1-1-2016

Considerations for Effective Requirements Analysis in Offshore Software Development Projects: Lessons from Multi-method Research

Vanita Yadav
Insitute of Rural Management Anand

Dhruv Nath
Management Development Institute

Monica Adya
Marquette University, monica.adya@marquette.edu

Varadharajan Sridhar
International Institute of Information Technology (IIIT)

Published version. *Communications of the AIS*, Vol. 39, No. 11 (August 2016): 188-213. DOI.

Communications of the Association for Information Systems

Research Paper ISSN: 1529-3181

Considerations for Effective Requirements Analysis in Offshore Software Development Projects: Lessons from Multi-method Research

Vanita Yadav
Institute of Rural Management Anand (IRMA)
India
vanitay@gmail.com

Monica Adya
Marquette University
USA

Dhruv Nath
Management Development Institute (MDI)
Gurgaon, India

Varadharajan Sridhar
International Institute of Information Technology (IIIT)
Bangalore, India

Abstract:

Offshore software development using geographically distributed teams is an accepted practice in software development today. However, software development companies have largely only offshored the software development lifecycle's coding and testing phases. However, lately, offshoring the requirements analysis (RA) phase has become increasingly viable for several reasons including the software industry's maturation and improved communication technologies. However, successfully evaluating this highly interactive phase between geographically dispersed client and provider teams requires special considerations. In this paper, we present practical insights garnered from conducting experiments and surveys of IS professionals from the Indian software industry and from extensively examining the literature. Our findings confirm that, subject to certain best practices, one can effectively conduct RA in software projects offshore. We present these practices as lessons learned and provide related recommendations for industry and academia.

Keywords: Requirements Engineering, Requirements Specification, Requirements Analysis, Virtual Teams, Offshore Outsourcing, Global Software Development.

This manuscript underwent peer review. It was received 06/12/2015 and was with the authors for 3 months for 2 revisions. The Associate Editor chose to remain anonymous.

1 Introduction

Many companies began to offshore software development projects, a widely accepted practice today, due to the wage differential and related cost arbitrage between client nations such as the US and provider countries such as India. Over the past decade, several factors have further bolstered the rationale for offshoring (Carmel & Tjia, 2005), such as a) a business friendly climate in which provider nations offer tax incentives for export-oriented software, b) improved technologies and decreased communication costs, c) increasing talent pool of engineers and software professionals in provider countries, and d) standardization and maturation of software development tools and practices. Most companies that use a phased software development methodology that comprises requirements analysis, design, coding, testing, and implementation have raised the obvious question: which of these phases can one successfully offshore? One important factor in this decision has been the level of interaction necessary between the client and software development teams in each phase. Typically, firms consider phases such as coding and testing (in which client-provider teams conduct work relatively independently) as apt for offshoring. Conversely, research has considered activities related to requirements analysis (RA) that demand significant and continuous interactions between client and provider teams as less conducive for offshoring (Bhat, Gupta, & Murthy, 2006). However, as software firms continue to face cost pressures and as practices mature, they will continue to explore offshoring requirements-gathering activities. Successfully executing this interactive phase requires special considerations that research has not yet fully explored. In this paper, we present lessons learned and recommendations for successfully offshoring the RA phase of software development based on findings from multiple empirical studies (Yadav, Adya, Nath, & Sridhar, 2007; Nath, Sridhar, Adya, & Malik, 2008; Yadav, 2008; Yadav, Adya, Sridhar, & Nath, 2009, 2013).

While much research has addressed best practices for successfully offshoring later phases of software development (e.g., Dibbern, Goles, Hirschheim, & Jayatilaka, 2004; Piccoli, Powell, & Ives, 2004; Yadav & Gupta, 2008), few studies have addressed offshoring the RA phase (Yadav, 2011) even though several researchers have identified a need to do so (e.g., Robinson, 1990; Ocker, Hiltz, Turoff, & Fjermestad, 1995; Edwards & Sridhar, 2005; Evaristo, Watson-Manheim, & Audy, 2005; Nath et al., 2008; Yadav et al., 2009; Yadav et al., 2013). With continuing improvements in communication technologies and best practices for software development, research has projected offshoring to become more apparent throughout the entire lifecycle, including in the RA phase (Evaristo et al., 2005; Nath et al., 2008; Yadav et al., 2013). Demonstrations of successfully using conferencing technologies to support requirements determination in an asynchronous distributed environment, such as in Ocker et al. (1995), further point to the feasibility of offshoring the RA phase. However, apart from such diffused research, we understand little about the processes, structures, and best practices necessary for successfully offshoring RA. We address this gap by exploring three research questions:

RQ1: Can one offshore the RA phase? How effective is RA with offshore virtual teams when compared to RA with traditional, co-located teams?

RQ2: How do changes in requirements impact success of the RA phase when conducted in an offshore mode?

RQ3: What factors impact the success of the RA phase in offshore projects?

This paper proceeds as follows. In Section 2, we discuss the motivation and theoretical background for our study on offshore RA. In Section 3, we review the literature, and, in Section 4, we present lessons learned from the collective findings across the various studies reported in this paper. In Sections 5 and 6, we discuss recommendations and implications for practice and research, respectively. Finally, in Section 7, we conclude the paper.

2 Motivation and Theoretical Background

2.1 Offshore Requirements Analysis (RA)

Requirements analysis (RA) involves deciding what a system should do and forms one of the most essential, yet challenging, phases in software development (Crowston & Kammerer, 1998; Guinan, Cooprider, & Faraj, 1998; Yadav, 2011). RA is a phase of the information system development lifecycle wherein “the information and information processing services needed to support select objectives and functions of the organization are i) determined and ii) coherently represented using well-defined artifacts such as entity-relationship diagrams, data flow diagrams, use cases, and screen prototypes” (Yadav et al.,

2009, p. 2). The RA phase presents some of the more significant project-related risks in offshore software development (Anudhe & Matthew, 2009).

Users often cannot identify complete and clear requirements upfront. Further, their requirements tend to evolve over time, which results in sub-optimal communication between the client and development teams (Crowston & Kammerer, 1998). Changing requirements can also infuse the RA process with confusion and, thus, make interpreting requirements across multiple user groups more laborious. Collectively, these factors can result in imprecise and ambiguous system documentation. Therefore, in the RA phase, analysts and users must communicate frequently and precisely to ensure accurate and complete requirements. Numerous authors have proposed that requirements elicitation is a process in and of itself. Effective RA practices require actively considering and supporting processes that are shaped by the complexity of system and representation of requirements (Nguyen & Swatman, 2003). Ramasubbu, Mithas, Krishnan, and Kemerer (2008) suggest that one should view offshore requirements gathering through the lens of process maturity. To this end, research has presented several perspectives of RA processes.

One can define offshore RA as a knowledge-acquisition and knowledge-sharing process that enables one to explore stakeholders' needs, the application domain, and technical solutions. These broad aspects of RA include information gathering, representation, verification (Browne & Ramesh, 2002), requirements discovery, classification, organization, prioritization, negotiation, documentation (Sommerville & Sawyer, 1998), assessing, auditing, and comparing the effectiveness of requirements gathering (Ramasubbu et al., 2008). RA processes are inherently iterative to allow one to reshape stakeholders' understanding and minimize client bias stemming from misinformation that analysts introduce (Appan & Browne, 2012). Firms use various artifacts and deliverables to support the RA process. Most commonly, these include requirements modeling such as context diagrams, data flow diagrams, entity relationship diagrams, process specifications, and wire frames or working prototypes (Nath et al., 2008). Project planning during RA phase may include communication and coordination plans, weekly status reports, contingency plans, and closing documents. Using experiments, Nath et al. (2008) and Yadav et al. (2009) report on firms that successfully offshored their entire RA phase, including all the RA artifacts and deliverables mentioned earlier.

To address offshore project challenges, firms have traditionally executed requirements-related interactions between analysts and users during early phases of the project in co-located mode wherein analyst and user teams reside near each other physically and meet face-to-face to gather and sort out requirements. However, in light of improved synchronous technologies, cost-cutting measures, and improved process management, software companies have increasingly explored offshoring requirements-related activities. In a study of the Indian software industry, Yadav (2008) found that studied firms offshored 25 to 75 percent of requirements gathering phase for projects having good client-vendor relationships or for small projects that involved expensive overheads to travel to client locations. Firms also often offshored RA when requirements captured in co-located mode required subsequent changes or enhancements offshore. By partially offshoring RA phases, firms can mitigate the high costs of keeping analyst teams at client locations rather than at offshore locations, which creates a stronger incentive to offshore earlier phases.

2.2 Control Theory in Offshore Projects and in RA

Past research has relied on control theory to better understand how firms can better manage software development projects. Our own findings are also rooted in control theory for several reasons (see last paragraph in this section). Control theory helps to explain how a person or group (controller) ensures that another person or group (controllee) works toward attaining a set of organizational goals. The theory proposes four fundamental modes of control: outcome control, behavior control, clan control, and self-control (Ouchi, 1979; Eisenhardt, 1985).

One can classify behavior control and outcome control as formal modes control (Eisenhardt, 1985; Kirsch, Sambamurthy, Ko, & Purvis, 2002). In behavior control, controllers define appropriate steps and procedures for task performance. Controllers evaluate controllees' performance based on whether they adhere to these prescribed procedures. In the case of outcome control, the controllers define appropriate targets (Kirsch et al., 2002). Controllees can decide how to meet those output targets. Controllers evaluate controllees' performance based on the degree to which the latter met set targets. Controllees' performance evaluations do not depend on the processes used to achieve the targets. Eisenhardt (1985)

suggests that behavior-based controls are more appropriate for programmed tasks and outcome-based controls are more appropriate for less programmed tasks.

Control literature also suggests two modes of informal control, self and clan (Kirsch, 1997) that are based on social or people strategies (Eisenhardt, 1985; Jaworski, 1988). Since it is difficult to create shared goals and shared beliefs in an offshored context (Lacity & Willcocks, 2001), clan control is more difficult to implement (Choudhury & Sabherwal, 2003). Further, self-control is initiated and implemented internally by the providers and resembles internal projects (Choudhury & Sabherwal, 2003). Clients can play an important role in promoting informal modes of controls in offshore projects. Clan control has a positive impact on project performance and is more difficult to promote than self-control in client-provider relationships (Wiener, Remus, Heumann, & Mähring, 2014). Both academic (Kirsch & Cummings, 1996) and practitioner (Project Management Institute, 2013) literature suggests that one can use both formal and informal controls to effectively manage projects.

