argue that users do not commit to the project or dedicate to ISD when they are not strongly bonded with the project team. Specifically, low support is received from users because they don't want to contribute (relational bond), they don't know how to contribute (cognitive bond), or there is no chance for them to contribute (structural bond). Since users and developers serve as two major stakeholders in the ISD project, this study attempts to examine whether users are more willing to contribute when they are bonded with the project and the development team. We view the users' structural bond as the interaction pattern between users and developers, the users' relational bond as users' emotional and affective connection with developers, and the users' cognitive bond as the extent to which users can understand the information system development process. Thus, the link between each bond and user risk is hypothesized.

Users' structural bond with the project represents the extent to which users have sufficient opportunity to interact with developers. This can be represented by whether a formal participation mechanism exists which allows users to engage in the development process. A formal participation mechanism provides opportunities which allow developers to interact with users. For example, users may be invited to the project planning and review meeting. They are informed of the progress of development work and requested to evaluate and approve the developed system. We expect that, through joining the formal meeting, users are more familiar with system development. With this opportunity, they can contribute their knowledge as well as validate the adequateness of the developed system. Participative decision making theory based studies have shown that users are more likely to stick with the project when they engage in the development process formally (Ives & Olson, 1984). User participation literature also indicates that users are more psychologically inclined to stick with the project and the developed system if they are included in the formal decision making process (Barki & Hartwick, 1989; He & King, 2008; Kappelman & McLean, 1991; Lin & Shao, 2000). As a consequence, lower level resistance or more collaborative users are then expected. Therefore, we hypothesize that:

H6. Users' structural bond with the project and the development team is negatively associated with user risk.

Users' relational bond with the project and the development team refers to the extent to which users possess a strong relationship with developers of the target project. We hypothesize a negative effect of users' relational bond on user risk based on two major reasons. First, users' relational bond is the basis of positive behaviors within a relationship. With strong trust, actors are more likely to dedicate themselves to collaboration (Saunders & Clark, 1992). ISD studies also indicated that relationships between two parties promote knowledge sharing, knowledge integration, and knowledge exchange (Chiu, Hsu, & Wang, 2006; Tiwana & Mclean, 2005; van den Hooff & de Winter, 2011; van den Hooff & Huysman, 2009; Wasko & Faraj, 2005). Second, a pre-existing relationship helps ease any negative process of interaction. For example, a relationship between users and developers has been shown to effectively reduce conflicts (Jiang, Klein, & Chen, 2006). Therefore, we hypothesize that:

H7. Users' relational bond with the project and the development team is negatively associated with user risk.

Users' cognitive bond represents the extent to which users have sufficient understanding of the nature of the ISD project. Users' cognitive bond is also argued to be able to reduce user risk. First, user's experience with and knowledge of IT are important factors of IS success (Sabherwal, Jeyaraj, & Chowa, 2006), and a lack of ISD knowledge blocks users from participating in the development process effectively (Saleem, 1996). For example, users will not be fully engaged in the meeting when they are unable to understand the content that developers express or when they are unable to express their own thinking using technical language. Users are more likely and more willing to contribute knowledge when they clearly understand what to contribute. In addition, communication can be more effective and the exchange of knowledge can be accelerated when shared understanding does exist (Ko et al., 2005; Wasko & Faraj, 2005). A shared understanding is founded on users' IT knowledge and developers' business knowledge (Tesch et al., 2009). Second, with sufficient IT knowledge, users can better predict the development process and, hence, monitor the development performance (Kirsch et al., 2002). Therefore, only limited extra learning loading is required for users to get involved in the development process. As an outcome, high willingness to engage is expected. Therefore, we hypothesize that:

H8. Users' cognitive bond with the project and the development team is negatively associated with user risk.

Based on prior literature and arguments, we proposed a theoretical model presented in Fig. 1 which summarizes the proposed hypotheses as well as the nature of each relationship.

3. Research method

We adopted a two-phase approach beginning with a survey of project developers. The second phase included semi-structured interviews with user representatives and developers to provide a richer and more complete understanding of the target issue and to validate the proposed theoretical model (Kaplan & Duchon, 1988; Mingers, 2001; Venkatesh, Brown, & Bala, 2013). It is noticeable that we used a survey approach to collect data from developers to examine the proposed hypotheses listed above. Opinions from developers only limited the explanation power of our conclusion since opinions from users are also needed to capture their bonds with the project (team) precisely. Therefore, the major reason for adopting the qualitative research method is to complement and comprehend the result that we found through the quantitative approach. In addition, the interview provides a chance for us to have a deeper understanding of the field while the quantitative approach allows us to ensure the generalizability with large, varied samples. While this study can be considered as a mixed-method type of research which aims at providing more abundant insights to readers, we cannot ignore the qualitative nature of this study, given that it is a field study which contains different approaches to collect opinions from individuals concerning certain phenomena.

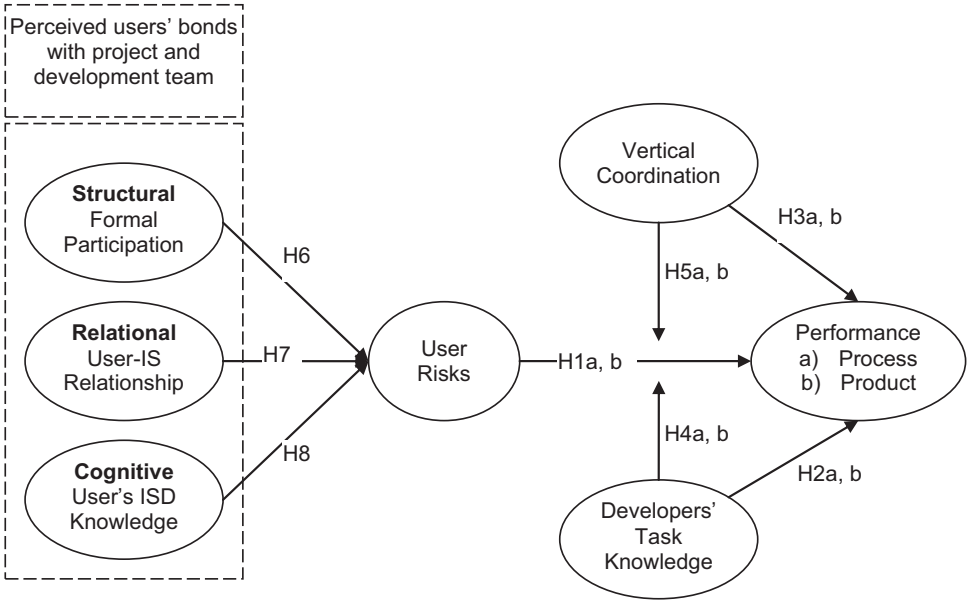


Fig. 1. Research model.

3.1. Phase one: survey

3.1.1. Sampling

Based on our research purpose, individuals familiar with an IS project in terms of process and outcome characteristics were preferred in this study. For this reason, the target sample was IS practitioners with experience of involvement in ISD project teams in the Top 1000 companies in Taiwan (CommonWealth, 2009). Given that users' bonds exist between team members and users exclusively, the key respondents of this study were team members rather than other stakeholders. A survey package was sent to the head of the MIS department in each company. The survey asked respondents to answer all questions with reference to their most recently completed project. The department head was asked to pass on the survey questionnaire to the most senior member in the department who had participated in the most recently completed ISD project. Senior members were asked to rate the team's overall interaction with users. Survey packages were sent out through regular mail in early March, 2011, and a reminder mail was then sent to those who did not return the survey after one month. A total of 253 surveys were returned. Among these, 13 returned instruments were unusable due to omissions, which yielded a final response rate of 24%. Hence, our data represents users' bonds with projects and teams (as perceived by developers), and the final project performance for 240 project teams. The demographic information about the sample is provided in Table 1.

