Fit of Development Methodologies in Software Projects

Full Paper

Sriram Rajagopalan
Department of Management Studies,
Indian Institute of Technology Madras,
Chennai – 600036, India
ms12d006@smail.iitm.ac.in

Saji K Mathew
Department of Management Studies,
Indian Institute of Technology Madras,
Chennai – 600036, India
saji@iitm.ac.in

Vijayan Sugumaran
School of Business Administration,
Oakland University
Rochester, MI 48309
sugumara@oakland.edu

Abstract

Software project outcomes characterized by meeting goal achievement and performance triad of budget, schedule and quality have been shown to be contingent upon project environment factors. However, choice of methodology and its implications on project outcomes still remain under-investigated. Following contingency theory, we empirically examine the effect of the fit between the choice of development methodology and project environment on the project outcome. We analysed a sample of 163 software development projects using PLS-SEM and our results show that the use of traditional methodology strongly countered the negative effect of requirement volatility on project outcome compared to agile methodologies and use of hybrid methodologies showed a stronger positive effect of project complexity on goal achievement. Further, for critical projects, use of agile methodologies favoured goal achievement.

Keywords

Software project outcome, contingency, goal achievement, agile development, traditional methodologies, methodology choice, requirement volatility, project complexity, project criticality.

Introduction

Although software project management has evolved over several decades to address the challenges caused by uncertain environment, project challenges continue to prevail (Linberg et al. 1999; Standish Group 2015). A stream of research in project management has identified several factors that influence project outcome and developed understanding about contingent factors that adversely affect project outcome (Xia and Lee 2004). Along these lines, some scholars have investigated how levers of project management could be potentially deployed to manage contingent factors that could impact desired project outcome. Agile and adaptive methodologies have evolved as projects were becoming more and more complex with uncertain requirements (Mohammad et al. 2013). However, the mechanism by which project management choices influence project outcomes is not very clear in the extant literature.

The main aim of our study is to analyse the fit between project environment factors and choice of methodology and its impact on project outcomes. We statistically analyse a sample of 163 projects from the Indian software industry to address our research question. We use contingency theory approach, originally developed for organizational design and later extended to study project organization (Barki et al. 2001).

Our study contributes to software project management literature in three ways. First, ours is one of the first studies which has empirically examined if the choice of development methodology, especially adaptive methods, lead to improved project outcome. Second, we develop an understanding of the role of choice of methodology (agile, hybrid and traditional) as a categorical moderator between project characteristics and project outcomes. This unique approach of 'fit as moderation' (Venkataraman 1989) results in understanding the "fit" between project environment factors and development methodology and how that fit influences project outcome. Third, our study provides useful insights for project management practice, particularly to software development community, in decisions pertaining to the choice of development methodology. Prior studies have reported a lack of scientific approach in the choice methodologies (Ahimbisibwe et al. 2015) and this study guides managers to look at their projects through factors relevant to their decision context.

Theoretical Background

Research in software projects has approached project management as a tool to align project environment factors which are volatile with software development process (Barki et al. 2001). Following this approach, prior research extended the use of contingency theory to software projects, whereby project outcome has been modeled as contingent on the fit between project characteristics and project management.

Contingency Theory

Information systems research has applied contingency theory in different contexts such as management information systems success (Venkatraman 1989) and software development (Franz 1985). But the use of contingency theory was also accompanied by criticisms (Weill and Olson 1989). Following the classification of Boehm and Turner (2005), we propose to empirically analyse the contingency of software project outcomes on three important project environment factors: dynamic changes of business scope (volatility), complexity and criticality and development methodology as project factor

Requirement volatility

Changes in requirements is common in software development due to changes in business needs and priorities of customers and other stakeholders (Verner et al. 2007). They are broadly characterized as change in scope, new additional requirements and deleting existing requirements which are not within the control of the project team but are directly or indirectly influenced through external environment. With the advent of large-scale complex systems, the uncertainty in business user needs and more globalization of development has caused requirements volatility to increasingly impact software reliability (Damian and Zowghi 2003).

Project Complexity

Project complexity is defined as the sum of all parts which make up a project (Richardson et al. 2001). To develop complex software systems, the highly preferable approach has been to modularize them (Mccabe 1976). Following agile development principles, Benbya and McKelvey (2006) proposed a co-evolutionary framework to develop complex adaptive systems including modular design and iterative development.