Primarily, two factors suggest control theory's relevance to our study. First, because the RA phase is one of the earlier software development phases, team relationship structures are still evolving. Formal and informal controls can expedite the development of such structures and working relationships. Kirsch et al. (2002) extend control theory to include the role of client liaisons/coordinators that exercise control of IS project leaders to ensure that IS projects meet their goals. Lee et al. (2006) also propose assigning point persons or coordinators to offshore sites to effectively manage globally distributed software projects. Control mechanisms, in such cases, can be formal or informal: management documents and initiates the former, and employees initiate the latter. Second, although true for all phases of offshored software projects, formal modes of control and management can be critical during the RA phase because artifacts of this phase determine success in the later stages of lifecycle to a large extent. Gopal and Gosain (2010) suggest that effective liaisons, boundary objects, and interaction processes at the interface between client and provider firms are critical for offshore projects and must be present on a continuous basis. The presence of liaisons and interaction processes in offshore sites help ensure that one fine-tunes control to the dynamic contextual conditions. Control exercised by client-site coordinators will likely have a positive impact on RA success (Yadav et al., 2013). Prior literature predominantly suggests using formal modes of control in offshore IS projects. Therefore, in this paper, we examine only formal modes of control in offshore RA. However, the role of informal control in offshore projects cannot be ruled out, and we need to better understand informal modes of control even in a highly interactive RA phase.

3 Multi-method Research Design

3.1 Overview of Research Process

Brewer and Hunter (2006) assert that a multi-method research strategy is simple but powerful. They argue that individual methods have weaknesses unique to them and that researchers can accept convergent findings from multiple methods with far greater confidence than those from any one method. Jarvenpaa (1988) also recommends using multiple research methodologies to effectively test the predictive ability of research results and to avoid biases due to the methods used. Acknowledging these strengths of multi-method research, in this paper, we integrate and present findings from multiple studies to advance our understanding of the issues that concern offshoring the RA phase of software projects. The prescriptions and findings we present in this study are based on over six years of research (2002-08) that comprised interviews of information systems (IS) professionals, two quasi-experiments carried out with student teams in the US and India, and an industry survey of IT managers in India.

Figure 1 overviews our research design. First, we engaged in exploratory research that entailed a systematic literature review and informal interviews with IT service providers in India. In this exploratory phase, we assessed the current state of RA offshoring, identified research gaps, and developed research questions to deepen knowledge in this space. Subsequently, we tested our developed research questions in a controlled setting using two academic quasi-experiments (study 1 and study 2). Next, we further tested these research questions in a field setting using an industry survey (study 3). Figure 2 illustrates the variables of interest for our research model. We briefly describe each phase in the research design in Sections 3.2 to 3.6.

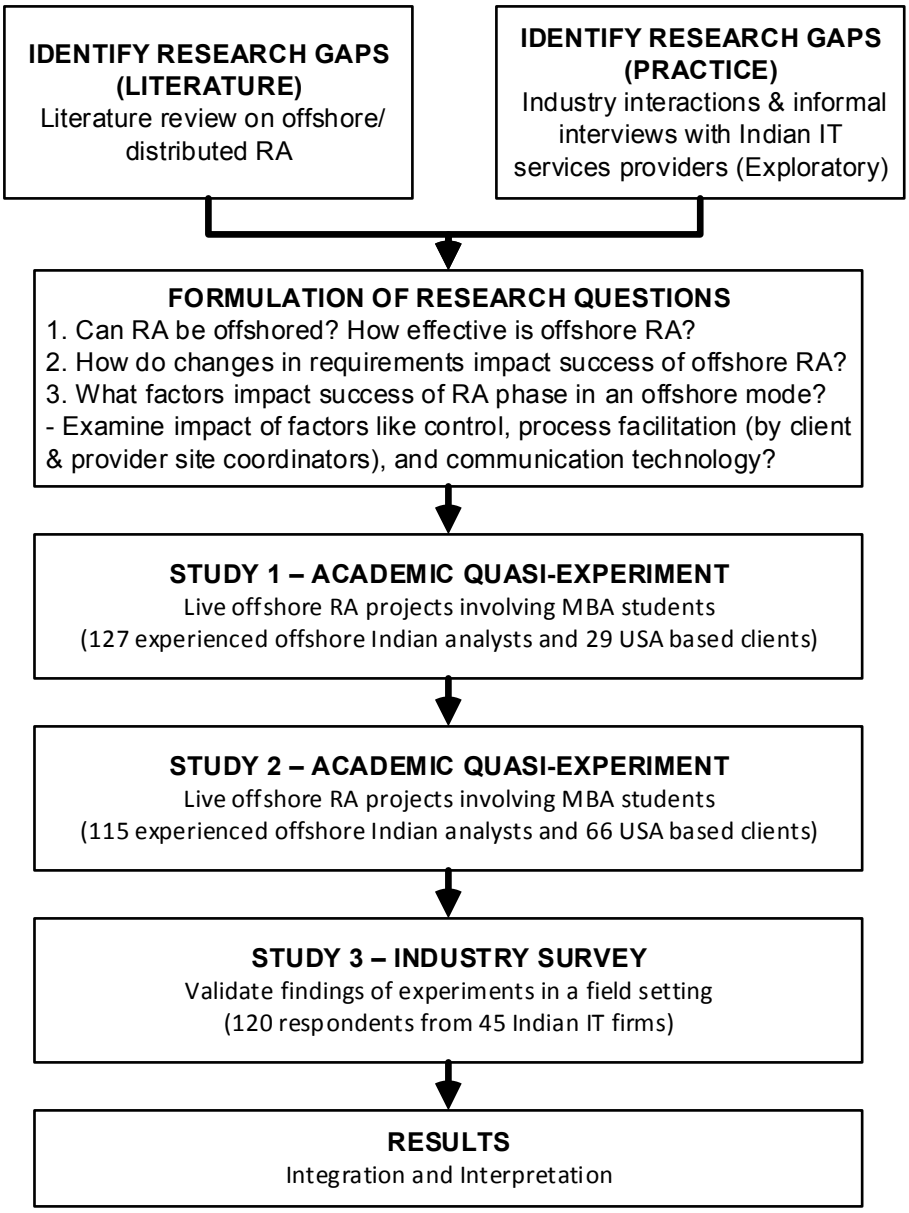

Figure 1. Research Design for Multi-method Study

3.2 Findings from Literature Review

In systematically reviewing the literature, we identified several themes that formed effective research areas for this domain. We examined all literature related to requirements analysis, requirements gathering, and requirements elicitation. We focused, in particular, on those related to offshored projects. Given the relatively recent maturation of technologies and processes to effectively support RA processes, we unsurprisingly did not find many studies on offshoring RA. We searched EBSCO, Proquest, and Google Scholar for papers using the keywords “requirements analysis”, “requirements engineering”, “requirements gathering”, and “offshore”. In Section 3.2, we extend Yadav’s (2011) review on distributed requirements engineering.

The review highlights the paucity of literature on offshore/distributed RA and the need for extensive research in this area. Key themes identified for future research include measuring the success of offshore RA, control’s impact on RA in offshore projects, the role of tools and communication technology in offshore projects and RA, the adoption of flexibility to address distribution challenges in offshore RA, RA’s relational aspects in offshore mode, and the impact of requirements change in offshore RA. We cover the relevance of formal and informal control modes to offshore RA according to control theory in detail in

Section 2.2. We discuss the other identified research themes in Sections 3.2.1 to 3.2.5, and Table 1 summarizes them (see Section 3.2.5).

3.2.1 Offshore RA Success

When offshoring RA, one needs to consider how to effectively measure its success. Research has examined project success as an aggregate of two or more of the following factors: system quality, information quality, user satisfaction, service quality, use, and net benefits (Delone & McLean, 2003, 1992). Mahaney and Lederer (2006) identify three dimensions of software project success: client satisfaction, perceived quality of the project, and success with the implementation process. Similarly, Baroudi and Orlikowski (1988) develop a short-form measure of user information satisfaction (UIS) that measures the success or effectiveness of a management information system (MIS). Thus, in essence, IS research has largely converged on defining project success as client or user satisfaction with the project outcome (i.e., the end product of software development and underlying processes) (Delisle, 2001). All these success measures focus on the end-result of the collective phases of software development lifecycle i.e. the end product.

However, considering that RA lays the groundwork for success in later phases, one needs phase-based success measures to determine the effectiveness of offshoring that phase. Ensuring this phase's success could potentially mitigate later risks and issues. However, researchers have only recently started measuring the success of a specific phase of the software development lifecycle. For instance, Yadav et al. (2009) and Yadav et al. (2013) adapt and extend the IS project success measures from literature to measure the RA phase in offshore projects. These measures define offshore RA success in terms of 1) client satisfaction with the RA phase, 2) perceived quality of the RA deliverables, and 3) perceived success of the RA process. We use these factors to measure the dependent variable RA success in our experiment (study 2) and industry survey (study 3).

3.2.2 Relational Aspects of RA

Many researchers have attributed poor requirements elaboration to provider limitations (Levina & Vaast, 2005). To capture requirements correctly, service providers can encourage clients to more collaboratively conduct the requirements-gathering phase (Anudhe & Matthew, 2009) to build trust and a shared understanding. Trust (Edwards & Sridhar, 2005), shared goals, culture, processes, and responsibility improve requirements' success. A shared culture among offshore teams shapes expectations about teams' tasks and relationships and provides context to knowledge sharing among such remote teams (Evaristo et al., 2005). Asymmetries in knowledge and experience across these teams prompt onsite and offshore teams to engage in acts of sense giving, sense demanding, and sense breaking (Vlaar, van Fenema, & Tiwari, 2008). Such processes enhance the likelihood of producing consistent and actionable set of requirements in an offshore engagement.

3.2.3 Role of Tools and Technology in Offshore RA

Studies have established that the right tools and technologies can be significantly beneficial for global collaboration in requirements management (Sinha, Sengupta, & Chandra, 2006). In particular, studies have demonstrated that the quality of solutions and the creativity of computer-supported conferencing groups were marginally higher than software groups that met face-to-face (Ocker et al., 1995). In requirements negotiation, for instance, groups meeting face-to-face performed no better than computer-supported groups (Damian, Eberlein, Shaw, & Gaines, 2000). As such, numerous studies have called for groupware support in requirements meetings and analysis (Boehm, Grünbacher, & Briggs, 2001; Damian, 2007). Studies have also found that using uninterrupted dedicated phone lines and meeting recordings for re-listening and clarification help support the iterative process of requirements discovery, sharing, and reformation (Akbar & Hassan, 2010). As software teams delve deeper into requirements analysis, virtual prototypes have also been effective in bringing offshore and onsite teams to a shared vision about project outcomes more rapidly than traditional techniques (Vlaar et al., 2008).