3.1.2. Constructs and measurement

Process performance refers to the success of the development process itself. It was measured by 5 items obtained from existing scales (Guinan, Cooprider, & Faraj, 1998; Henderson & Lee, 1992; Jones & Harrison, 1996) that assessed subjects' perceptions of

Table 1
Demographic data – project ($N = 240$).

Measure	Categories	#	%	Measure	Categories	#	%
Project size	Less than 10	182	75.8	Project duration (month)	Less than 5	81	33.8
	11–20	36	15.0		6–10	53	22.1
	21–30	9	3.8		11–15	48	20.0
	31–40	2	0.8		16–20	17	7.1
	41–50	4	1.7		21–25	18	7.5
	More than 50	2	0.8		26–30	1	0.4
	Missing	5	2.1		More than 30	11	4.6
					Missing	11	4.6
Gender	Male	184	76.7	Working experience (year)	1–5	51	21.3
	Female	55	22.9		6–10	66	27.5
	Missing	1	0.4		11–15	41	17.1
Age	21–25	7	2.9		16–20	35	14.6
	26–30	40	16.7		More than 20	43	17.9
	31–35	43	17.9		Missing	4	1.7
	36–40	56	23.3	Position	Programmer	48	20.0
	41–45	27	11.3		System Analyst	29	12.1
	46–50	35	14.6		Project Leader	54	22.5
	More than 50	25	10.4		IS Manager	56	23.3
	Missing	7	2.9		Network Engineering	9	3.8
Educational background	College	13	5.4		Database Administrator	7	2.9
	Bachelor	100	41.7		Testing member	5	2.1
	Master	114	47.5		Maintaining members	10	4.2
	Doctoral	12	5.0		Others	19	7.9
	Missing	1	0.4		Missing	3	1.3

Table 2
Measurement validity and reliability.

Constructs	Items	Factor loading	*t*-Value	ITC
Perceived users' structural bond	Users and developers meet on a routine basis	0.60	5.25	0.45
User participation (α = 0.78; CR = 0.84; AVE = 0.51)	Users are kept informed about project process and project related problems	0.86	18.27	0.54
	Users formally approved the work done by the project team	0.66	6.56	0.52
	Users are participating in joint planning for details of project approach, timing, and success criteria	0.72	5.85	0.69
	Users are engaging in discussion that links client co-production to key client motivators and training concerning key performance barriers	0.70	5.97	0.59
Perceived users' relational bond	There is mutual respect between developers and users	0.83	25.69	0.65
User-IS relationship	There is mutual trust between developers and users	0.86	32.39	0.67
(α = 0.80; CR = 0.87; AVE = 0.62)	There is personal friendship between developers and users	0.75	11.81	0.59
	There is high reciprocity among developers and users	0.71	13.64	0.52
Perceived users' cognitive bond	Users are familiar with ISD task and life cycle stages	0.85	18.75	0.64
Users' ISD knowledge (α = 0.78; CR = 0.89; AVE = 0.81)	Users are not familiar with data processing.	0.95	87.10	0.64
Developer's task knowledge	Developers have overall knowledge of organizational operations	0.82	27.66	0.67
(α = 0.82; CR = 0.88; AVE = 0.65)	Developers have in-depth knowledge of user departments	0.83	25.51	0.68
	Developers have overall administrative experience and skill	0.78	20.82	0.65
	Developers expertise in the specific application area of this system	0.79	28.34	0.58
Vertical coordination	An individual formally responsible for coordination (e.g., a project manager)	0.79	18.84	0.46
(α = 0.71; CR = 0.83; AVE = 0.63)	An external committee or group responsible for coordination (e.g., a steering committee)	0.70	11.23	0.51
	Referral of coordination decision to senior DP or senior user management	0.87	36.26	0.60
User risk	Users resistant to change	0.78	23.81	0.71
(α = 0.90; CR = 0.93; AVE = 0.72)	Conflicts between users	0.77	23.84	0.70
	Users with negative attitudes toward the project	0.90	53.74	0.82
	Users not committed to the project	0.89	55.22	0.81
	Lack of cooperation from users	0.89	50.00	0.76
Process performance	Team is able to achieve project predefined goal	0.82	33.70	0.68
(α = 0.88; CR = 0.91; AVE = 0.67)	Team is able to complete a series of predefined tasks	0.80	23.44	0.71
	Team is able to complete tasks on schedule	0.85	33.83	0.77
	Team is able to complete tasks within budget	0.81	26.07	0.72
	Team is able to effectively perform set tasks	0.82	31.18	0.69
Product performance	The application is easy to maintain.	0.79	22.53	0.64
(α = 0.84; CR = 0.90; AVE = 0.68)	The users perceive that the system meets intended functional requirements	0.84	25.54	0.70
	The system meets user expectations with respect to response time	0.85	31.42	0.72
	The overall quality of the developed application is high	0.82	29.95	0.67

Note: α: Cronbach's alpha; CR: composite reliability; AVE: average variance extracted; ITC: item-total correlation.

project performance in terms of schedule, budget, and work. *Product performance* refers to the quality level of the developed system and was measured by 4 items adapted from Wallace et al. (2004a, 2004b). *User risk*, users' low willingness to cooperate during a development effort, were measured by 5 items from (Wallace et al., 2004a, 2004b). *Vertical coordination* is the extent to which coordination between users and information system staff is undertaken through vertical means (Nidumolu, 1995). It was measured by 3 items adapted from Nidumolu (1995). For *developers' task knowledge*, a total of 4 items adapted from Barki et al. (2001) were used to capture developers' knowledge of the new application areas. For users' bond with project and development team, we used 5 items from Barki and Hartwick (1994) to capture *perceived users' structural bond*, the extent to which a formal participation mechanism is available for users to engage in the system development process; 5 items from Tiwana and Mclean (2005) to measure *perceived users' relational bond*, the level of mutual trust, respect, reciprocity and closeness of the relationship between users and developers during the ISD project (Kale, Singh, & Perpmutter, 2000; Tiwana & Mclean, 2005); and 2 items from Barki et al. (2001) to measure *perceived users' cognitive bond*, users' overall knowledge/expertise in IS development methods and processes.

All items were adapted from previous research and rated on a five-point Likert scale anchored at 1 = "strongly disagree" and 5 = "strongly agree" to represent whether each individual perceived the item to be true during the most *recently* completed project. To ensure content validity, the instrument was pre-tested by two professors with experience in ISD research, 5 doctoral students, and 3 project professionals. In addition, 5 doctoral students performed a card-sorting task. They were asked to categorize all items into a number of categories based on the perceived semantic meanings. Several minor corrections and clarifications to the instrument were made based on professional feedback and a pre-test to ensure content validity.

In this study, partial least squares (PLS) analysis was used to test item reliability, convergent validity, and discriminant validity (Hulland, 1999). Tables 2 and 3 provide factor loadings of the items for the underlying dimensions. As shown in Table 4, most factor loadings are greater than the recommended 0.7 except for 2 items in perceived users' structural bond. These two items were kept since they were not lower than the minimum cut-off value (0.5) (Hair, Black, Babin, Anderson, & Tatham, 2006). Furthermore, the item-total correlations (ITC) are higher than the recommended cut-off value (0.3). Convergent validity is also ensured (composite reliability > 0.7 and AVE > 0.5). As Tables 2 and 4 show, discriminant validity is also ensured assured (correlations < 0.8 and the square root of AVE > correlation coefficient) (Chin, 1998).