Project Criticality

Project criticality is defined based on the extent of stakeholder involvement and monitoring from both client and vendor perspectives (Stock and Tatikonda 2008). Some studies have used contingency approach and observed that the relationship between stakeholder participation and system use was contingent on top management support, monitoring, user attitudes, task complexity and participation characteristics of both vendor and customer (Tait and Vessey 1988). Barki and Hartwick (1989) brought out significant differences between stakeholder participation and involvement which goes close to the spirit of agile methodologies where a customer is continually involved in the development, involving and facilitating the change.

Software Development Methodologies

Royce (1970) proposed the sequential process model of analysis, design, development and testing, which is the fundamental block of waterfall methodologies. Basili and Turner (1975) proposed iterative enhancement technique for software development, which provided top-down step-wise refinement approach. Agile methodologies have shown flexibility to address constraints, without demanding major upfront investments, also being adaptable to changing market conditions (Mohammad et al. 2013). Petersen and Wohlin (2009) highlighted the limitation of agile methods such as a visible increase in project efforts, lack of focus on architectural design, lack of scalability, expectation of highly technical expertise, lack of inter-team communication and dedicated commitment from business stakeholders.

Software Project Outcome

Extant literature in project management has taken two different views of project management outcome- one stream of studies that examine project outcome as success or failure (eg.: DeLone and McLean 2003) and the other, which deals with project outcome in terms of the three specific project goals (or constraints) of budget, schedule and scope (eg.: Barki et al. 2001; Lee and Xia 2010). Saarinen (1990) found that organizations were using methodologies inconsistently following arbitrary approaches resulting in poor project outcomes. The study strongly suggested that, methodologies should receive adequate coverage in all phases of the development cycle which was found majorly lacking. Howell et al. (2010) in another recent study reported that the existence of numerous methodology choices in itself posed difficulty in choosing the best fit option and so, the developer community opted for the tried and tested methodologies in their organization and ignored alternatives. However, irrespective of whether a methodology was standardized, customized or a combination of both were used, if the implementation was limited and not utilized to its potential, the project efficiency was found to be severely impacted (Joslin and Müller, 2015).

In summary, there have been very limited empirical studies on the contingency effect of project environment factors and project factors on the project outcomes. Ahimbisibwe et al. (2015) very recently studied the contingency approach of selection of project management approaches to be adopted to fit project characteristics and project environment to determine project success. However, their study was more conceptual at meta-analytic level. Our study addresses this gap by developing an analytical framework from extant literature and analyses the effect of the relationship between the project environment and project outcome and how the choice of methodology influences this relationship.

Hypotheses Development

Drawing on contingency theory applied to organizations, we argue that development methodology is a choice that a project organization uses to align project contingency factors with project outcomes. In other words, organizations try to find a fit between project environment and project context by exercising its choice on the development methodology. As this study seeks to understand the effect of the choice of methodology on project outcome, we hypothesized methodology as a moderator to test the "fit" between contingency variables and project management (Venkataraman 1989). Since our focus is on testing the effect of methodology in addressing project contingency factors, we choose *goal achievement* as a project outcome variable (Linberg et al. 1999). The research model in Fig. 1 shows the relationships being studied.

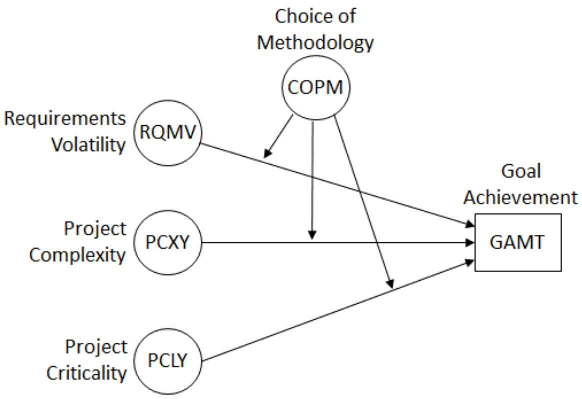

Figure 1. Research Model

Project Environment and Contingency Factors

As our study seeks to understand how development methodology is used as a choice variable to deal with project contingencies, we selected changes in project scope, project complexity and project criticality as the key contingency variables that influence project outcome. These are project environment factors contingent on the nature of the software project being developed.