3.2.4 Using Agile Methods in Offshore Projects and RA

Studies have found agile processes to improve communication and coordination and to mitigate control problems across offshore teams (Holstrom, Fitzgerald, Agerfalk, & Conchuir, 2006; Lee, Delone, & Espinosa, 2006). The agile practices of daily stand-up meetings, users' involvement in the core agile team, and strong focus on customer collaboration can also benefit offshoring RA and managing

requirements change. Studies have found agile approaches to improve teams' cohesion and morale such that individuals relay critical information more readily (Cockburn & Highsmith, 2001) as opposed to when they do not follow such practices. However, research offers mixed evidence on the successful use of agile practices with globally distributed teams. For instance, in a study of 12 distributed agile software development projects, Bose (2008) found that distributed teams were likely to face difficulties related to communication, time zone differences, the building of trust in relationships and work-culture, and knowledge management. Similarly, Ramesh, Mohan, & Cao (2012) found that use of agile methods in distributed software development process posed conflicting demands of alignment and adaptability in the context of ambidexterity in a multi-site case study of three agile projects. In a study of 66 projects spanning Europe, Asia, and the Americas, Estler, Nordio, Furia, Meyer, and Schneider (2014) compared and contrasted agile processes such as scrum and extreme programming (XP) with structured processes such as the rational unified process (RUP) and waterfall. They examined the impact of these methodologies on outcome variables such as success and economic gains, team motivation, and communication requirements during distributed development. The authors found no significant difference between the outcomes for distributed projects that adopted agile processes and those that adopted the structured processes.

Carefully incorporating agility in offshore software development can potentially address communication-, control-, and trust-related challenges across distributed teams (Ramesh, Cao, Mohan, & Xu, 2006). In a study of a large distributed project experiencing unanticipated changes, Batra, Xia, VanderMeer, & Dutta, (2010) found a balance between agile and traditional plan-based methods to be feasible. The authors suggest that both control and agility were beneficial for meeting challenges of a distributed project. For instance, Korkala and Maurer (2014) found that using waste as a lens to detect non-value adding communication elements in globally distributed agile projects was beneficial. Focusing on artefacts, Bass (2016) studied how firms tailor development processes in large-scale offshore software development projects. These studies reporting applications of agile principles in offshore projects offer a fresh perspective on the feasibility of using flexibility as a potential approach to mitigate the challenges of distribution. This literature also highlights a need to further examine best practices around using agile approaches throughout the entire development lifecycle, including the RA phase, in offshore projects (Yadav, 2011; Yadav et al., 2013).

3.2.5 Requirements Changes in Offshore Projects

Firms need to manage requirements in offshore software development projects (Yadav, 2011; Ali & Lai, 2016), but few studies have examined the impact of requirements change on a project's success or failure (Yadav et al., 2013). Yadav et al. (2009) report that team members who experienced requirements changes in offshore RA also experienced greater control, computer-mediated communication, and process facilitation by vendor site-coordinators. Ali and Lai (2016) propose a three-stage requirements-management process for globally distributed software projects using a case study of an online shopping system to minimize risks in a globally distributed environment.

Table 1. Key Themes/Findings from the Offshore/Distributed Requirements Literature

Role of control in offshore projects and RA	
Formal modes of control (namely, outcome and behavior controls) dominate portfolios of control in outsourced projects.	Choudhury & Sabherwal (2003)
Knowledge of the systems development process is a key antecedent of formal modes of control. It also plays a key role in the choice of informal control mode (clan and self-control). Client liaisons/coordinators exercise control of IS project leaders to ensure that IS projects meet their goals.	Kirsch et al. (2002) Kirsch (2004)
Formal and informal control can have a significant impact on software projects' outcomes provided they are fine-tuned and directed towards project objectives. Effective liaisons, boundary objects, and interaction processes at the interface between client and provider firms are critical for offshore projects. These liaisons must be present on a continuous basis to ensure that one fine-tunes control to one's dynamic contextual conditions.	Gopal & Gosain (2010)
Offshore projects predominantly use formal controls, and trust has a negative effect on the amount of formal control. Projects with high task complexity likely have high levels of control.	Remus & Wiener (2012)
Control by client-site coordinators has a positive impact on RA success and process facilitation by client site-coordinators affects RA success indirectly through control.	Yadav et al. (2013)

Table 1. Key Themes/Findings from the Offshore/Distributed Requirements Literature

Clients play an important role in promoting informal controls in offshore projects. Clan control is more difficult to promote than self-control in client-provider relationships. Clan control has a direct positive impact on project performance.	Wiener et al. (2014)
Offshore RA Success	
Developed measures to determine the success of offshored RA phase by adapting project success measures from Delone and McLean (1992, 2003) and Mahaney and Lederer, (2006).	Nath et al. (2008), Yadav et al. (2009), Yadav et al. (2013)
Task and relational aspects of offshore projects and RA	
Trust improves requirements success.	Bhat et al. (2006), Edwards & Sridhar (2005)
Shared goals, culture, processes, and responsibility improves requirements success.	Bhat et al. (2006)
Culture shapes expectations about teams' task and orientation and about context sharing.	Evaristo et al. (2005)
Knowledge and experience asymmetries and requirements and task characteristics (complexity, instability, ambiguity, and novelty) prompt sense making actions.	Vlaar et al. (2008)
"Straddlers" can assist in transferring codified knowledge.	Mattarelli & Gupta (2009)
Role of technology and tools in offshore RA	
Tools are beneficial for global collaboration in requirements management.	Sinha et al. (2006)
Groups meeting face-to-face perform no better than those using video conferencing and computer support in negotiating requirements.	Damian et al. (2000)
Computer-conferencing groups had marginally better solutions and creativity than face-to-face groups.	Ocker et al. (1995)
Firms must provide groupware support for requirements meetings and analysis.	Damian (2002), Boehm et al. (2001)
Virtual prototypes can be effective in bringing offshore and onsite teams to a common ground more rapidly than traditional techniques.	Vlaars et al. (2008)
IS managers show an increasing interest in off-shoring RA phase using computer-mediated communication, and the deliverable quality in pure off-shore mode is comparable to that of RA in collocated mode.	Nath et al. (2008)
Research has also found using uninterrupted dedicated phone line and, meeting recordings for re-listening and clarification to be beneficial in supporting the iterative process of requirements discovery, sharing, and reformation.	Akbar & Hassan (2010)
Use of agile methods in offshore/distributed projects and RA	
Agile processes improve communication, coordination, and control problems in offshore environments	Holstrom et al. (2006)
Carefully incorporating agility in offshore software development is essential in addressing communication-, control-, and trust-related challenges across distributed teams.	Ramesh et al. (2006)
One needs to modify/tailor conventional agile development approaches for offshore environments to overcome the challenges of time zone differences, geographic distance, and sociocultural differences. One needs to use ambidextrous coping strategies.	Lee et al. (2006)
Creating a flexible "agile-rigid" environment can help organizations mitigate various risks inherent in offshore RA.	Yadav et al. (2007)
Applying agile principles in globally distributed teams can be challenging. Distributed teams face challenges related to communication, time zone differences, the building of trust in relationships and work-culture, and knowledge management.	Bose (2008)
A balance between agile and traditional methods is feasible. One needs both control and agility for meeting the challenges of a distributed project.	Batra et al. (2010)
Few studies on distribution requirements engineering and agility exist.	Yadav (2011)

Table 1. Key Themes/Findings from the Offshore/Distributed Requirements Literature

Firms need contextual ambidexterity for distributed software development, which is the capability to simultaneously address the conflicting demands of alignment and adaptability.	Ramesh et al. (2012)
No significant difference between the outcomes of distributed projects that adopt agile processes versus the distributed projects that adopt the structured processes.	Estler et al. (2014)
Use waste as a lens to detect non-value-adding communication elements in globally distributed agile projects.	Korkala et al. (2014)
Requirements change in offshore/ globally distributed projects	
Team members with changing requirements also experience greater control, greater computer-mediated communication, and greater process facilitation by vendor site-coordinators in offshore RA.	Yadav et al. (2009)
Requirements change in early stages such as RA can help in improving clarity on offshore project outcomes.	Yadav et al. (2013)
One needs to manage changes in requirements at the right time to minimize risks in a globally distributed software project.	Ali & Lai (2016)

3.3 Exploratory Study: Industry Interactions and Interviews

With this exploratory research, we provide insights and help explain the research problem. We began formulating research questions around offshore RA by simultaneously examining gaps in literature and by engaging in discussions with senior managers from the Indian IT industry with experience in offshore projects. We conducted open-ended interviews with 15 such managers from six provider firms: Tata Consultancy Services (TCS), NIIT Technologies, Sapient, IBM, Hewitt Associates, and Aricent. With the industry interviews, we better prioritized the research themes and findings that we list in Table 1. They also facilitated our formulating research questions centered on constructs of practitioner and academic relevance. In particular, we identified three research questions:

RQ1: Can one offshore the RA phase? How effective is RA with offshore virtual teams when compared to RA with traditional, co-located teams?

RQ2: How do changes in requirements impact success of the RA phase when conducted in an offshore mode?

RQ3: What factors impact the success of the RA phase in offshore projects?

We explored RQ1 in the exploratory study and study 1, and we examined RQ2 and RQ3 with studies 2 and 3.Figure 2 presents the research model showing variables of research interest that we examined across the multiple studies. Specifically, we studied the impact of factors (namely, control, process facilitation by client and provider-site coordinators, changing requirements, and computer-mediated communication) on the success of the RA phase carried out offshore.