In addition, the Harman's single factor test was implemented to ensure that there was no significant common method effect on the predefined causal relationship. The exploratory factor analysis shows that a total of 8 factors can be derived, the first factor explaining 27% of the variance. Therefore, the data do not suffer from common method variance. The descriptive information

Table 3
Factor analysis results.

	Structural bond	Relational bond	Cognitive bond	Developer's task knowledge	Vertical coordination	User risk	Process performance	Product performance
SB1	**0.60****	0.14	−0.10	0.16	0.49	−0.11	0.25	0.24
SB2	**0.86****	0.26	0.08	0.25	0.38	−0.30	0.32	0.39
SB3	**0.66****	0.19	0.01	0.36	0.27	−0.10	0.30	0.35
SB4	**0.72****	0.26	−0.05	0.35	0.37	−0.08	0.24	0.32
SB5	**0.70****	0.31	0.06	0.37	0.29	−0.14	0.25	0.36
RB1	0.31	**0.83****	0.09	0.29	0.16	−0.29	0.35	0.50
RB2	0.22	**0.86****	0.15	0.30	0.17	−0.33	0.30	0.44
RB3	0.21	**0.75****	0.04	0.30	0.15	−0.22	0.29	0.36
RB4	0.28	**0.71****	0.09	0.23	0.16	−0.25	0.26	0.42
CB1	0.13	0.01	**0.85****	0.02	−0.02	−0.13	0.09	0.06
CB2	0.00	0.14	**0.95****	0.03	−0.05	−0.44	0.02	0.13
DTK1	0.39	0.26	0.01	**0.82****	0.20	−0.17	0.37	0.35
DTK2	0.29	0.35	0.01	**0.83****	0.17	−0.18	0.35	0.37
DTK3	0.28	0.20	0.05	**0.78****	0.12	−0.08	0.37	0.23
DTK4	0.26	0.32	0.04	**0.79****	0.13	−0.12	0.42	0.40
VC1	0.33	0.14	0.00	0.13	**0.79****	−0.08	0.29	0.25
VC2	0.41	0.13	−0.11	0.15	**0.70****	0.01	0.21	0.13
VC3	0.44	0.20	−0.04	0.18	**0.87****	−0.05	0.32	0.28
UR1	−0.11	−0.23	−0.32	−0.09	−0.03	**0.78****	−0.17	−0.16
UR2	−0.12	−0.21	−0.39	−0.10	0.05	**0.77****	−0.19	−0.16
UR3	−0.27	−0.31	−0.34	−0.18	−0.05	**0.90****	−0.30	−0.29
UR4	−0.25	−0.27	−0.36	−0.12	−0.04	**0.89****	−0.15	−0.23
UR5	−0.29	−0.40	−0.33	−0.21	−0.15	**0.89****	−0.26	−0.40
PCP1	0.41	0.39	0.09	0.41	0.38	−0.26	**0.82****	0.58
PCP2	0.23	0.26	0.04	0.33	0.17	−0.18	**0.80****	0.36
PCP3	0.25	0.27	−0.01	0.36	0.27	−0.16	**0.85****	0.41
PCP4	0.23	0.30	−0.04	0.35	0.26	−0.18	**0.81****	0.40
PCP5	0.39	0.32	0.06	0.45	0.34	−0.25	**0.82****	0.50
PDP1	0.33	0.42	0.06	0.35	0.21	−0.24	0.43	**0.79****
PDP2	0.36	0.45	0.14	0.35	0.25	−0.30	0.51	**0.84****
PDP3	0.38	0.47	0.11	0.37	0.23	−0.20	0.42	**0.85****
PDP4	0.45	0.47	0.10	0.35	0.26	−0.27	0.48	**0.82****

** $p < 0.01$.

and correlation coefficients of all used variables are shown in Table 4. Moreover, a comparison between the groups of respondents (project duration less than 5 months and project duration longer than 11 months) was conducted to assess the potential for sampling bias. In this test, 81 projects were less than 5 months in duration, and 95 projects were longer than 11 months. The results show no significant differences in the responses of the two groups. Thus, no evidence of sampling bias was found.

3.1.3. Data analysis

We used PLSgraphic 3.0 with bootstrapping as a resampling technique (500 random samples) to test the structural model and the significance level of the paths. Path coefficients, their significance levels, and the R^2 were used jointly to evaluate the model (Chin, 1998). As shown in Fig. 2, user risk is found to have a significant negative impact on both product performance and process performance (H1). On the other hand, risk coping had a positive impact on both project outcomes including positive relationships between vertical coordination and project outcomes (H2) and between task knowledge and project outcomes (H3). Furthermore, task knowledge moderated the effect of user risk on project performance (H4); and vertical coordination partially moderated the effect of user risk on project performance (H5) since the interaction terms are only significant at a level of 0.1. User risk, vertical coordination, task knowledge, and the interactions between risk and vertical coordination and between risk and task knowledge totally explain 31.9% of the variance of product performance and 36.1% of the variance of process performance. Moreover, user risk can be reduced by users' bonds with the project and the development team including structural (H6), rational (H7) and cognitive (H8) bond. In addition, three types of bond totally explain 27.9% of the variance of user risk.

To further understand the moderating effects of vertical coordination and task knowledge on the relationships between user risk and the two types of project outcomes, we created four diagrams, as shown in Figs. 3a–d. The sample is separated into three groups based on the level of task knowledge (low, medium, and high task knowledge) and vertical coordination (low, medium, and high vertical coordination). Specifically, we used one standard deviation lower and higher than the mean to represent the low and high level.

Table 4
Descriptive analysis and correlation matrix.

Variables	Mean	Std	M3	M4	SB	RB	CB	DTK	VC	UR	PCP	PDP
Structural bond	3.74	0.90	−0.65	0.23	**0.71**							
Relational bond	3.84	0.77	−0.75	1.09	0.32	**0.79**						
Cognitive bond	3.19	0.97	−0.28	0.53	0.05	0.13	**0.90**					
Developers' task knowledge	3.74	0.74	−0.44	0.23	0.38	0.36	0.03	**0.81**				
Vertical coordination	3.85	0.97	−0.77	−0.39	0.49	0.20	−0.02	0.19	**0.79**			
User risk	2.79	1.02	0.23	−0.71	−0.26	−0.35	−0.41	−0.18	−0.07	**0.85**		
Process performance	3.81	0.80	−0.72	0.83	0.37	0.38	0.04	0.47	0.35	−0.26	**0.82**	
Product performance	3.86	0.70	−0.70	1.16	0.46	0.55	0.12	0.43	0.29	−0.30	0.56	**0.82**

Notes: Square root of average variance extracted (AVE) appears along diagonal. M3: skewness; M4: kurtosis.