Prior research has shown that an increase in requirements volatility adversely affects project outcome impacting the project goal due to residual performance risk (Nidumolu 1996). Scholars suggested the adoption of agile methodologies to accept and manage changes in customer requirements (Lee and Xia 2010). However, it has been reported that, under conditions of very high requirements volatility, even the use of agile methodologies would compromise the outcome of functionality achievement (Sharma et.al. 2012). Drawing on the inferences from these studies we hypothesize the following:

H1a: Software project requirements volatility negatively influences goal achievement

H1b: The choice of software development methodologies moderates the relationship between requirements volatility and goal achievement

Xia and Lee (2004), broadly categorized project complexity along the dimensions of IT and organization (structural and dynamic). They operationalized project outcome in terms of delivered functionality, cost, quality and schedule. Their empirical study reported a negative effect of project complexity on project outcome and suggested measures such as process focus and use of appropriate methodology to reduce the negative effect. Whitney et al. (2013) showed that, as the complexity increases in terms of modularity and components being involved, use of appropriate methodology will improve the project outcome. Based on the findings from previous studies, we hypothesize the following:

H2a: Software project complexity negatively influences goal achievement

H2b: The choice of software development methodology moderates the relationship between project complexity and goal achievement

Project criticality is related to the importance of the project to client organization and in turn to a vendor who executes the project. Criticality is characterised by commitment from business users and vendor's top management and is usually accompanied by project monitoring by representatives of the client and vendor organizations (Schmidt et al. 2001). While traditional methods of software development do not stress stakeholders' involvement continuously, new methods like agile, iterative development expect dedication and commitment from customer and vendor to achieve the project goals, either stakeholders aligning to project criticality and importance (Hajjdiab et al. 2011). Hence, we hypothesize the following relationships:

H3a: Software project criticality positively influences goal achievement

H3b: The choice of software development methodology moderates the relationship between project criticality and goal achievement

Research Methodology

Data Collection and Measurement

We developed a survey instrument using measures already detailed in existing literature on project requirements volatility, complexity, criticality and outcome. In order to check the face validity, the survey instrument was reviewed by 6 experts - 4 IS professionals from the IT industry and 2 senior faculty members in the IS area and measurement scales were slightly modified. The sample was chosen from IT organizations across the globe engaged in software development projects. We pursued key informants approach where the identified respondents were qualified specialists knowledgeable about software development methodologies, who played a significant role in the process of decision making in choosing the methodology for software development projects (Kumar et.al., 1993). Selected respondents were requested to choose a project, which they managed end to end, while responding the questionnaire.

We developed an online version of the survey instrument and the survey link was shared directly through email to the participants with detailed instructions and the participation in the survey was promoted through different industry forums All our constructs were identified as reflective from previous literature (Hair et al. 2014). Table 1 lists the details of the measure development with references.

Construct	Item Description	Scale	Reference
Requirement Volatility (RQMV)	RQMV1: level of requirements fluctuation during initial phase	Likert (1-5): 1= Very Low; 5= Very High	Nidumolu, 1996; Verner et al. 2007;
	RQMV2: degree of variation in requirements from start to end	1=Never; 5=Very Often	