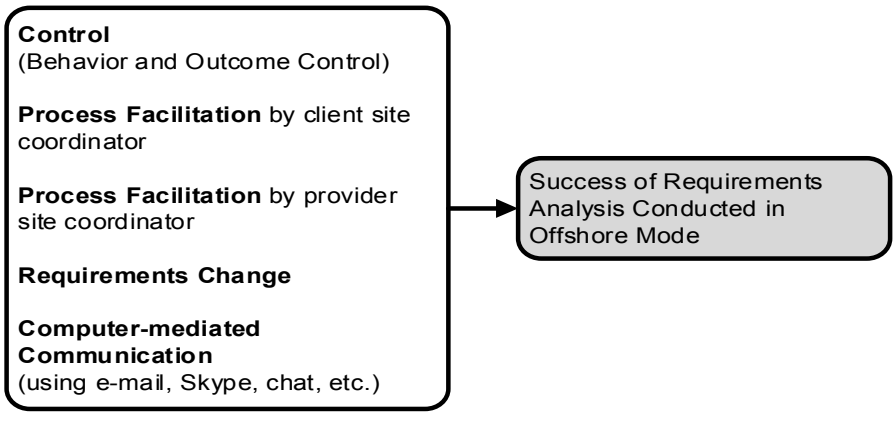

Figure 2. Research Model

3.4 Quasi-experiments (Studies 1 and 2)

During the industry interviews described earlier, managers indicated that they only rarely completely offshored requirements engineering. However, for projects related to enhancing existing solutions and those with modest scope and size, the managers indicated they offshored about 25 to 75 percent of the requirements phase (Yadav, 2009). For this reason, in our studies, we could not analyze the "total" offshoring of requirements analysis on live projects. To counter these challenges, we designed quasi-experimental research studies in an academic setting. A quasi-experimental setting has treatments, outcome measures, and experimental units but does not have random assignment of subjects as is the case in pure experiments (Campbell & Stanley, 1963). Further, the researcher might not have full control over the experimental setting. We refer to the quasi-experiment as "the experiment". We describe these experiments in brief here. Readers can find further details in Nath et al. (2008) and Yadav et al. (2009).

3.4.1 Quasi-experiment (Study 1): Offshore RA Success

In study 1, we explored RQ1. Specifically, we tested the following hypothesis:

H1: Collocated teams using face-to-face communication produce higher quality RA deliverables compared to virtual teams using computer-mediated communication during the requirements analysis phase of software projects.

We performed study 1 with student participants from two academic institutions (one in India and the other in the US). In India, 127 graduate students enrolled in a management information systems (MIS) course in the MBA program at MDI, Gurgaon India acted as IS provider team members. In the US, 29 graduate students enrolled in the MBA program at Marquette University, USA acted as project managers. Both sets of participants had two to five years' prior experience. Indian participants primarily had work experience in the software industry. As such, participant profiles were similar to those of early-career client and provider team members.

Half of the participants from India functioned as software developers and half as their co-located client teams. The U.S. teams were remote client teams. This arrangement created a total of 10 co-located and 10 offshore teams (see Figure 3). Both offshore and co-located teams had similarly scoped software projects to execute. The projects were real-world as they related to developing software solutions for non-profit and small for-profit organizations.

The development teams submitted the following RA artifacts to both co-located and offshore clients: 1) a project plan at start of the project, 2) a vision document describing the high-level purpose and scope of the product and project, 3) process models, 4) detailed process specifications, 5) data models, and 6) a screen-based prototype using wire-frames. We expected no working prototypes. The development teams communicated with offshore clients through lean technologies such as email, instant messaging, and voice chats (e.g., Skype) and with their co-located teams through face-to-face meetings.

We assessed the RA phase's success in study 1 using measures for completeness and adherence to user requirements, consistency of RA artifacts, and users' perceived quality of the RA deliverables. For example, we asked the offshore and co-located clients to rate the RA phase's success based on how closely the final artifacts that the developer teams submitted reflected client requirements. For further validation, we invited experts with two to three years' software development experience to review and evaluate the completeness and adherence of project artifacts to industry standards and to assess consistency of project artifacts. The experts assessed artifact consistency between submitted screenshots and process and data flow diagrams submitted as part of analysis and design phases. The experts evaluated both completeness and consistency on a seven-point Likert scale. The results revealed no significant difference in the quality of RA deliverables produced by virtual teams using computer-mediated communication and that of collocated teams using rich face-to-face communications. Table 3 presents the results (we discuss these results more in Section 3.6).

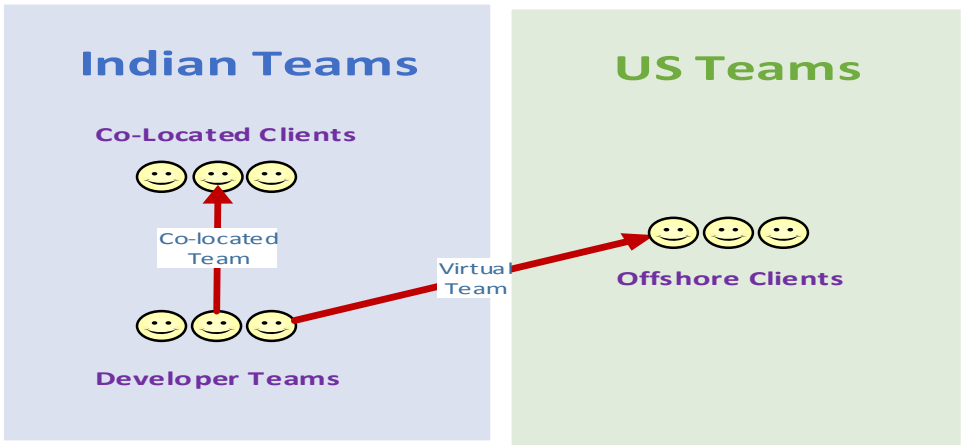

Figure 3. Offshore and Co-located Team Setting for Studying RA Success

3.4.2 Quasi-experiment (Study 2): Offshore RA Success with Changing User Requirements

Irrefutably, user requirements are often unclear and tend to evolve progressively through the project. The senior managers also identified these issues. However, are changes in requirements beneficial or detrimental to the RA phase's success, and what factors would impact the success of the RA phase, especially when conducted in an offshore mode with globally dispersed virtual teams (i.e., RQ2 and 3)? To address these questions, we conducted a second set of quasi-experiment with Indian and U.S. student teams. We paired 20 teams (total 181 participants) including 115 analysts in India with 66 clients in the US. Specifically, we examined the impact of factors (requirements change, facilitation by provider and client site-coordinators, control, and computer-mediated communication) on the success of the RA phase conducted offshore.

Figure 4 illustrates the setup for this study. The clients were students in Marquette University, USA, enrolled for a course in IT project management. The analysts were graduate students enrolled in the management information systems (MIS) course of the MBA program at MDI Gurgaon, India. This live experiment lasted for eight weeks and comprised a part of the course project. Both sets of participants had one to three years' prior experience. Indian participants primarily had work experience in the software industry. Participant profiles were similar to that of early-career client and provider team members. Experiment participants developed offshore RA deliverables iteratively using the structured software development methodology and lean communication technologies such as chat, Skype, and email. The RA deliverables were the same as for study 1.

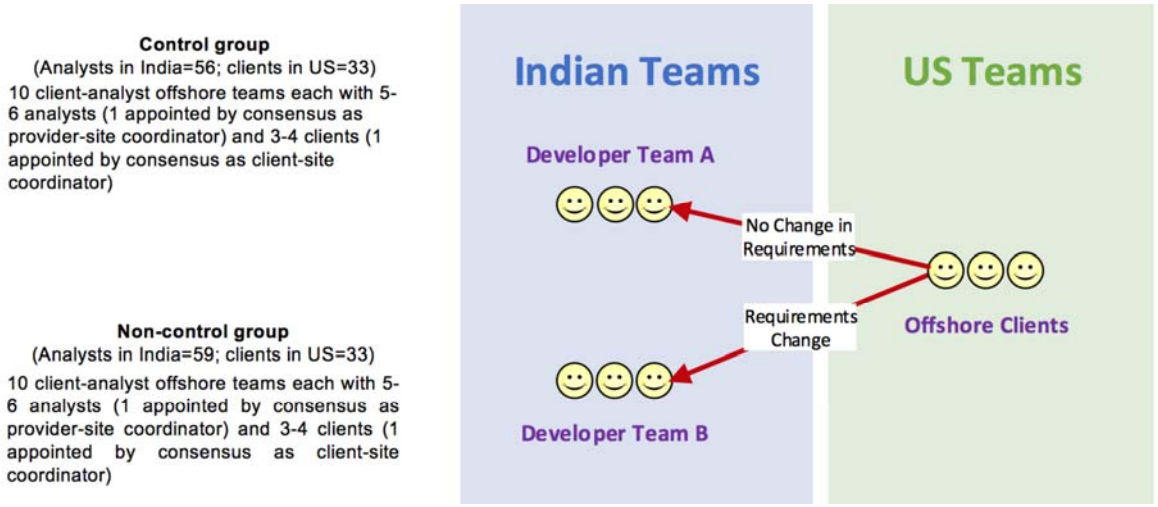

Figure 4. Experiment Setting for Studying Offshore RA Success with Changing User Requirements

U.S. client teams collaborated with two developer teams each (teams A and B) that were both based in India. Both teams had the same initial requirements, which were stable throughout the project for team A. In contrast, requirements for team B changed midway through the project. Based on the RQ2 and RQ3, we developed the following hypotheses:

H2: Changes in requirements impacts offshore RA success.

H3: Formal modes of control positively impact offshore RA success.

H4: Process facilitation by client site-coordinators positively impacts offshore RA success.

H5: Process facilitation by provider site-coordinators positively impacts offshore RA success.

H6: Task-related computer-mediated communication positively impacts offshore RA success.

At the end of the quasi-experiment, we administered a survey questionnaire to collect data from client and provider team members. We developed the survey instrument using measures adapted from prior literature. Yadav et al. (2009) present details of the instrument measures we used in the experiment for control (behavior and outcome), facilitation (by provider and client coordinators), computer-mediated communication and RA success. We later adopted the same questionnaire was for study 3 (see Appendix A). We used structural equation modeling (SEM) using AMOS version 7 for analysis.

Measures for the dependent variable RA success included 1) client satisfaction with the RA phase, 2) perceived quality of the RA deliverables, and 3) perceived success of the RA process (Yadav et al., 2009). We measured clients' satisfaction in terms of how involved and committed provider analysts were with the clients during the offshore RA phase; clients' comprehension of the RA deliverables that offshore analysts submitted; offshore analysts' willingness and commitment towards completing the RA phase's goals and tasks; and clients' acceptance of final RA phase's deliverables (such as context analysis diagrams, data flow diagrams, process specifications, entity relationship diagrams, and screen-based prototypes/screenshots). We measured perceived quality in terms of how well the final RA deliverables clearly specified and captured client requirements. We measured perceived success of the offshore RA process in terms of how well analysts completed the RA phase in its planned schedule and how satisfied clients were with the entire offshore RA process. We measured requirements change using multi-group analysis that included testing for differences in the latent means. Here, we introduced requirements change as a treatment in the experiment (study 2) and tested whether team members whose requirements changed had higher perceptions of the latent variables (H3-6) than team members who had no changes in requirements (Yadav et al., 2009).