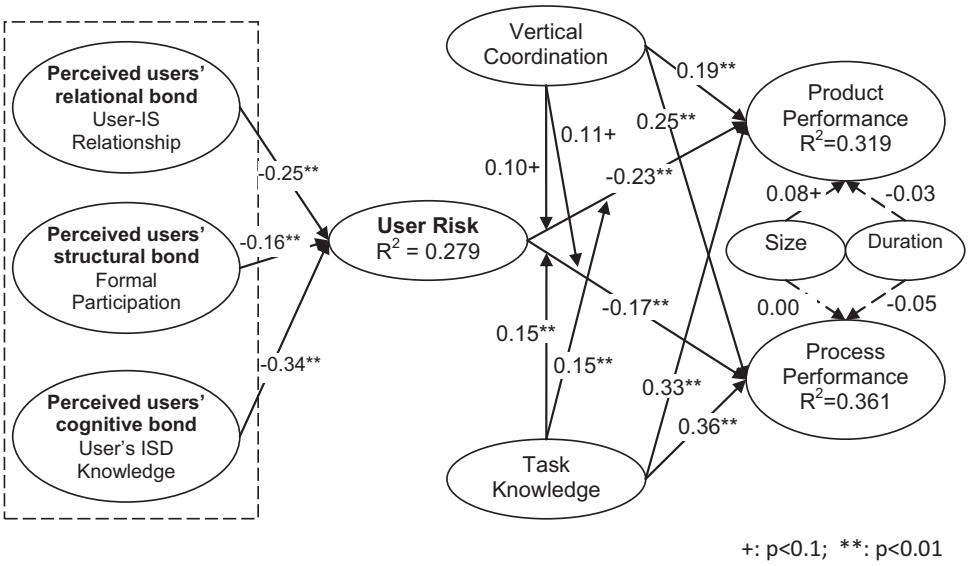


Fig. 2. Path analysis.

These four diagrams show a similar pattern with only slight differences. Take task knowledge, for example: two major points can be observed. First, project performance is high when user risk is low, and performance decreases gradually as the risk level increases. Second, for project teams with high task knowledge, process performance remains the same and product performance is slightly lower when the risk level is high. However, for those teams without sufficient task knowledge, both process and product outcomes decreased significantly. A similar pattern can be found when vertical coordination is employed. These four diagrams convey one important message: that both developers' task knowledge and vertical coordination can significantly suppress the unwanted effect arising from user risk and keep project performance at a relatively high level.

3.2. Phase two: post-survey Interview

3.2.1. Interview process and subjects

Although the quantitative results support most of the hypotheses, a semi-structured interview approach was conducted to validate the quantitative results and provide the richer explanations and insights. To align with our research purpose, we interviewed the project manager, programmers and user representatives (individuals from user department and engaged in the development process). User representatives usually are functional managers of department and direct users. Of 10 interviews, 3 were conducted with project managers, 4 with senior/junior programmers, and 3 with functional managers. The interviewee descriptions are shown in Table 5.

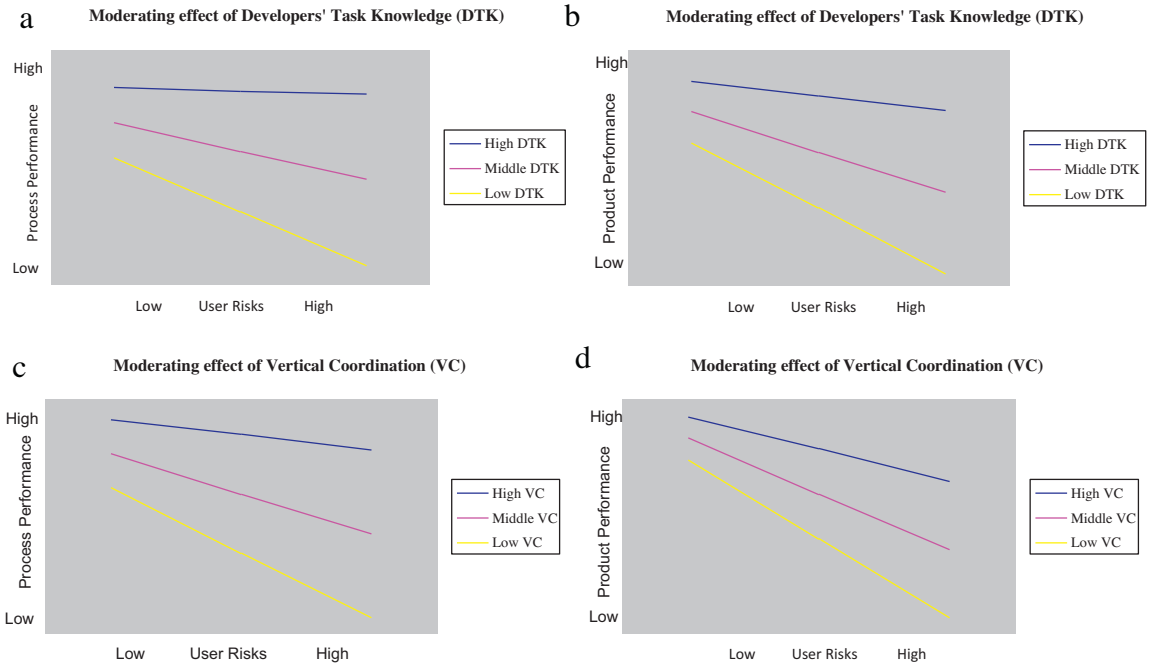


Fig. 3. (a) Moderating effect of DTK (DV: dependent variable). (b) Moderating effect of DTK (DV: product performance). (c) Moderating effect of VC (DV: process performance). (d) Moderating effect of VC (DV: product performance).

Table 5
Interviewee descriptions.

Case	Interviewee	Gender	Department	Position	Role	Tenure (years)
Hospital A	A1	Male	MIS department	Project Manager	Developer	22
	A2	Female	MIS department	Senior programmer	Developer	15
	AU1	Female	Office of medical affairs	Functional manager of department	User representative	14
Hospital B	B1	Male	MIS department	Senior programmer	Developer	9
	B2	Male	MIS department	Project manager	Developer	11
	BU1	Female	Social work department	Functional manager of department	User representative	16
	BU2	Female	Registration of outpatient department	Functional manager of department	User representative	20
Bank C	C1	Male	MIS department	Project manager	Developer	10
	C2	Female	MIS department	Senior programmer	Developer	8
	C3	Male	MIS department	Junior programmer	Developer	1

Specifically, opinions from users and developers were obtained to understand the major factors which cause users to be less willing to be cooperative. On the other hand, developers were asked to provide information about approaches adopted to counter risks caused by users. The duration of the interviews ranged from 30 min to an hour. The interviews were transcribed and then analyzed. Furthermore, all transcripts were sent to the interviewees to confirm the content of the transcripts. The questions asked during the interview included the following:

- Have you ever experienced a situation in which users were unwilling to cooperate with developers during the development process in your organization?
- How would you describe the relationship between user risk and project performance?
- How would you describe those cooperative and uncooperative users, in terms of their bonds with the project and development team?
- Under what conditions can the project team still properly carry out the project when a lack of support from users is observed?
- What mechanisms can ease the effects when user risk does occur?

4. Research findings and discussion

This paper examined the effects of user risk on project performance. Table 6 summarizes the interview findings and corresponding hypotheses. The findings and insights of the constructs proposed in the conceptual model are presented as follows.

4.1. User risk and its impact

The analysis showed that the coefficient between user risk and product performance is slightly higher than the one between user risk and process performance. This implies that low motivated users damage product performance more severely. Follow-up interviews also raised issues relevant to the nature of user risk impact. Overall, all of interviewed developers are confirmed that user risk occurs frequently during the ISD project. All of them have experience working with low cooperative user representatives. The most direct impact of user risk is a delayed schedule or a system with limited responsiveness. Moreover, a programmer mentioned that the priority of the project is downgraded gradually when users are not willing to cooperate, or are even against the project.