	RQMV3: level of requirements fluctuation during testing phase	Likert (1-5): 1= Very Low; 5= Very High	
	RQMV4: How much effort was required to consolidate the requirements and bring the users to common understanding as against the planned effort for requirements	Likert (1-5): 1=Strongly Disagree; 5=Strongly Agree	
Project Complexity (PCXY)	PCXY1: How complex was the project in terms of number of stated project objectives?	Likert (1-5): 1= Very Low; 5= Very High	Xia and Lee 2004; Wallace and Keil 2004
	PCXY2: How complex was the project in terms of number of phases involved?	Likert (1-5): 1= Very Low; 5= Very High	
	PCXY3: How complex was the project in terms of number of modules and components developed during the project?	Likert (1-5): 1= Very Low; 5= Very High	
	PCXY4: How complex was the project in terms of number of interfaces designed for the project?	Likert (1-5): 1= Very Low; 5= Very High	
Project Criticality (PCLY)	PCLY1: Effort spent by client in project monitoring was	Likert (1-5): 1= Very Low; 5= Very High	Adapted from (Verner et al. 2007; Schmidt et al. 2001)
	PCLY2: Importance given to this project in my organization	Likert (1-5): 1= Very Low; 5= Very High	
Development Methodology (COPM)	COPM: Choice of project methodology	Categorical: 1= Agile; 2= Hybrid; 3= Traditional	Boehm and Turner 2005; Ahimbisibwe et al. 2015
Goal Achievement (GAMT)	GAMT1: The end outcome of the project met the functional goals defined by the customer	Likert (1-5): 1=Strongly Disagree; 5=Strongly Agree	Wallace and Keil 2004; Lee and Xia 2010;
	GAMT2: The end outcome of the project met the user requirements	Likert (1-5): 1=Strongly Disagree; 5=Strongly Agree	
	GAMT3: The end outcome of the project met the technical requirements	Likert (1-5): 1=Strongly Disagree; 5=Strongly Agree	

Table 1. Measure Development

Data Analysis Procedure

We received 180 responses to our survey out of which 17 cases were dropped from further analysis due to incomplete responses or the respondents were not meeting the key informant criteria, resulting in 163 responses for further analysis. The average size of the customer and the service provider organization in our sample was about 40,000 employees with minimum 1000 and maximum 100,000 employees. The average size of the customer organization was about 30,000 employees, the size ranging from 1,000 employees to 50,000 employees. The average planned project size was 1370 person months, ranging from 300 to 3000 person months. 43% projects were for customers from US–Canada region and 18% customers from Europe-UK region. While projects for customers from Asia were 15% and Australia-New Zealand were 5%, 19% of the customer projects were global who had presence in more than one geography. Regarding the type of the development projects for which the respondents have provided their responses, new development projects were 32%, reengineering projects constituted 12%, enhancement projects were 10% and maintenance projects were 8%. Projects under "others" category were combinations of more than 1 of these project types and covered 38% of the sample size. The projects represented different business domains and the five top domains which covered 65% of the sample were banking and finance, manufacturing, retail, healthcare and telecom.

We used Partial Least Squares (PLS) based Structural Equation Modeling (SEM) to test our research model and used smartPLS software V3.2.3. PLS-SEM estimation is less sensitive to sample size and does not assume normality of data (Hair et al. 2014). PLS uses a nonparametric bootstrapping method, involving repeated random samples, replacing from original sample to create a new set of a bootstrap sample. This bootstrap sample enables to test the significance of the path coefficients estimated (Hair et al. 2014).

Measurement Model

We followed the procedure used by Liang et al. (2007) for the evaluation of our measurement model. We estimated construct validity through Confirmatory Factor Analysis (CFA) using the measure of the construct (loadings), other theoretically associated measures (convergent validity) and measures varying independently (discriminate validity).

Table 2 describes our measurement model and gives the item loadings and Average Variance Extracted (AVE). We dropped three items which were designed to be part of Requirements Volatility (RQMV), two of Project Complexity (*PCXY*) and one of Project Criticality (*PCLY*), as they had poor validity measure. All loadings are higher than 0.6 to be accepted as measures of the respective constructs and the values of AVE is greater than the prescribed minimum value of 0.5 showing that the constructs accounts at least for 50% of the variance in their respective item measures (Bagozzi and Yi 1988).

Discriminant validity was assessed through two procedures: one, comparing the square root of values of AVEs of each construct with inter-construct correlations and two, comparing the cross correlations of the items with their constructs and other constructs (Gefen and Straub 2005). Table 3 displays the inter-construct correlations and the values highlighted in bold across the diagonal represent the square root of AVE values shared with the measures. All values across the diagonal are sufficiently greater than the desired value of 0.5 and all these values are greater than the off-diagonal values in their corresponding row and corresponding column (Fornell et al. 1981). So the two tests affirm the discriminant validity of our measurement model. Table 2 shows the composite reliability coefficient values for all constructs to be above 0.7, which demonstrates good reliability indicating internal consistency among all the reflective latent constructs (Hair et al. 2014).