Study 2's findings reveal that requirements change, formal control, and presence of a client-site coordinator had a positive impact on offshore RA success. The experimental treatment (requirements changes) in the non-control group led to increased interaction between clients and provider analysts, which led to better understanding of client requirements in the non-control group. However, we did not find support for H5 (provider-site coordinator's positive impact on offshore RA success) and H6 (computer-mediated communication's positive affect on offshore RA success). While provider teams have not typically viewed requirements changes positively in traditional settings, agile approaches encourage teams to iteratively elaborate on requirements to elicit clarity in user needs. Further, such elaboration enables client teams to reconsider and, perhaps, enhance the specifications provided during initial requirements gathering. As such, one can expect that progressive elaboration and related changes will bring the final project artifacts closer to user expectations and, thereby, result in greater client satisfaction, but the matter requires further investigation.

3.5 Industry Survey (Study 3)

Though one would ideally conduct a field experiment to help validate findings from quasi-experiments, in reality, field experiments focusing on the requirements phase have many risks. This phase involves high customer interaction and has much client-related information. For this reason, the IT industry experts indicated field experiments as an infeasible alternative during our exploratory industry interviews. As such, an industry survey based on findings from the experiments offered a pragmatic alternative to validate our research findings. For this survey, we surveyed 115 representatives from 45 provider firms in major Indian cities (New Delhi, Gurgaon, Noida, Bangalore, Hyderabad, and Pune). These firms ranged in size and capabilities (examples include TCS, HCL Technologies, Avaya Global, and Wipro Technologies). Non-disclosure agreements with provider firms limited our ability to gather demographic-, project-, or client-

related information from respondents. Survey respondents were mainly IT managers in client-facing leadership roles.

We examined RQ2 and RQ3 in the industry survey from the perspective of Indian providers. As in study 2, we tested H2-H5. The unit of analysis was the individual with reference to a specific project. We adapted and used the same measures that we used for study 2. Since we analyzed requirements change in study 2 using multi-group analysis, we developed measures for requirements change for study 3. For analysis, we used structural equation modeling (SEM) using AMOS version 7. Yadav et al. (2013) provide details of the industry survey, lists of participating organizations, and detailed analysis of the results and survey instrument. Table 2 summarizes key highlights of the industry survey, and Appendix A provides the questionnaire items.

Table 2. Summary of Industry Survey Results (Yadav et al., 2013)

Hypothesis path	Hypothesized relationship	Path coefficient (std. beta)	Critical ratio (CR > ±1.96)	Sig. (p < .05)	Hypothesis supported?
Control → req. analysis success	+	0.849*	4.291*	.000*	**Supported**
Process facilitation (provider) → req. analysis success	+	0.108	0.885	.376	Not Supported
Process facilitation (client) → req. analysis success	+	-0.123	-0.811	.417	**Supported indirectly** (indirect effect on success mediated via control)
Process facilitation (provider) → Control	+	0.046	0.329	.742	Not supported
Process facilitation (client) → Control	+	0.501*	3.668*	.000*	**Supported**
Req. change → req. analysis success	-/+	-0.185	-1.874	.061	Not supported

The results in Table 2 indicate control's (including both behavior and outcome) strong positive impact on RA success in offshore projects. Further, the presence of client site-coordinators had an indirect positive impact via control on RA success, but provider site-coordinators did not show any influence. From the perspective of providers, changes in requirements did not have an impact on RA success. Possibly, clients better received changing requirements as indicating improved clarity on requirements. However, as we did not explore the client perspective in our study, this finding requires further examination.

3.6 Integration of Results

Creswell (2003) suggests that one can integrate data from one stage with data from other stages in the process of research (i.e. data collection, data analysis, interpretation, and some combination of these). In this study, we integrated data from the experiments with data from the industry survey at the end when we interpreted our results to assimilate findings and propose key lessons learned. Table 3 revisits our research questions and summarizes findings across the experimental and industry studies.

Table 3. Summary of Integrated Findings on RA Success in offshore projects

Research question	Hypothesis	Experimental findings	Survey findings
RQ1: Can one offshore the RA phase? How effective is RA with offshore virtual teams when compared to RA with traditional, co-located teams?	Requirements analysis phase of the project in virtual mode is as effective as in co-located mode	√	√
RQ2: How do changes in requirements impact success of the RA phase when conducted in an offshore mode?	Changes in requirements positively impact offshore RA success	√	-
RQ3: What factors impact the success of the RA phase in offshore projects?	Formal modes of control positively impact offshore RA success	√	√
	Presence of client site-coordinator positively impacts offshore RA success	√	√ (indirect effect)

To draw holistic insights, we compared results of the literature review, exploratory study, experiments, and the industry survey and examined potential implications for the software development community. Notably, such integration can potentially inform both academia and practice holistically regarding RA challenges, opportunities, and best practices. We discuss these aspects in Section 4 as lessons learned from our multi-method research.

4 Lessons Learned: Managing the Effectiveness of Offshore Requirements Analysis

4.1 Lesson 1

Lesson 1: One can successfully execute RA offshore.

Whether one can conduct requirements gathering offshore has great significance to both client and provider firms. Provider firms were particularly interested in whether results suggested difference in the effectiveness between offshore and co-located teams. At the most fundamental level, insignificant difference between the two could lead firms to repatriate large, expensive teams of on-site analysts and, thus, yield significant cost efficiencies to both client and provider firms.

Table 4 (based on study 1) provides the mean scores on the project-success measures we discuss in Section 3.4.1 for co-located and offshore teams. The analysis of variance test results suggest teams can successfully conduct requirements gathering and analysis in both offshore and onshore mode (Nath et al., 2008). Further examining the findings from the industry survey, however, revealed some additional insights that the experiments did not (see Section 4.1.1).

Table 4. Requirements Analysis Success for Co-located and Offshore teams

RA success factors (measured on a 1-7 scale)	Mean (collocated teams)	Mean (offshore teams)	F-value (significance)
User perceived quality	4.91	4.75	0.616 (0.435)
Completeness and adherence of project artifacts	4.92	4.56	0.551(0.467)
Consistency of project artifacts	6.32	6.51	1.025(0.323)

4.1.1 Lesson 1b

Lesson 1a: Requirements analysis is an embedded process in the software engineering cycle. Firms should find a common framework to align these processes across client and provider teams.

Consistent with past literature, the findings from our experiments and interviews with industry experts also suggest that RA is an embedded process in the software engineering development cycle. An effective RA process requires managers to actively consider the complexity of the IS system as the accurate representation of user requirements help shape it. To successfully conduct RA offshore, the process necessitates adjustments in light of globally dispersed client and provider teams. Most fundamentally, such teams must first identify and then align their RA processes. In both experimental studies 1 and 2, we prescribed well-defined RA deliverables, which the analysts created totally in offshore mode. Hence, to offshore RA, firms must clearly specify expected RA deliverables and build them iteratively. For clear specification of expected RA deliverables, teams must either develop or agree on a common framework. Well-accepted and benchmarked frameworks such as the capability maturity model (CMM) can provide such a starting point. For instance, adherence to CMM processes and learning-based investments in offshore software teams can improve knowledge discovery, interpretation, and retention (Bhat et al., 2006; Ramasubbu et al., 2008). Embedding training programs, formal change management, and peer reviews in this initial knowledge-acquisition stage can enhance the quality of outcomes. Similarly, focusing on organizational processes and using peer reviews can be useful in interpreting this acquired knowledge.

4.2 Lesson 2

Lesson 2: Offshore client and provider teams must deliberate on mechanisms for improving requirements richness.

Surveys of the software professionals involved in evaluating the project artifacts for study 1 suggested that requirements gathered from offshore teams were not as rich as those gathered from co-located teams. However, they found such remotely gathered requirements sufficient in terms of feasibility, completeness, adherence to user requirements, and user satisfaction. Further, we found offshore project teams to be as successful as co-located teams in producing the artifacts necessary for the requirements phase. In spite of these positive findings, the industry participants expressed discomfort about conducting the requirements phase for large projects entirely offshore possibly due to the fact that most respondents had not actually done so as part of their professional experience or the right mechanisms were not in place do to so. Lessons 2a and 2b provide more insights on the role of technology and processes in enhancing the richness of requirements elicited offshore.

4.2.1 Lesson 2a

Lesson 2a: The choice of tool and techniques to support requirements gathering and elicitation offshore requires careful consideration.

In study 1, we examined the use of computer-mediated communication technology (email, Skype, and chat). However, we did not find a positive relationship between computer-mediated communication and offshore RA success. A possible explanation for this finding could be that respondents in the experiments relied less on synchronous technologies such as Skype and mainly used asynchronous technologies such as email in their offshore RA projects. Findings from our post-industry survey interactions with respondents reveal that communication and coordination techniques using rich synchronous communication technologies in offshore RA phases are crucial in supporting an iterative understanding. For instance, best practices that help avoid misunderstanding user requirements include: processes that help provider teams accurately capture each and every major and minor details of meetings, re-consultation meetings with the client, master requirements documents and final lists for each release milestones, and formal processes for tracking and managing requirements (Akbar & Hassan, 2010). In alignment with past studies (e.g. Bhat et al., 2006), our findings suggest that having a common set of documents and templates such as requirements-specification and change-management templates can be a useful step toward developing a shared process.

4.2.2 Lesson 2b

Lesson 2b: Firms must design processes and technologies to support both task and relationship-development aspects of RA.

To better understand user requirements, the RA phase requires a certain degree of interaction between client and provider teams. We observed that the quality of the RA deliverables in our experiments were higher in groups that had more interactions. Clients and their offshore providers can enhance the quality of RA deliverables by allowing change requests and iteratively communicating more. Therefore, we suggest that firms must design processes and technologies to support both the RA phase's task and relational needs.

Past literature also recommends that such processes must support reflection and communication and provide opportunities for developing and enhancing communications and soft skills (Vlaar et al., 2008). Mattarelli and Gupta (2009) report on an organization that used "straddlers" between onsite client teams and offshore provider teams to assist in transferring codified knowledge. Such straddlers can mitigate status differences across onsite and offshore teams that harm the knowledge-sharing process. They can also foster active learning between onsite and offshore teams. Technologies such as computer-based interviewing and group-support tools for identifying differences in understanding (Vlaar et al., 2008) and resolving conflicts among stakeholders (Bhat et al., 2006) require active consideration by clients and offshore providers, especially when dealing with complex and large projects. Most significantly, while costs can be prohibitive, clients and provider teams can support face-to-face online meetings as much as possible.