> *Most of [the] experiences I had [indicated that] users [had] negative attitudes and low commitment toward [a] project, and those projects ended up [with a] delayed schedule or even [in] failure.* (Interviewee B1)
> *While having more than one project in hand, we usually schedule to complete [the] higher priority project. Besides urgency, the priority is based on whether users can cooperate with us. If the users show their unwillingness to cooperate, we will downgrade the priority of the project. This may lead the project [to fall] behind [in] the schedule or fail.* (Interviewee C1)
> *This situation seldom occurs in my department. We all understand that if we do not work well with ISD team, we may not get the satisfied system.* (Interviewee BU1)

4.2. Risk coping

With respect to the risk coping, quantitative results show that both mechanisms have expected effects. Further, the coefficients and diagrams showed that developers' task knowledge has a stronger moderating effect, compared with vertical coordination. During interviews, the responses also echoed that developers' task knowledge and vertical coordination are helpful for project performance when support from users is low. In general, programmers first attempt to solve problems on their own when user cooperation is low. Furthermore, when junior programmers report low cooperative willingness from user representatives, senior programmers, who sometimes serve as system analysts and coordinators, tend to first clarify the actual needs based on their own experiences.

> *When I cannot get required support from users, I often ask help from my supervisor (senior programmer). In general, he teaches me how [he] does the in-patient process works [sic] and also shows me how to link different modules together to meet the real situation.* (Interviewee C3)
> *Since I have worked in the hospital for more than 10 years, I am pretty familiar with the process of the outpatient department. Usually, I can develop a system without users' full support.* (Interviewee A2)

When programmers cannot solve the problem by their own, vertical coordination is the next step. If experiences do not work, senior programmers may report the issue to project managers who then bring it up to higher level committee meetings. As an outcome, functional managers are forced to resolve this issue.

> *In general, I communicate with the functional manager first to clarify his or her true intention on this project. If he or she [is] really concerned about the project, he or she will assign a user who really knows the business and is willing to spend time with us.* (Interviewee C2)
> *When conflicts occur, I usually talk to the representative of my department to understand why he/she did not work well with project team.* (Interviewee BU2)
> *If user representatives are not cooperative and the functional managers cannot help to resolve the problems, I could report to [the] MIS committee which is the top level of management, but in our culture, we would like to keep harmony at [the] workplace; therefore, this situation does not happen always.* (Interviewee C1)

Table 6
Key findings from interviews.

Issues	Qualitative Findings	Mapped Hypothesis
User risk and its impact	User risk indeed has negative impacts on project performance, including product and process performance.	H1a,b
Risk coping	When developers have strong business knowledge,	H2a
	(1) they do not have to rely on users providing required support and they can ensure the schedule of the project;	
	(2) they can develop a system which is closer to users' needs.	H2b
	When user risks happen during the project,	H3a
	(1) top level management or the steering committee can help ease the conflicts between users and developers and make the project complete on schedule;	
	(2) top level management or the steering committee can help clarify the direction and goal of the project (when support from users is absent) so that the developed system can react to business needs.	H3b
Risk reduction	When users have more chances to participate in the meetings, especially in early stages of the project, they are more likely to cooperate with the developers.	H6
	When users have strong relationships with developers, they are more willing to be involved in the project and make contributions.	H7
	When users have more IT knowledge, they can clarify their actual needs; further, developers can capture their requirements more easily.	H8
Other causes of user risks	Belief Proactive users are willing to contribute to the project and do extra work.	N/A
	Commitment Low user risks are observed when projects are initiated by users since the developed system is directly related to their routine work	N/A

4.3. Risk reduction

With respect to the risk reduction, we proposed that user risk is reduced when users have structural, relational and cognitive bond with developers and project team. The survey findings support that users commit more to the project and are more willing to dedicate to the project when they have a strong bond with the project and the development team. Among the three types of bond, cognitive bond was found to have the strongest effect, relational bond was next, and structural bond had the most minor effect.

During the interview, cognitive bond is highly valued by both users and developers. Users' information technology capabilities allow them to express what they need precisely in the language that developers can understand. The cognitive bond also allows users to jointly solve problems faced by junior programmers who, in general, do not have sufficient business domain knowledge. In addition, many developers agree that users who are familiar with information technologies are also able to appreciate what has been done by the project team. However, one potential side effect of users with a strong cognitive bond is that those users may attempt to control the direction of each project in which they are engaged.

> *Most of [our] users are now familiar with basic IT knowledge. We found that users with more IT knowledge or more experiences with ISD projects can easily clarify their actual needs and [find it] easier to communicate. They are more likely to understand what developers can do.* (Interviewee C3)
> *Without sufficient knowledge about information technology and system development, it is difficult for you to truly engage in the development process. The only thing you can do is tell them (developers) your needs and pray that they can carry out what you want.* (Interviewee AU1)

The importance of relational bonds between developers and users are emphasized by developers and users. A strong relationship increases the users' motivation and commitment. When users have strong relationships with developers, they are more willing to be involved in the project and make contributions.

> *We worked very hard to maintain a smooth relationship with user representatives, especially when the project [was] initiated by [the] IS department. Since the newly developed system contains potential bugs [that] need to be fixed, we rely on users to perform required tests to figure those bugs out [sic]. Without a strong relationship, [we cannot likely expect] users to perform [the extra work].* (Interviewee C2)
> *When users have good relationship with us, they would like to spend more time on meetings and clarifying specification.* (Interviewee B2)
> *We may need them (the MIS department) to develop another system for us in the future. Being cooperative in this project serves as [a] relationship exchange which definitely benefits the future projects.* (Interviewee BU1)
> *Having good relationship [sic] with developers is good for our department because we sometimes can ask developers a favor to do some modifications for us or make our project a priority.* (Interviewee AU1)

Lastly, most of the interviewees indicated that formal interaction mechanisms between user representatives and developers do exist in their organizations. For most ISD projects, users or their representatives are required to attend routine meetings at different stages of the project. Most of the programmers agreed that users or their representativeness should be invited to engage in the project in the very early stage to clarify the goal and to build mutual trust and respect with each other. A lack of this participation in the early stage may lead to negative impacts. However, several junior programmers and senior programmers expressed that it is not necessary to have user representatives in coding stage.

> *In the beginning of the project, we usually ask [the] user representative [to] attend every meeting to clarify and confirm the needs and requirements. After that, user representatives only attend [the] monthly meeting or milestone meeting to track the process and make sure the developed functions work.* (Interviewee A1)
> *I expect my department to participate [in] all the projects related to our works [sic]. We will try our best to work with the developers to make the developed system [a] success. Take a project initiated by IS department as [an] example: my department was not invited to participate in the meeting until the project was almost done. [By] that time I [did] not feel respected and [did] not have [a] strong willingness to cooperate with the project team.* (Interviewee BU2)
> *Based on my experiences, early stage of the project is the most important part. Therefore, the representative of my department attends the meeting at the early stage to clarify our needs and requirements. After that, the representative only attends monthly meeting in the developing stage.* (Interviewee BU1)

Having user representatives in the coding stage is unnecessary and a waste of time. (Interviewee B1)

4.4. Other causes of user risk

There are some other causes of user risk derived from interviews: negative attitude and low commitment. Negative attitude is largely related to personal issues, such as a negative working attitude, fear of change and heavy work loading. Furthermore, the major drivers of low commitment can be classified into two types: the developed system is not directly related to their routine work and high work loading from the users' original functional department.