Construct	**Indicator**	**Loadings**	**Composite Reliability**	**Average Variance Extracted (AVE)**
Requirements Volatility (RQMV)	RQMV1	0.715	0.822	0.544
	RQMV2	0.733		
	RQMV3	0.792		
	RQMV4	0.684		
Project Complexity (PCXY)	PCXY1	0.831	0.864	0.616
	PCXY2	0.829		
	PCXY3	0.706		
	PCXY4	0.765		
Project Criticality (PCLY)	PCLY1	0.727	0.768	0.625
	PCLY2	0.849		
Goal Achievement (GAMT)	GAMT1	0.833	0.842	0.642
	GAMT2	0.837		
	GAMT3	0.727		

Table 2. Reliability and Convergent Validity of the Measurement Model

	RQMV	PCXY	PCLY	GAMT
Requirements Volatility (RQMV)	**0.732**			
Project Complexity (PCXY)	0.329	**0.785**		
Project Criticality (PCLY)	0.250	0.456	**0.791**	
Goal Achievement (GAMT)	-0.122	0.258	0.254	**0.801**

Table 3. Discriminant Validity: Inter-correlations between Reflective Constructs

Common Method Bias

Two approaches were adopted to test common method bias in our model. First, we employed Harman single-factor test for all items of constructs (Podsakoff et al. 2003). The test showed four distinct factors (Eigenvalues > 1.0) and the largest of the factor accounted only for 24.9% of the variance of the model. To strengthen this result, we further used unmeasured latent method factor (Podsakoff et al. 2003). It was found that the average substantively explained variance of the indicators is significantly higher than the average method based variance. Also, all the method factor loadings were not significant ($p < 0.05$). Therefore, we rule out common method variance in our measurement based on the two tests.

Structural Model

We followed two approaches for testing the proposed structural model covering the hypothesised relationships among the project environment factors and the criterion variables. (i) Using all the constructs in one iteration and testing the direct effects of the exogenous constructs on the endogenous constructs. (ii) Testing the moderating effects of choice of methodology on the same relationships using PLS multi-group analysis. The results of full structural model estimation are presented in Table 4.

For the first approach, we used PLS-SEM conducting a sequence of regressions in terms of weight vectors. We analysed the signs and magnitude of the relationship of each exogenous construct with an endogenous construct for the original sample (n=163). To further determine the significance of the estimates, we used bootstrapping method (Henseler et al. 2009) with n1=3000 bootstrapped data. Comparing the path coefficients value determined through normal PLS-SEM without and with bootstrapping, we found that signs were consistent and observed no major difference in the magnitude of path coefficients. The model explained 18% variance for *goal achievement*.

Structural model's results in Table 4 further show that the *requirements volatility* had a significant negative effect on the *goal achievement* (β= -0.276; $p<0.05$) and the *project criticality* had a significant positive effect (β=0.204; $p<0.05$) on the *goal achievement* supporting our hypotheses H1a and H3a respectively. However, at the same significance level, the *project complexity* is observed to have a significant positive effect on the *goal achievement*, in contrast to our hypothesis (β=0.284; $p<0.05$). So, although the relationship was significant, H2a was not accepted.

Moderating Effects Hypotheses Testing

Direct Effects – Goal Achievement				
Variables	Original β (n=163)	Mean boot β (std error) (n1=3000)	R^2 (n1= 3000)	Comments
Requirements Volatility (RQMV)	-0.260	**-0.276 (0.095) ***	0.02	H1a supported
Project Complexity (PCXY)	0.268	**0.284 (0.087) ***	0.08	H2a not accepted
Project Criticality (PCLY)	0.204	**0.204 (0.087) ***	0.08	H3a supported
Moderation Effects of Choice of Methodology- Goal Achievement				
Variables	Path Coefficients Mean Boot β (n=3000)			Comments
	Agile (n1=30)	Hybrid (n2=69)	Traditional (n3=64)	
Requirements Volatility (RQMV)	-0.526	-0.039	**-0.319***	H1b partially supported
Project Complexity (PCXY)	0.075	**0.499***	0.220	H2b partially supported
Project Criticality (PCLY)	**0.681***	-0.010	0.087	H3b partially supported

Table 4. Structural Model

To test the moderation effect of *choice of methodology* on the various relationships, we used PLS Multi-Group Analysis (PLS-MGA) technique. The *choice of methodology* variable in our study is designed to be a categorical measure having three discrete values and in such cases, the test for pure moderator effect can be performed using a multi-group specification of the structural equation model (Sauer and Dick 1993).