4.3 Lesson 3

Lesson 3: There should be a shared understanding between onshore and offshore teams about how requirements changes should be viewed.

The impact of requirements changes on project success yielded mixed results from studies 1 and 2. However, teams in study 2 reported that requirements changes had a positive impact on offshore RA

success (Yadav, 2008). The experimental treatment (requirements changes) in the non-control group led to increased interaction between clients and provider analysts. As such, the respondents felt that they understood client requirements better.

In contrast, responses of software professionals in the industry survey (study 3) suggest some ambivalence regarding the effect of requirements change on RA success. We found that the software professions did not perceive requirements changes to have a significant impact on RA success in the industry survey. This finding may be unsurprising because developers viewed changes in requirements as adding to the complexity of the project and, hence, felt uncomfortable about handling change requests during early phases of project. The lack of client representation in the survey sample may have potentially skewed these findings as well.

However, the findings draw attention to the need to rally development teams around a common definition of RA success. On the one hand, requirements changes may better understand solution needs. On the other hand, teams may perceive them as more work, which, in turn, may have cultural and contextual implications. For instance, to a provider team from India, requirements changes leading to rework may imply that the client poorly perceives ongoing work. In contrast, from the client's perspective, requirements changes may simply represent a changing face of the organization or of the client's better understanding its users' needs. Our findings from the experiment draws forth the need to study offshore requirements changes further from a client perspective to develop a more comprehensive understanding of its impact on success in an offshore environment.

4.4 Lesson 4

Lesson 4: Offshore teams should balance control and flexibility to enhance RA success.

We considered the impact of client-driven project monitoring (formal modes of control) on requirements success as part of the experiments in study 2 and the industry survey in study 3. Participating offshore teams in the study 2 had the flexibility to decide how to monitor and control their projects. For some offshore teams, clients enforced extensive project-monitoring practices and formal control often in terms of behavioral control through formal work assignments, completion of project plans, and formal status reporting. The remaining teams used outcome controls that defined appropriate targets for the development teams but allowed them to define their own approach to accomplishing these targets. Users' extent of monitoring and control was correlated with measures of success discussed earlier.

Integrated results of the experiments and industry survey also support the applicability of control theory (and, specifically, formal modes of control) in offshore environments. Results suggest that formal modes of control including behavior and outcome control positively impact RA success (Yadav, 2008; Yadav et al., 2009; Yadav et al. 2013). However, when using control, the industry survey respondents noted that clients must clearly state expected targets. More importantly, site coordinators must rapidly convey changes in project targets to the virtual team counterpart. The literature also suggests considering agile practices for effective change management and control. Lesson 4a elaborates more on incorporating flexibility in RA.

4.4.1 Lesson 4a

Lesson 4a: Agile approaches to software development are often effective in providing the flexibility necessary for successfully gathering requirements.

Current and ongoing research continues to reveal the benefits of agile processes for RA and requirements change management. Due to shortened project lifecycle and rapid technology changes, more software development companies have begun to elaborate requirements progressively. Firms have adopted agile methods both in software engineering and project management to cater to changing requirements. While much of the software industry has been tentative about using agile practices across remotely situated client and provider teams, with current technologies and processes, one may effectively deploy these practices may to manage rapid and frequent requirement changes. Findings from the quasi-experiments that we describe in Section 3.4 suggest that one can successfully handle requirements changes even in offshore mode and, thus, open up the possibility of adopting agile/flexible approaches for iteratively eliciting requirements and using wireframes or prototypes in offshore teams (Yadav et al., 2009). As such, agile processes might effectively meet both the task and relational needs for RA success. In our exploratory study involving interviews with industry experts, we found that, in reality, offshore projects that claim to adopt agile practices are often not fully agile in nature. Rather, they are hybrid "agile-rigid"

projects that integrate agile practices with the traditional software development approach to meet distribution challenges in offshore environments (Yadav et al., 2007).

4.5 Lesson 5

Lesson 5: Client-site coordinators that facilitate RA processes more significantly impact project success than provider-site coordinators.

Our study also supports extending control theory to include the role of site coordinators for process facilitation. Such facilitation helps in creating formal structures for better coordination and communication in offshore environments. Site coordinators or facilitators play a crucial role in aligning and controlling processes and communications between onshore and offshore teams. In study 2, all offshore teams had facilitators appointed at the client (US) and provider (India) sites (see Figure 4). Considering that multiple users often present numerous and possibly conflicting perspectives on requirements that can confuse developers, client-site coordinators can clarify and streamline requirements discrepancies. Such coordinators can also apportion work to developers and, thereby, further reduce confusion. Findings from our experiment suggest that the presence and active control of a client-site coordinator has a direct positive impact on the RA phase's success. However, increased process facilitation by provider site-coordinators only had an indirect effect on RA success.

The industry survey did not provide support for the direct effect of site coordinator's facilitating and controlling processes on RA success. However, the survey provided indirect evidence. Specifically, respondents suggested that that presence of client-site coordinators improves project monitoring and control. Since improved control positively impacts RA success (as we discuss in Lesson 4), client-site coordinators can enhance RA success through more effective monitoring and control. Survey respondents also indicated that their offshore projects had provider-site coordinators by default. Project managers usually take the role of development site-coordinators in project-governance structure.

5 Recommendations for Practice

5.1 Lesson 6

Lesson 6: Cost arbitrage is not yet a thing of the past.

The first and most important finding across these studies is that one *can* conduct the RA phase of software development projects offshore with as much success as co-located mode. The right mechanisms and processes must be in place, which can enhance the cost effectiveness of offshored projects without any significant impact on the quality of project outcomes. This finding is extremely significant because it opens up the possibility of extending the cost arbitrage phase of new client-provider engagements. With decades of best practices already in place, a rapid shift to offshoring the RA phase is quite feasible and may even be desirable considering cost-cutting pressures on most business units.

5.2 Lesson 7

Lesson 7: Restricting user change requests in early project phases are not particularly beneficial to project success.

When identifying factors that could impact the success of the RA phase in offshore mode, one factor we examined was the impact of permitting changes during the requirements phase. It has become clear over the years that permitting users to make changes during the requirements phase leads to a greater match between user requirements and final systems. Our study indicates that this situation is true even when offshoring the RA phase. In other words, permitting requirements changes during the requirements phase enhances the likelihood that a project will succeed even if carried out offshore, which suggests that firms should perform offshore projects in iterative mode in which they progressively elaborate requirements—similar to co-located projects.

5.3 Lesson 8

Lesson 8: Expend effort early in developing strong practices around project monitoring and control.

Research has extensively documented the importance of clients' continuously monitoring projects, especially in outsourced projects. Further, many software development firms do so. Our findings have

pointed to similar results for the RA phase. In keeping with control theory, offshore projects' success increases with the user's closely monitoring and controlling the project in the requirements phase. Firms should consider developing strong practices for monitoring and controlling projects early in their lifecycle to enhance offshore RA's success.

5.4 Lesson 9

Lesson 9: Experienced client-side coordinators reduce the need for bridge head teams.

Finally, we considered the effect of having a client-side coordinator on a project. Our study indicates that the presence of a client-side coordinator during the requirements phase leads to a more successful project. In practice, providers usually position their "bridge head" teams at client locations for bridging communication and cultural gaps between user and developer teams. Our analysis indicates that facilitators from a project sponsor at the client site can effectively monitor and control projects carried out in virtual team mode and, thereby, drive projects towards successful outcomes. Table 5 summarizes all the lessons and links them to each research question.

Table 5. Lessons Learned and Recommendations for Offshore RA success

Lesson	Findings and recommendations
RQ1: Can one offshore the RA phase?	
1	One can successfully execute RA offshore.
1a	Requirements analysis is an embedded process in the software engineering cycle. Firms should find a common framework to align these processes across client and provider teams.
6	Cost arbitrage is not yet a thing of the past.
RQ2: How do changes in requirements impact success of the RA phase when conducted in an offshore mode?	
3	There should be a shared understanding between onshore and offshore teams about how requirements changes should be viewed.
7	Restricting user change requests in early project phases are not particularly beneficial to project success.
RQ3: What factors impact the success of the RA phase in offshore projects?	
2	Offshore client and provider teams must deliberate on mechanisms for improving requirements richness.
2a	The choice of tool and techniques to support requirements gathering and elicitation offshore requires careful consideration.
2b	Firms must design processes and technologies to support both task and relationship-development aspects of RA.
4	Offshore teams should balance control and flexibility to enhance RA success.
4a	Agile approaches to software development are often effective in providing the flexibility necessary for successfully gathering requirements.
5	Client-site coordinators that facilitate RA processes more significantly impact project success than provider-site coordinators.
7	Restricting user change requests in early project phases are not particularly beneficial to project success.
8	Expend effort early in developing strong practices around project monitoring and control.
9	Experienced client-side coordinators reduce the need for bridge head teams.

6 Recommendations for Future Research

Our integrated findings, along with insights from current literature, enable one to develop best practices for successfully offshoring the RA phase (Table 5). However, to further our knowledge on offshore software projects, we need additional studies that move beyond the presently conceptualized variables. We propose a conceptual model (see Figure 5) that positions findings from our multi-method study in the context of control and flexibility for successfully offshoring RA. It highlights three key elements of client-provider roles, facilitating technologies, and the need for a good balance between control and flexibility. We integrate all findings from our study along these dimensions in Figure 5. Key future research needs include examining informal and self-control, trust, cohesion, and role of synchronous technologies such as social media, messaging systems, and shared spaces on effectiveness of offshore RA. Researchers can

use this conceptual model to further enhance and develop theories in the domain of RA offshoring. In the following paragraphs, we elaborate on specific areas for future research.

We focused on examining only formal modes of control. However, as control theory supports, one must consider informal modes of control, such as self and clan control, in offshore projects. We need further research to understand the conditions under which teams apply such informal modes of control in offshore projects and whether they impact success. Informal controls, for instance, may be difficult to implement between client and provider teams in the RA phase because teams are only beginning to get to know each other. As such, informal controls may be more visible in provider teams and may be even strong in the RA phase as provider teams try to gain rapid understanding around client needs while working towards developing client relationships.