Our bank was a public organization before transforming to a private [one] back several years ago. Some senior users think that participating in the project is extra work for them. Even [though] they always say YES to you for everything in the meeting, however, only [a] few of them actually take actions to get things done later. (Interviewee C3)

Some senior users have [a] negative attitude toward the new system project because they are afraid of changing [sic] brought from new system such as [a] different working process. Therefore, they are not willing to participate in the project. (Interviewee A2)

In many cases, users are not cooperative with us because the developed systems are not related to their work. This would lead users [to have a] commitment toward [the] project [that is] low. (Interviewee B2)

In our department, the user representatives are selected carefully. The developed system is quite related to their job responsibilities; therefore, they are usually willing to cooperate with the developers (or programmers). (Interviewee BU2)

We have worked with different levels of users from different departments. Compared to administrative users, doctors and nurses [exhibit a] lack of cooperation since they already have heavy work loadings [sic] and are usually busy at their works [sic]; hence, they are not willing to spend extra time on extra works[sic]. (Interviewee A1)

5. Conclusion

The purpose of this study was to understand the impacts of user risk on project performance and propose potential approaches to reduce the risk and cope with its negative impacts. We hypothesized that (1) user risk reduces project performance, (2) users' bonds with the project and the development team can avoid user risk, and (3) developers' task knowledge and a vertical coordination mechanism can increase project performance directly and indirectly through suppressing the negative effect arising from risk. We adopted a survey to verify our hypotheses and utilized semi-structured interviews to serve as a supplement. Survey data collected from 240 developers shows that, except for the moderating effect of vertical coordination which is partial significant at a level of 0.1, the remaining proposed paths are all significant at a level of 0.01. Interviews with seven developers and three user representatives found consistent results and a few additional insights.

5.1. Limitations

Although the findings are encouraging and useful, the present study has certain limitations. First, while previous studies identified a broad set of factors that may affect the project performance, only user risk is considered in this study. For example, Wallace et al. (2004a, 2004b) categorized six major risk factors which may undermine the performance of ISD project and Clarke and O'Connor (2012) further provide a comprehensive set of environmental factors that affect the software development. Future research is encouraged to explore possible means to reduce other risks or ease their impacts.

Second, the results of quantitative data may have been impacted by bias because our sample is comprised of developers only. Users might have different perceptions about the bonds with the project and the development team, so users could have been affected by them differently. Therefore, future studies should collect data from users and developers to avoid possible bias. Third, only one variable was used to represent each bond. Some variables, such as users' identification with the project, were omitted in the design stage to maintain parsimony. Further studies are encouraged to explore more possible antecedents of user risk. Fourth, even though we adopted a survey and interviews to confirm our hypotheses, the data presented are cross-sectional. The development of projects is an ongoing phenomenon. We measured factors of risk coping and reduction at a static point rather than as they were developing, thus losing the time richness of the explanation. An ideal empirical design for testing the proposed model would be longitudinal in order to capture the complex, dynamic interrelationships in all respects and reveal how these factors develop over time in ISD projects. Fifth, only regional data was used to examine the proposed model. Although we believe that user risk is a universal phenomenon and most countering approaches have common effects, data from multiple areas are still recommended to exclude potential biases caused by cultural issues. Last, interview findings are from only seven developers and three users. The sample size might be too small, so the results may not be validated by a larger sample. Future research could conduct more interviews with users and developers to have robust findings.

5.2. Implications

5.2.1. For academia

In this study, we highlight that risks can be eased by using two different approaches: reduction and coping. In particular, we proposed that users having strong bonds with the project and the development team can effectively reduce user risk. On the other hand, when user risk is high, vertical coordination and developers' task knowledge serve as useful tools in easing its impact on project performance. By separating risk countering into two approaches (reduction and coping), two major implications can be obtained.

First, user risk can be directly reduced, as we proposed, when users maintain a strong bond with the project and the development team. The major causes of user risk include low motivation, incapable representatives, and no demand or no chance to engage. Users' relational bond serves as an effective means by which to increase motivation and reduce potential barriers such as conflict. Users' cognitive bond represents representatives' capability to contribute to the project which also indirectly promotes motivation. Users' structural bond can be viewed as the foundation which allows or demands users to engage in the ISD process. In summary, our data successfully showed that users' bonds with the project and the development team can usefully reduce user risk. However, these three types of bonds totally explain less than 30% of the variance of user risk. This implies that some critical antecedents have not been identified. In addition, we conducted the post-survey interviews to confirm that the quantitative results are reliable. Future research should further explore other potential factors that can directly reduce user risk to develop a more comprehensive model.

Second, the moderating effects of vertical coordination and task knowledge imply that the negative impact of risk on performance can be suppressed through coordination with higher levels of management or enhancing the task knowledge of developers. Previous studies considered task knowledge as a critical antecedent

of project performance (Barki et al., 2001), and also illustrated how vertical coordination can reduce project risks (Nidumolu, 1996). In this study, we highlight the effect of task knowledge and vertical coordination on easing the negative impact of user risk. These two approaches serve as useful tools in mitigating problems arising from user risk. As shown in Fig. 3a–d, project performance is high when developers have sufficient task knowledge and when vertical coordination does exist. The combination of the first and second implications means that attempts can be made to reduce risk directly or mitigate its negative impact on performance.

5.2.2. For practitioners

Previous IS research has pointed out the need for users to participate in the ISD process. The popularity of the agile method further highlights the importance of having effective collaboration from users. Drawing on this issue, this study identified several determinants of users' willingness to cooperate with developers in the ISD context. It is important for managers to (1) create formal mechanisms that allow users to engage in the development process, (2) maintain a strong relationship with user representatives, (3) ensure that users have sufficient ISD knowledge, (4) increase users' commitment to the project and the development team, and (5) select user representatives who possess a positive attitude toward the project.

Our findings indicate that users are unwilling to cooperate with developers when they possess a negative attitude toward the system. This finding is consistent with what past user resistance studies have posited that users may oppose the system for various reasons. Therefore, project managers should first clarify the political issues involved in the project. Organizations should (1) exclude possible barriers which drive users to oppose the system, or (2) provide extrinsic or intrinsic motivations to ease the unwanted impact resulting from negative attitudes. Also, there is a need to have users engaged as early as possible not only to build mutual trust, respect and understanding but also to clarify the needs and requirements of the system. Therefore, user representatives are strongly encouraged to attend each meeting in the early stages of the project. However, having users in the development process is not without cost. User participation may also lead to conflict. According to the literature (Boehm & Turner, 2003; Hsu, Lin, Cheng, et al., 2012; Hsu, Lin, Zheng, et al., 2012), users should engage more in the early stages than in the later stages. Specifically, users should work with developers intensively to ensure the quality of requirements elicited in the design stage. However, in the development stage, users should attend the review meetings periodically to ensure the developed system actually meets users' requirements (Hsu, Chen, Jiang, & Klein, 2010).

Since developers interact with user representatives more frequently, the results of this study highlight the importance of selecting the correct user representatives. For example, users having sufficient IT and ISD knowledge can easily understand the ongoing activities and engage in those processes. Therefore, users with abundant IT and ISD knowledge are strongly recommended. Additionally, training should be provided to empower users—especially those with insufficient ISD knowledge—and increase their willingness to engage in the project. On the other hand, there is also a strong need for developers to strengthen their business domain knowledge and learn how to communicate with users. Organizations should offer corresponding training to developers or design certain mechanisms (e.g., job rotation) which allow developers to enhance their business domain knowledge. Furthermore, users having strong relationships with developers are also recommended to serve as qualified representatives. Project managers may provide a candidate list for functional managers' reference based on past experiences or suggestions from members. When user representatives who do not have a relational link with developers are selected, managers may create opportunities for users and developers to know each other better through formal meetings or informal events, such as lunch or parties.