The results of the PLS-MGA using *choice of methodology* as moderator variable are also given in Table 4. Our results show that using traditional methodologies in projects led to a stronger and significant negative effect of *requirements volatility* on *goal achievement* (β=-0.319; $p<0.05$) compared to the effect based on the whole sample (β=-0.276; $p<0.05$). But the effect was not significant while using agile and hybrid methodologies. So the hypotheses H1b was partially supported. Furthermore, using hybrid methodologies led to stronger and significant positive effect of *project complexity* on *goal achievement* (β=0.499; $p<0.05$), as compared to the effect based on whole sample (β=0.284; $p<0.05$). However, the results were not significant either for agile or for traditional methodologies. So the hypotheses H2b was only partially supported. Using agile methodologies in projects led to a stronger positive effect of *project criticality* on *goal achievement* (β=0.681; $p<0.05$) compared to the effects of the whole sample (β=0.204; $p<0.05$). But, the choice of traditional and hybrid methodologies did not have a significant moderating effect on the relationship of *project criticality* on *goal achievement*. So the hypotheses H3b was partially supported.

Discussion and Implications

Our research followed contingency theory based approach to determine how the choice of methodologies influences the relationship of project characteristics and project outcomes in terms of goal achievement. Our findings on the negative effects of *requirements volatility* on *goal achievement* is consistent with earlier studies (Wallace and Keil 2004). However, the moderating effects of *choice of methodologies* on the relationship between *requirements volatility* on *goal achievement* did not favor agile methodologies, contradicting the propositions from some prior studies (e.g.: Mohammad et al. 2013).

The comparatively weaker negative effect of requirements volatility on project outcomes when using traditional methodology may be due to few reasons. Because of frequent change in requirements and changes in priorities, there is often deviation from original scope and very high variability between originally planned goals and goals achieved (Sharma et al. 2012). Further, while using traditional methodologies the decision making lies with project manager whereas in agile methodologies decision making is driven by highly empowered cohesive teams. Many times, individuals in teams privately agree on the means of resolving an issue but not shared with the group due to Abilene Paradox symptom (Harvey 1974). So, this leads to lack of consensus and cohesion (McAvoy and Butler 2009).

While using agile, when requirement volatility is very high, there is a possibility of resource wastage thereby cost could escalate due to continual rework (Sharma et al. 2012). For large systems, the amount of inter-dependencies and integration requirements are high; if not well-defined upfront, may lead to development of a system different from what was envisaged originally (Turk et al. 2005). In large hierarchical organizations, for effective requirements management, it is suggested to follow traditional approach at the global level and agile approach at the local level thereby fostering mixed approach (Kumar & Kumar 2011).

It is also interesting to note the positive effect of *project complexity* on *goal achievement* contradicting some findings from prior literature (Xia and Lee 2004). This is in alignment to the process maturity in large organizations developing complex systems, which can also be observed through our survey responses. The average size of vendor organization was 40,000 and in response to the question about adhering to process practices consistently across all stages of projects, 75% of respondents have concurred. The moderating effect of *choice of methodology* on the relationship between *project complexity* and *goal achievement* was not found to be significant either for agile or for traditional methodologies while it was significantly positive for hybrid methodologies. This is consistent with the practice that, for large mission-critical complex projects it is highly preferable to be moderate in the approach and follow hybrid methodologies during every phase of the development (Turk et al. 2005).

The significant and positive direct effects of *project criticality* on *goal achievement* were found to be resonating with the study of prior researchers (Soderberg et al. 2013). It appears that when the project is highly critical, the tendency of the stakeholders is to prioritize completing the project (delivering the defined scope) primarily at the expense of other factors. Also, our results show significant and positive moderating effects of agile methodologies on the relationship between *project criticality* and *goal achievement,* in agreement with prior studies (Hajjdiab et al. 2011). This implies that while using agile methodologies, monitoring and involvement is high from stakeholder which positively influences to meet the desired goals.

Limitations and Future Research

Although our study produced interesting results useful for theory and management practice, the study has some limitations. Our sample size was small as, often the project managers approached were constrained by the non-disclosure agreement with their organization and customers. To overcome the limitations due to this, we chose PLS path modelling and bootstrapping procedure for significance testing which produced moderately stable results.