In Section 3.4.2, we suggest that requirements changes in study 2 resulted in offshore teams' better perceiving RA success. These findings arose not completely from the industry surveys, the findings of which suggest that not provider team members expects all requirements changes to have a positive impact on project success. This topic calls for more detailed investigation because it likely impacts the other software development phases. Individual and team factors such as motivation, cohesion, and trust between offshore software team members and emotional intelligence of individual team members can influence the outcome of the RA phase. In some cases, the presence of these traits may help teams more quickly clarify the requirements. We did not consider these traits in our studies, but research has shown them to influence overall project success. Considering the challenges and rigor associated with the RA phases, one might expect that such factors will influence the RA phases more strongly than later phases, but we do yet know, and, as such, future research could address this area.

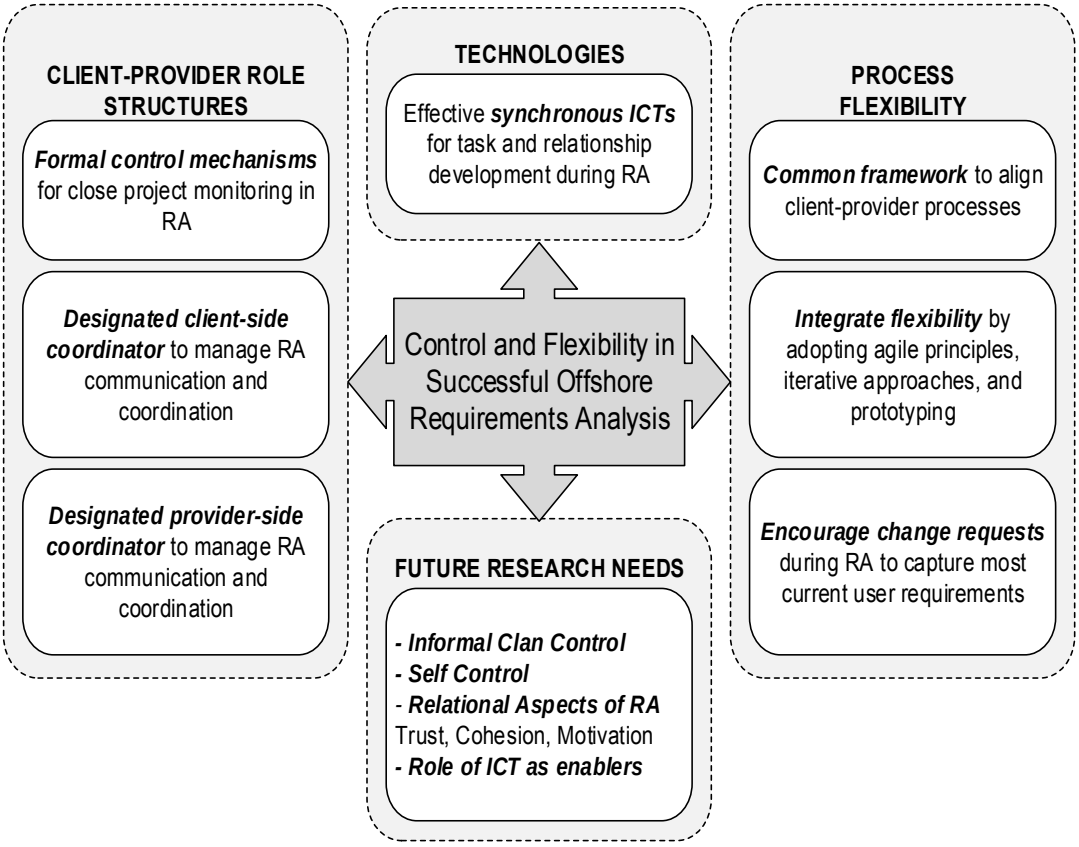

Figure 5. Conceptual Framework for Successful RA Offshoring

7 Conclusion

Global software development using virtual teams is an accepted practice in today's software industry because, for one, of the cost arbitrage between developed countries and developing countries such as India. However, firms have typically adopted this approach only for the coding and testing phases of the software development lifecycle. Quite naturally, extending this virtual team collaboration to earlier phases, such as RA, could significantly enhance the cost arbitrage argument. So far, however, research has largely not explored whether or not one can perform the RA phase offshore. In this paper, we present findings from a series of experimental and survey efforts directed at demystifying this issue.

Our multi-method study strongly points to the fact that one can conduct requirements analysis in software projects offshore using advanced synchronous communication technologies. Further, permitting changes during this phase would enhance the success of RA phase, as would the presence of a client-side coordinator and the user's closely monitoring the project. Firms need to blend structure using control mechanisms and flexibility in requirements processes to ensure that the offshore RA phase succeeds. These findings are significant and could lead to extending the cost arbitrage of offshored software projects.

Readers should view our results in the light of certain limitations. We designed our experiments using well-defined and comparably small projects from real life. In these projects, the complexity and scope of requirements analysis process was not high. However, real-world projects in industry can be larger in scope and size. As a result, they can pose complex challenges for offshoring RA. Further, our industry survey was from a provider perspective, and one still needs to examine our model in the industry from a client perspective.

References

Akbar, R., & Hassan, M. F. (2010). Limitations and measures in outsourcing projects to geographically distributed offshore teams. In Proceedings of *the International Symposium in Information Technology* (Vol. 3, pp. 1581-1585). *IEEE*.

Ali, N., & Lai, R. (2016). A method of requirements change management for global software development. *Information and Software Technology*, *70*, 49-67.

Appan, R., & Browne, G. J. (2012). The impact of analyst-induced misinformation on the requirements elicitation process. *MIS Quarterly*, *36*(1), 85-106.

Aundhe, M. D., & Mathew, S. K. (2009). Risks in offshore IT outsourcing: A service provider perspective. *European Management Journal*, *27*(6), 418-428.

Baroudi, J. J., & Orlikowski, W. J. (1988). A short-form measure of user information satisfaction: A psychometric evaluation and notes on use. *Journal of Management Information Systems*, *4*(4), 44-59.

Bass, J. M. (2016). Artefacts and agile method tailoring in large-scale offshore software development programmes. *Information and Software Technology*, 75, 1-16.

Batra, D., Xia, W., VanderMeer, D., & Dutta, K. (2010). Balancing agile and structured development approaches to successfully manage large distributed software projects: A case study from the cruise line industry. *Communications of the Association for Information Systems*, *27*, 379-394.

Bhat, J. M., Gupta, M., & Murthy, S. N. (2006). Overcoming requirements engineering challenges: Lessons from offshore outsourcing. *IEEE Software*, *23*(5), 38-43.

Boehm, B., Grünbacher, P., & Briggs, R. O. (2001). Developing groupware for requirements negotiation: Lessons learned. *IEEE Software, 18*(3), 46-55.

Bose, I. (2008). Lessons learned from distributed agile software projects: A case-based analysis. *Communications of the Association for Information Systems*, *23*, 619-632.

Brewer, J., & Hunter, A. (2006), *Foundations of multimethod research: synthesizing styles*. Thousand Oaks, CA: Sage.

Browne, G. J., & Ramesh, V. (2002). Improving information requirements determination: A cognitive perspective. *Information and Management*, *39*(8), 625-645.

Campbell, D. T., & Stanley, J. C. (1963). *Experimental and quasi-experimental designs for research*. Chicago: Rand McNally.

Carmel, E., & Tjia, P. (2005) *Offshoring information technology: Sourcing and outsourcing to a global workforce,* Cambridge, UK: Cambridge University Press.

Choudhury, V., & Sabherwal, R. (2003). Portfolios of control in outsourced software development projects. *Information Systems Research*, *14*(3), 291-314.

Cockburn, A., & Highsmith, J. (2001). Agile software development: The business of innovation. *IEEE Computer*, *34*(9), 120-122.

Creswell, J. W. (2003) *Research design: Qualitative, quantitative, and mixed method approaches*. Thousand Oaks, CA: Sage.

Crowston, K., & Kammerer, E. E. (1998). Coordination and collective mind in software requirements development. *IBM Systems Journal*, *37*(2), 227-241.

Damian, D. (2007). Stakeholders in global requirements engineering: Lessons learned from practice. *IEEE Software, 24*(2), 21-27.

Damian, D. E. H., Eberlein, A., Shaw, M. L., & Gaines, B. R. (2000). Using different communication media in requirements negotiation. *IEEE Software*, (3), 28-36.

Delisle, C. L. G. (2001) *Success and communication in virtual project teams* (UMI published doctoral dissertation). University of Calgary, Alberta.

Delone, W., & McLean, E. R. (1992). Information systems success: The quest for the dependent variable. *Information Systems Research*, *3*(1), 60-95.

Delone, W., & McLean, E. R. (2003). The DeLone and McLean model of information systems success: A ten-year update. *Journal of Management Information Systems*, *19*(4), 9-30.

Dibbern, J., Goles, T., Hirschheim, R., & Jayatilaka, B. (2004). Information systems outsourcing: A survey and analysis of the literature. *The DATABASE for Advances in Information Systems*, *35*(4), 6-102.

Edwards, H. K., & Sridhar, V. (2005). Analysis of software requirements engineering exercises in a global virtual team setup. *Journal of Global Information Management*, *13*(2), 21-41.

Eisenhardt, K. M. (1985). Control: Organizational and economic approaches. *Management Science*, *31*(2), 134-149.

Estler, H. C., Nordio, M., Furia, C. A., Meyer, B., & Schneider, J. (2014). Agile vs. structured distributed software development: A case study. *Empirical Software Engineering*, *19*(5), 1197-1224.

Evaristo, R., Watson-Manheim, M. B., & Audy, J. (2005). E-collaboration in distributed requirements determination. *International Journal of e-Collaboration*, *1*(2), 40-55.

Gopal, A., & Gosain, S. (2010). Research note: The role of organizational controls and boundary spanning in software development outsourcing: Implications for project performance. *Information Systems Research*, *21*(4), 960-982.

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.

Holmstrom, H., Fitzgerald, B., Agerfalk, P. J., & Conchuir, E. O. (2006). Agile practices reduce distance in global software development. *Information Systems Management*, *23*(3), 7-18.

Jarvenpaa, S. (1988). The importance of laboratory experimentation in IS research. *Communications of the ACM*, *31*(12), 1502-1505.

Jaworski, B. J. (1988). Toward a theory of marketing control: Environmental context, control types, and consequences. *Journal of Marketing*, *52*(3), 23-39.

Kirsch, L. J. (1997). Portfolios of control modes and IS project management. *Information Systems Research*, *8*(3), 215-239.