Moreover, when users notice that the developed system is related to their future work, they will have a strong commitment and be more willing to engage in the project. Functional managers could choose the most directly affected end users as representatives to engage in the ISD process. In addition, functional managers could assign proactive users as representatives since they are more willing to spend time and participate in activities other than their regular work.

When a low level of willingness to cooperate is observed, project managers can take different strategies to reduce negative impacts. Two approaches proposed by this study can promote project performance directly and indirectly. First, our study highlights the importance of vertical coordination to the easing of the negative effect of user risk. For example, when user representatives are not willing to cooperate with developers, developers could report to higher management. It is noticeable that the effectiveness of vertical coordination is contingent on the capability of project managers. This highlights the importance of project manager selection and training. In addition to hard skills, such as planning and control, project managers with strong soft skills, such as communication and coordination, are preferred when more coordinating needs are expected. Functional managers may assign different representatives or even adopt approaches to shift the climate in the department. Project managers should also pay attention to enhancing developers' task knowledge when user representatives are not willing to cooperate. Since user representatives contribute to the project through contributing their business domain knowledge, developers with strong business domain knowledge can ease the negative impact arising from users' unwillingness to participate.

References

Addison, T. (2003). E-commerce project development risks: Evidence from a Delphi survey. *International Journal of Information Management*, *23*(1), 25–40.

Alhawari, S., Karadsheh, L., Nehari Talet, A., & Mansour, E. (2012). Knowledge-based risk management framework for information technology project. *International Journal of Information Management*, *32*(1), 50–65.

Andres, H. P., & Zmud, R. W. (2002). A contingency approach to software project coordination. *Journal of Management Information Systems*, *18*(3), 41–70.

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

Barki, H., & Hartwick, J. (1994). Measuring user participation, user Involvement, and user attitude. *MIS Quarterly*, *18*(1), 59–82.

Barki, H., Rivard, S., & Talbot, J. (2001). An integrative contingency model of software project risk management. *Journal of Management Information Systems*, *17*(4), 37–69.

Baroudi, J. J., Olson, M. H., & Ives, B. (1986). An empirical study of the impact of user involvement on system usage and information satisfaction. *Communications of the ACM*, *29*(3), 232–238.

Boehm, B., & Turner, R. (2003). Using risk to balance agile and plan-driven methods. *Computer*, *36*(6), 57–66.

Carlile, P. R. (2002). A pragmatic view of knowledge and boundaries: Boundary objects in new product development. *Organization Science*, *13*(4), 442–455.

Carlile, P. R. (2004). Transferring, translating, and transforming: An integrative framework for managing knowledge across boundaries. *Organization Science*, *15*(5), 555–568.

Chatzoglou, P. D., & Diamantidis, A. D. (2009). IT/IS implementation risks and their impact on firm performance. *International Journal of Information Management*, *29*(2), 119–128.

Chin, W. W. (1998). The partial lease squares approach to structural equation modeling. In G. A. Marcoulides (Ed.), *Modern methods for business research* (pp. 295–336). New York: Lawrence Erlbaum Associates.

Chiu, C. M., Hsu, M. H., & Wang, E. T. G. (2006). Understanding knowledge sharing in virtual communities: An integration of social capital and social cognitive theories. *Decision Support Systems*, *42*(3), 1872–1888.

Choe, J. M. (1996). The relationships among performance of accunting information systems, influence factors, and evolution level of information systems. *Journal of Management Information Systems*, *12*(4), 215–239.

Choe, J. M. (1998). The effects of user participation on the design of accounting information systems. *Information & Management*, *34*(3), 185–198.