Our study included only the relationship of project environment and project outcome but there are other constructs which could significantly influence project outcome as highlighted in earlier studies such as team turnover, team experience (Gopal et al. 2003), team response extensiveness and team response efficiency (Lee and Xia 2010). Our future work aims to empirically analyse how these team related variables influence project outcomes based on the choice of methodologies. Our study was conducted at a high-level classification of methodology and we analysed them as traditional, agile and hybrid. However, it would be insightful to further investigate the moderating effects of specific types of agile methodologies like Scrum and XP, and specific types of traditional methodologies like waterfall and spiral.

References

Ahimbisibwe, A., Cavana, R. Y., and Daellenbach, U. 2015. “A contingency fit model of critical success factors for software development projects: A comparison of agile and traditional plan-based methodologies,” *Journal of Enterprise Information Management* (28:1), pp. 7-33.

Bagozzi, R. P., and Yi, Y. 1988. “On the evaluation of structural equation models,” *Journal of the academy of marketing science* (16:1), pp. 74-94.

Barki, H., and Hartwick, J. 1989. “Rethinking the concept of user involvement,” *MISQ* (13:1), pp. 53-63.

Barki, H., and Suzanne Rivard, J. T. 2001. “An integrative contingency model of software project risk management,” *Journal of Management Information Systems* (17:4), pp. 37-69.

Basili, V. R., and Turner, A. J. 1975. “Iterative enhancement: A practical technique for software development,” *IEEE Transactions on Software Engineering* (4), pp. 390-396.

Benbya, H., and McKelvey, B. 2006. “Toward a complexity theory of information systems development,” *Information Technology & People* (19:1), pp. 12-34.

Boehm, B., and Turner, R. 2005. “Management challenges to implementing agile processes in traditional development organizations,” *IEEE Software* (22:5), pp. 30-39.

Damian, D. E., and Zowghi, D. 2003. “RE challenges in multi-site software development organisations,” *Requirements engineering* (8:3), pp. 149-160.

DeLone, W. H., and McLean, E. R. 2003. “The DeLone and McLean model of information systems success: a ten-year update,” *Journal of management information systems* (19:4), pp. 9-30.

Fornell, C. G., and Larcker, D. F. 1981. “Evaluating structural equation models with unobservable variables and measurement error,” *Journal of Marketing Research* (18:1), pp. 39–50.

Franz, C. R. 1985. “User Leadership in the Systems Development Life Cycle: A Contingency Model,” *Journal of Management Information Systems* (2:2), pp. 5-25.

Gefen, D., and Straub, D. 2005. “A practical guide to factorial validity using PLS-Graph: Tutorial and annotated example,” *Communications of the Association for Information systems* (16:1), pp. 91-109.

Hair, J. F., Hult, G. T. M., Ringle, C., and Sarstedt, M. 2014. *A primer on partial least squares structural equation modeling (PLS-SEM)*, Sage Publications. USA.

Hajjdiab, H., and Taleb, A. S. 2011. “Adopting agile software development: Issues and challenges,” *International Journal of Managing Value and Supply Chains* (2:3), pp. 1-10.

Harvey, J. 1974. “The Abilene paradox: the management of agreement,” *Organizational Dynamics* (3:1), pp. 63–80.

Henseler, J., Ringle, C. M. and Sinkovics, R. R. 2009. “The use of partial least squares path modeling in international marketing,” *Advances in International Marketing* (20:1), pp. 277-320.

Howell, D., Windahl, C., and Seidel, R. 2010. “A project contingency framework based on uncertainty and its consequences,” *International Journal of Project Management* (28:3), pp. 256-264.

Joslin, R., and Müller, R. 2015. “Relationships between a project management methodology and project success in different project governance contexts,” *International Journal of Project Management* (33:6), pp. 1377-1392.

Kumar, S. A. and Kumar, T. A. 2011. “Study the Impact of Requirements Management Characteristics in Global Software Development Project: An Ontology Based Approach,” *International Journal of Software Engineering and Application* (2:4), pp. 107.

Kumar, N., Stern, L. W., and Anderson, J. C. 1993. “Conducting interorganizational research using key informants,” *Academy of management journal* (*36*:6), pp. 1633-1651.