Kirsch, L. J. (2004). Deploying common systems globally: The dynamics of control. *Information Systems Research*, *15*(4), 374-395.

Kirsch, L. J., & Cummings, L. L. (1996). Contextual influences on self-control of IS professionals engaged in systems development. *Accounting, Management Information Technology*, *6*(3), 191-219.

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.

Korkala, M., & Maurer, F. (2014). Waste identification as the means for improving communication in globally distributed agile software development. *Journal of Systems and Software*, *95*, 122-140.

Lacity, M., & L. Willcocks, L. (2001) *Global information technology out-sourcing: In search of business advantage*. Chichester, UK: John Wiley & Sons.

Lee, G., Delone, W., & Espinosa, J. A. (2006). Ambidextrous coping strategies in globally distributed software development projects. *Communications of the ACM*, *49*(10), 35-40.

Levina, N., & Vaast, E. (2005). The emergence of boundary spanning competence in practice: Implications for implementation and use of information systems. *MIS Quarterly*, *29*(2), 335-363.

Mahaney, R. C., & Lederer, A. L. (2006). The effect of intrinsic and extrinsic rewards for developers on information systems project success. *Project Management Journal*, *37*(4), 42-54.

Mattarelli, E., & Gupta, A. (2009). Offshore-onsite subgroup dynamics in globally distributed teams. *Information Technology and People*, *22*(3), 242-269.

Nath, D., Sridhar, V., Adya, M. and Malik, A. (2008). Project quality of offshore virtual teams engaged in software requirements analysis: An exploratory comparative study. *Journal of Global Information Management*, *16*(4), 24-45.

Nguyen, L., & Swatman, P. A. (2003). Managing the requirements engineering process. *Requirements Engineering*, *8*(1), 55-68.

Ocker, R., Hiltz, S. R., Turoff, M., & Fjermestad, J. (1995). The effects of distributed group support and process structuring on software requirements development teams: results on creativity and quality. *Journal of Management Information Systems*, *12*(3), 127-153.

Ouchi, W. G. (1979). A conceptual framework for the design of organizational control mechanisms. *Management Science, 25*(9), 833-848.

Piccoli, G., Powell, A., & Ives, B. (2004). Virtual Teams: Team control structure, work processes, and team effectiveness. *Information Technology and People*, *17*(4), 359-379.

Project Management Institute. (2013). *A guide to project management body of knowledge (PMBOK).* Newtown Square, PA.

Ramasubbu, N., Mithas, S., Krishnan, M. S., & Kemerer, C. F. (2008). Work dispersion, process-based learning, and offshore software development performance. *MIS Quarterly*, *32*(2), 437-458.

Ramesh, B., Cao, L., Mohan, K., & Xu, P. (2006). Can distributed software development be agile? *Communications of the ACM*, *49*(10), 41-46.

Ramesh, B., Mohan, K., & Cao, L. (2012). Ambidexterity in agile distributed development: An empirical investigation. *Information Systems Research*, *23*(2), 323-339.

Remus, U., & Wiener, M. (2012). The amount of control in offshore software development projects. *Journal of Global Information Management*, *20*(4), 1-26.

Robinson, W. N. (1990). *Negotiation behavior during requirements specification*. Paper presented at the 12th International Conference of Software Engineering, Los Alamitos, California.

Sinha, V., Sengupta, B., & Chandra, S. (2006). Enabling collaboration in distributed requirements management. *IEEE Software*, *23*(5), 52-61.

Sommerville, I., & Sawyer, P. (1998) *Requirements engineering: A good practice guide*. Chichester, England: John Wiley and Sons.

Vlaar, P. W., van Fenema, P. C., & Tiwari, V. (2008). Cocreating understanding and value in distributed work: How members of onsite and offshore vendor teams give, make, demand, and break sense. *MIS Quarterly*, *32*(2), 227-255.

Wiener, M., Remus, U., Heumann, J., & Mähring, M. (2014). The effective promotion of informal control in information systems offshoring projects. *European Journal of Information Systems*, *24*(6), 569-587.

Yadav, V. (2008) *Flexible global software development (GSD): Factors affecting success in the requirements analysis phase* (PhD dissertation). Management Development Institute, Gurgaon, India.

Yadav, V. (2011). Research review: Globally distributed requirements engineering and agility. *International Journal of Innovation in the Digital Economy*, *2*(1), 1-11.

Yadav, V., Adya, M., Nath, D., & Sridhar, V. (2007). Investigating an “agile-rigid” approach in globally distributed requirements analysis. In *Proceedings of the Pacific-Asia Conference on Information Systems.*

Yadav, V., Adya, M., Sridhar, V., & Nath, D. (2009). Flexible global software development: Antecedents of success in the requirements analysis. *Journal of Global Information Management*, *17*(1), 1-31.

Yadav, V., Adya, M., Sridhar, V., & Nath, D. (2013). Control, process facilitation, and requirements change in offshore requirements analysis: The provider perspective. *Journal of Information Technology, Theory and Application*, *14*(3), 30-47.

Yadav, V., & Gupta, R. (2008). A paradigmatic and methodological review of research in outsourcing. *Information Resource Management Journal*, *21*(1), 27-43.

Appendix A: Questionnaire Items Used In Industry Survey (Yadav et al., 2013)

Requirements analysis success

Item	Client satisfaction with offshore RA
s1	The client was highly involved with our team during the requirements gathering process.
s2	The client clearly understood the requirements deliverables submitted by our team.
s3	The client was highly committed to the goals and tasks of requirements phase.
s4	The requirements deliverables were readily accepted by the client.
	RA artifact quality
s5	Our requirements phase deliverables adequately covered client requirements.
s6	Our team has been able to accurately capture and document requirements.
	RA process quality
s7	The requirements were captured within the original time schedule.
s8	The client was satisfied with the process by which the requirements were captured.

Requirements change

Item	
req1	We did not have any changes in requirements during the requirements phase.
req2	The level of requirements change was high during the requirements phase
req3	There were frequent changes in requirements during the requirements phase.

Process facilitation

Did your client have a dedicated liaison at the client-site; for example, a client representative who acted as a point-of-contact for your team? If yes, please answer the questions below otherwise proceed to question number Z.

Item	Process facilitation by client site-coordinator
fcl1	During the requirements phase the client liaison helped coordinate the workflow between client and our team members.
fcl2	During requirements gathering the client liaison constructively responded to our team's needs for assistance.

Did your team have a dedicated liaison here in India for the client; for example, a project manager or a team lead or a team representative who acted as a point-of-contact? If yes, please answer the questions below otherwise proceed to question number X.

Item	Process facilitation by vendor site-coordinator
fv1	During the requirements phase our liaison helped coordinate the workflow between client and our team members.
fv2	During requirements gathering our liaison constructively responded to our team's needs for assistance.

Control

Item	Outcome control
out_cnt1	The client insisted on complete and on-time submission of project status reports during the requirements phase.
out_cnt2	The client insisted on complete and on-time submission of requirements deliverables.
out_cnt3	The client insisted on timely completion of requirements phase.

Item	Behavior control
beh_cnt1	The client regularly monitored the progress of requirements phase.
beh_cnt2	The process for communication between client and our team members was well defined.
beh_cnt3	A project management plan (specifying schedules, deliverables, milestones, roles etc.) was developed for capturing and documenting requirements.
Response scale: “Please answer each of the following questions related to globally distributed requirements analysis by encircling the appropriate response”. Seven-point scale (1 = “strongly disagree”, 4 = “neutral” and 7 = “strongly agree”).	

About the Authors

Vanita Yadav is an Assistant Professor of Strategy, Innovation and Entrepreneurship. She is a *Fulbright Fellow* and a Postdoc from MIT, USA. She was also a Research Affiliate at the South Asia Institute, Harvard University for 2012-13. She has a Doctorate in Management from MDI Gurgaon, India. Vanita has published and presented research in refereed international journals including *JITTA*, *JGIM* and *IRMJ*, conferences including *AoM, PACIS*, and *SMS*, and book chapters. She has published two case studies at the Harvard Business School publications online. She is the Associate Editor of the *International Journal of Rural Management* and the *International Journal of Innovation in the Digital Economy*. She has worked on research projects funded by the Indian Government, ICSSR, European Union and institutional seed funded projects. Before joining academia, she worked in the Indian IT industry with NIIT Ltd.

Monica Adya is an Associate Professor of Management at Marquette University. She received her PhD in Management Information Systems from The Weatherhead School of Management, Case Western Reserve University. Monica researches in two areas: a) IT workforce spanning themes related to workplace diversity, offshore practices, and virtual team management, and b) forecasting support systems. For some of this work, she has received grants from 3M Foundation and Naval Surface Warfare Center. Monica has published in several journals including *Decision Sciences Journal of Innovative Education (DSJIE)*, *Human Resource Management*, *International Journal of Forecasting (IJF)*, and *Information Systems Research (ISR)*. She serves on the editorial boards of *IJF*, *DSJIE*, and *Communications of the AIS*.

Dhruv Nath is Professor and Chairman of the Information Management Area at the Management Development Institute (MDI), Gurgaon, India. He was also Chairman of the Doctoral Program at MDI. Prior to MDI, he was Senior Vice President at NIIT Ltd, a leading software and education company. Prof. Nath holds a B. Tech in Electrical Engineering and a PhD in Computer Science, both from Indian Institute of Technology (IIT), Delhi. He has conducted Top Management Workshops in IT and has been a consultant in the area to several organizations such as Glaxo, Unilever, Gillette, Nestle, etc. He has published research papers in international journals such as the *IEEE Transactions on Computers* and has written two books on Managing IT Technologies.

Varadharajan Sridhar is a Professor at IIIT Bangalore India. Dr. Sridhar has taught at many Institutions in the USA, New Zealand, Finland and India. He was also a Research Fellow at Sasken Communication Technologies, Bangalore, India. At Sasken, he looks after research and development, idea incubation, knowledge management, and strategic initiatives. He has published in *Telecommunications Policy*, and *Harvard Business School publishing*. His books titled *The Telecom Revolution in India: Technology, Regulation and Policy and The Dynamics of Spectrum Management: Legacy, Technology, and Economics* were published by the Oxford University Press India in 2012 and 2014. He has authored more than 175 articles in Indian business newspapers and magazines. He is the Co-Editor of *International Journal of Business Data Communications* and *Networking*. He has a PhD from the University of Iowa (USA). His work can be accessed at: http://www.vsridhar.info.