Clarke, P., & O'Connor, R. V. (2012). The situational factors that affect the software development process: Towards a comprehensive reference framework. *Information and Software Technology*, *54*(5), 433–447.
CommonWealth. (2009). Top 1000 manufacturing firms in Taiwan. *CommonWealth Magazine*, *421*, 204–220.
Faraj, S., & Sproull, L. (2000). Coordinating expertise in software development teams. *Management Science*, *46*(12), 1554–1568.
Field, J. M., Ritzman, L. P., Safizadeh, M. H., & Downing, C. E. (2006). Uncertainty reduction approaches, uncertainty coping approaches, and process performance in financial services. *Decision Sciences*, *37*(2), 149–175.
Galbraith, J. R. (1977). *Organization design*. Reading,MA: Addison-Wesley.
Gallivan, M. J., & Keil, M. (2003). The use developer communication process: A critical case study. *Information Systems Journal*, *13*(1), 37–68.
Grant, R. M. (1996). Toward a Knowledge-based Theory of the Firm. *Strategic Management Journal*, *17*(Winter 96 Special Issue), 109L 122.
Guimaraes, T., & Igbaria, M. (1997). Client/server system success: Exploring the human side. *Decision Sciences*, *28*(4), 851–876.
Guinan, P. J., Cooprider, J. G., & Faraj, S. (1998). Enabling software development team performance during requirements definition: A behavioral versus technical approach. *Information Systems Research*, *9*(2), 101–125.
Hair, J., Black, W., Babin, B. J., Anderson, R., & Tatham, R. (2006). *Multivariate data analysis* (6th ed.). London, UK: Prentice-Hall International.
He, J., & King, W. R. (2008). The role of user participation in information systems development: Implications from a meta-analysis. *Journal of Management Information Systems*, *25*(1), 301–331.
Henderson, J. C., & Lee, S. (1992). Managing I/S design teams: A control theories perspective. *Management Science*, *38*(6), 757–777.
Hsu, J. S.-C., Chan, C.-L., Liu, J. Y.-C., & Chen, H.-G. (2008). The impacts of user review on software responsiveness: Moderating requirements uncertainty. *Information & Management*, *45*(4), 203–210.
Hsu, J. S.-C., Chen, H.-G., Jiang, J., & Klein, G. (2010). The role of user review on information system project outcomes: A control theory perspective. *International Journal of Information Technology Project Management*, *1*(1), 1–14.
Hsu, J. S.-C., Lin, T.-C., Cheng, K.-T., & Linden, L. P. (2012). Reducing requirement incorrectness and coping with its negative impact in information system development projects. *Decision Sciences*, *43*(5), 929–955.
Hsu, J. S.-C., Lin, T.-C., Zheng, G.-T., & Hung, Y.-W. (2012). Users as knowledge co-producers in the information system development project. *International Journal of Project Management*, *30*(1), 27–36.
Hulland, J. (1999). Use of partial least squares (PLS) in strategic management research: A review of four recent studies. *Strategic Management Journal*, *20*(2), 195–204.
Hunton, J. E., & Price, K. H. (1997). Effects of the user participation process and task meaningfulness on key information system outcomes. *Management Science*, *43*(6), 797–812.
Hwang, M. I., & Thorn, R. G. (1999). The effect of user engagement on system success: A meta-analytical integration of research findings. *Information & Management*, *35*(4), 229–236.
Ives, B., & Olson, M. H. (1984). User involvement and MIS success: A review of research. *Management Science*, *30*(5), 586–603.
Jiang, J. J., Klein, G., & Chen, H. G. (2006). The effects of user partnering and user non-support on project performance. *Journal of the Association for Information Systems*, *7*(2), 68–88.
Jones, M. C., & Harrison, A. W. (1996). IS project team performance: An empirical assessment. *Information & Management*, *31*(2), 57–65.
Kale, P., Singh, H., & Perpmutter, H. (2000). Learning and protection of proprietary assets in strategic alliances: Building relational capital. *Strategic Management Journal*, *21*(3), 217–237.
Kaplan, B., & Duchon, D. (1988). Combining qualitative and quantitative methods in information systems research: A case study. *MIS Quarterly*, *12*(4), 571–586.
Kappelman, L. A., & McLean, E. R. (1991). The respective roles of user participation and user involvement in information system implementation success. In *The thirteenth international conference on information system* New York, NY, USA, (pp. 339–349).
Kim, E., & Lee, J. (1986). An exploratory contingency model of user participation and MIS use. *Information & Management*, *11*(2), 87–97.
Kim, H. W., & Kankanhalli, A. (2009). Investigating user resistance to information systems implementation: A status quo bias perspective. *MIS Quarterly*, *33*(3), 567–582.
Kirsch, L. J., Sambamurthy, V., Ko, D. G., & Purvis, R. L. (2002). Controlling information systems development projects: The view from the client. *Management Science*, *48*(4), 484–498.
Ko, D. G., Kirsch, L. J., & King, W. R. (2005). Antecedents of knowledge transfer from consultants to clients in enterprise system implementations. *MIS Quarterly*, *29*(1), 59–85.
Lapointe, L., & Rivard, S. (2005). A multilevel model of resistance to information technology implementation. *MIS Quarterly*, *29*(3), 461–491.
Lee, G., & Xia, W. (2010). Toward agile: An integrated analysis of quantitative and qualitative field data on software development agility. *MIS Quarterly*, *34*(1), 87.
Lin, W. T., & Shao, B. B. M. (2000). The relationship between user participation and system success: A simultaneous contingency approach. *Information & Management*, *37*(6), 283–295.
Mak, B., Schmitt, B. H., & Lyytinen, K. (1997). User participation in knowledge update of expert systems. *Information & Management*, *32*(2), 55–63.
Mingers, J. (2001). Combining IS research methods: Towards a pluralist methodology. *Information Systems Research*, *12*(3), 240–259.
Misra, S. C., Kumar, V., & Kumar, U. (2009). Identifying some important success factors in adopting agile software development practices. *Journal of Systems and Software*, *82*(11), 1869–1890.
Mitchell, V. L. (2006). Knowledge integration and information technology project performance. *MIS Quarterly*, *30*(4), 919–939.
Nidumolu, S. (1995). The effect of coordination and uncertainty on software project performance: Residual performance risk as an intervening variable. *Inforamtion Systems Research*, *6*(3), 191–219.
Nidumolu, S. R. (1996). A comparison of the structural contingency and risk-based perspectives on coordination in software-development projects. *Journal of Management Information Systems*, *13*(2), 77–113.
Ouchi, W. G., & Maguire, M. A. (1975). Organizational control: Two functions. *Administrative Science Quarterly*, *20*(4), 559–569.
Parolia, N., Goodman, S., Li, Y., & Jiang, J. J. (2007). Mediators between coordination and IS project performance. *Information & Management*, *44*(7), 635–645.
Rumpe, B., & Schröder, A. (2002). Quantitative survey on extreme programming projects. In *Third International Conference on Extreme Programming and Flexible Processes in Software Engineering (XP2002)* Alghero, Italy, (pp. 95–100).
Sabherwal, R., Jeyaraj, A., & Chowa, C. (2006). Information system success: Individual and organizational determinants. *Management Science*, *52*(2), 1849–1864.
Saleem, N. (1996). An empirical test of the contingency approach to user participation in information systems development. *Journal of Management Information Systems*, *13*(1), 145–166.
Saunders, C. S., & Clark, S. (1992). EDI adoption and implementation: A focus on interorganizational linkages. *Information Resources Management Journal*, *5*(1), 9–20.
Schwalbe, K. (2002). *Information technology project management*. Cambridge, MA: Course Technology.
Tesch, D., Sobol, M. G., Klein, G., & Jiang, J. J. (2009). User and developer common knowledge: Effect on the success of information system development projects. *International Journal of Project Management*, *27*(7), 657–664.
Thompson, J. D. (1967). *Organizations in action*. New York: McGraw-Hill.
Tiwana, A., & Mclean, E. R. (2005). Expertise integration and creativity in information systems development. *Journal of Management Information Systems*, *22*(1), 13–43.
Van de Ven, A. H., Delbecq, A. L., & Koenig, R. J. (1976). Determinants of coordination modes within organizations. *American Sociological Review*, *41*(2), 322–338.
van den Hooff, B., & de Winter, M. (2011). Us and them: A social capital perspective on the relationship between the business and IT departments. *European Journal of Information Systems*, *20*(3), 255–266.
van den Hooff, B., & Huysman, M. (2009). Managing knowledge sharing: Emergent and engineering approaches. *Information & Management*, *46*(1), 1–8.
Venkatesh, V., Brown, S. A., & Bala, H. (2013). Bridging the qualitative–quantitative divide: Guidelines for conducting mixed methods research in information systems. *MIS Quarterly*, *37*(1), 21–54.
Wagner, E. L., & Piccoli, G. (2007). Moving beyond user participation to achieve successful IS design. *Communications of the ACM*, *50*(12), 51–55.
Wallace, L., & Keil, M. (2004). Software projects risks and their effect on outcomes. *Communications of the ACM*, *47*(4), 68–73.
Wallace, L., Keil, M., & Rai, A. (2004a). How software project risk affects project performance: An investigation of the dimensions of risk and an exploratory model. *Decision Sciences*, *35*(2), 289–321.
Wallace, L., Keil, M., & Rai, A. (2004b). Understanding software project risk: A cluster analysis. *Information & Management*, *42*(1), 115–125.
Wasko, M. M. L., & Faraj, S. (2005). Why should I share? Examining social capital and knowledge contribution in electronic networks of practice. *MIS Quarterly*, *29*(1), 35–57.

Yu Wen Hung is a Ph.D. candidate in the Department of Information Management, National Sun Yat-sen University. She earned her master degrees from Southern New Hampshire University in United States and National Sun Yat-Sen University in Taiwan. Her current research interests include information system development project management, information security, and electronic commerce. She has published papers in Information & Management, International Journal of Project Management and Project Management Journal and several proceedings of academic conference.

Dr. Shih-Chieh Hsu is an Associate Professor in the Department of Information Management, National Sun Yat-sen University, Taiwan. He received his Ph.D. degree from University of Central Florida. His research interests include electronic commerce and information system development project management. His research has been published in academic journals such as Decision Sciences, Decision Support Systems, Information Systems Journal, Information & Management, Communications of the AIS, and International Journal of Project Management.

Dr. Zhi-Yuan Su is an Associate Professor and Chairman in Department of Information Management, Chia-Nan University of Pharmacy and Science, Taiwan. His research focuses on project management, medical information and decision support system. He has published articles in journals such as Expert Systems With Applications and Journal of Medical Systems.

Dr. Hsieh-Hong Huang is Assistant Professor of the Department of Information Science and Management Systems at National Taitung University, Taiwan. His research interests include information systems development, decision support systems, and e-learning. His work has been published in several journals including Decision Support Systems, Communications of the Association of Information Systems, Project Management Journal, Educational Technology & Society, and British Journal of Educational Technology.