Lee, G., and Xia, W. 2010. “Toward agile: an integrated analysis of quantitative and qualitative field data,” *MISQ* (34:1), pp. 87-114.

Liang, H., Saraf, N., Hu, Q., and Xue, Y. 2007. “Assimilation of enterprise systems: the effect of institutional pressures and the mediating role of top management,” *MIS quarterly* (31:1), pp. 59-87.

Linberg, K. R. 1999. “Software developer perceptions about software project failure: a case study,” *Journal of Systems and Software* (49:2), pp. 177-192.

McAvoy, J. and Butler, T. 2009. "The role of project management in ineffective decision making within Agile software development projects," *European Journal of Information Systems* (18:4), pp. 372-383.

McCabe, T. J. 1976. "A complexity measure," *IEEE Trans on Software Engineering* (4), pp. 308-320.

Mohammad, A.H., Alwada, T., and Ababneh, J.M.A. 2013. "Agile software methodologies: strength and weakness," *International Journal of Engineering Science and Technology* (5:3), pp. 455–459.

Nidumolu, S. R. 1996. "Standardization, requirements uncertainty and software project performance," *Information & Management* (31:3), pp. 135-150.

Petersen, K., and Wohlin, C. 2009. "A comparison of issues and advantages in agile and incremental development between state of the art and an industrial case," *Jl of Sys & software* (82:9), pp. 1479-1490.

Podsakoff, P. M., MacKenzie, S. B., Lee, J. Y., and Podsakoff, N. P. 2003. "Common method biases in behavioral research: a critical review of the literature and recommended remedies," *Journal of applied psychology* (88:5), pp. 879-903.

Richardson, K. A., Cilliers, P., and Lissack, M. 2001. "Complexity Science," *Emergence* (3:2), 6-18.

Royce, W. W. 1970. "Managing the development of large software systems," in *Proceedings of IEEE WESCON* (26:8), pp. 1-9.

Saarinen, T. 1990. "System development methodology and project success: An assessment of situational approaches," *Information & Management* (19:3), pp. 183-193.

Sauer, P. L., and Dick, A. 1993. "Using moderator variables in structural equation models," *Advances in consumer research* (20:1), pp. 637-640.

Schmidt, R., Lyytinen, K., and MarkKeil, P. C. 2001. "Identifying software project risks: An international Delphi study," *Journal of management information systems* (17:4), pp. 5-36.

Sharma, S., Sarkar, D., and Gupta, D. 2012. "Agile processes and methodologies: A conceptual study," *International journal on computer science and Engineering* (4:5), pp. 892-898.

Standish Group., 2015. "Chaos Report," *The Standish Group International Inc.*

Stock, G. N., and Tatikonda, M. V. 2008. "The joint influence of technology uncertainty and interorganizational interaction on external technology integration success," *Journal of operations management* (26:1), pp. 65-80.

Tait, P., and Vessey, I. 1988. "The effect of user involvement on system success: a contingency approach," *MIS quarterly* (12:1), pp. 91-108.

Turk, D., France, R., and Rumpe, B. 2005. "Assumptions underlying agile software development processes," *Journal of Database Management* (16:4), pp. 62-87.

Venkatraman, N. 1989. "The concept of fit in strategy research: Toward verbal and statistical correspondence," *Academy of management review* (14:3), pp. 423-444.

Verner, J., Cox, K., Bleistein, S., and Cerpa, N. 2007. "Requirements engineering and software project success: an industrial survey in Australia and the US," *Australasian Journal of Information Systems*, (13:1), pp. 225–238.

Wallace, L., and Keil, M. 2004. "Software project risks and their effect on outcomes," *Communications of the ACM* (47:4), pp. 68-73.

Weill, P., and Olson, M.H. 1989. "An Assessment of the Contingency Theory of Management Information Systems," *Journal of Management Information Systems (*6:1), pp. 59-85.

Whitney, K. M., and Daniels, C. B. 2013. "The Root Cause of Failure in Complex IT Projects: Complexity Itself," *Procedia Computer Science* (20), pp. 325-330.

Xia, W., and Lee, G. 2004. "Grasping the complexity of IS development projects," *Communications of the ACM* (47:5), pp. 68-74.