

# A QUALITATIVE INVESTIGATION OF THE SECI MODEL'S KNOWLEDGE CONVERSIONS IN THE APPLICATIONS DEVELOPMENT CONTEXT

Christobal Cayaba, De La Salle University, Manila, Philippines, [chriscayaba@yahoo.com](mailto:chriscayaba@yahoo.com)

Zelinna Pablo, De La Salle University, Manila, Philippines, [zcdpablo@gmail.com](mailto:zcdpablo@gmail.com)

Preferred Journal: PAJAIS

## Abstract

*The Theory of Knowledge Creation generally suggests that tacit and explicit knowledge are converted through the four modes known as the SECI Model (Nonaka 1991; Nonaka & Takeuchi 1995). In applications development, the knowledge conversions are mobilized through the use of tools (video conference, development editor) and practices (code review, design patterns, pair programming) (Henninger 1997; Avram 2007). However, the model is criticized for having strong Japanese cultural influence and little empirical basis in practice resulting in several debates on its applicability, existence/non-existence of the SECI cycle and unidirectional/multidirectional property of the conversions (Gourlay 2003; Rice & Rice 2005; Hong 2010).*

*Therefore, we studied how tacit and explicit knowledge are converted (tacit-tacit, tacit-explicit, explicit-explicit, and explicit-tacit) in an empirical setting and explored what the implications are within the context of applications development using the Theory of Knowledge Creation's SECI Model. We did this by immersion in a non-Japanese organization where applications development tools and practices were employed. Interviews, document reviews, and observations were used as primary data gathering techniques, which consequently required qualitative study analysis, specifically phenomenological, and discourse analysis techniques.*

*Keywords: Theory of Knowledge Creation, SECI Model, Knowledge Conversions, Applications Development.*

# 1 INTRODUCTION

In this paper, we studied how tacit (intangible) and explicit (tangible) knowledge are converted in an empirical setting and explored the implications of the conversions within the context of applications development using the theory's SECI Model, which is an acronym for Socialization (tacit-tacit), Externalization (tacit-explicit), Combination (explicit-explicit) and Internalization (explicit-tacit) (Nonaka 1991; Nonaka & Takeuchi 1995). The Theory of Knowledge Creation plays a significant role in knowledge management paradigms (Aggestam 2006; King 2009; Alipour et al. 2011; Khalili et al. 2011). For one, the SECI Model has implications both for managerial style and organizational structure, and which was able to emphasize human communication as an essential component (Rice & Rice 2005). Understanding the extent of its applicability determines how organizations may then further strategize their knowledge management models to custom-fit their goals.

The Theory of Knowledge Creation, as applied in applications development, is important because literatures suggest that applications development tools and practices primarily rely on tacit and explicit knowledge interactions and conversions (Bailin 1997; Endres et al. 2007). Understanding how these interactions and conversions work within the context of applications development helps key entities, such as applications developers and program codes, to act or be organized in ways that may eventually contribute to a higher success rate of applications development projects.

Further studies on knowledge creation and how it might apply to applications development are therefore significant. As mentioned earlier, the Theory of Knowledge Creation suggests that tacit and explicit knowledge are converted through the four modes defined in the SECI Model. During the applications development process, there is a series of tacit and explicit knowledge interactions and conversions involved. In addition, the ability to develop applications is considered a skill, a form of tacit knowledge, while the outputs (web, desktop and mobile applications, technical documents, etc.) are in explicated forms. Applications development tools (video conference, development editor, etc.) are used for tacit and explicit knowledge interactions. These tools allow knowledge to be stored for reference and mobilize interactions through chat, email, video-conferencing, and prototyping user interfaces. Moreover, applications development practices (code review, design patterns, pair programming, etc.) are rich with tacit and explicit knowledge interactions. There is the so-called "best practices", which aims to achieve standard coding process and to address commonly recurring technical issues such as the use of coding standards (capitalization, indentation, variable naming, etc.) and applicable design patterns (Singleton, Model-View-Controller, etc.). More practices also include defined methodologies that identify the stages in applications development projects; primarily, these are requirements gathering and analysis, design, coding, and testing, which heavily rely on knowledge interactions and conversions (Bailin 1997).

However, much of the studies delved in the general application of the theory's conceptual model. Although it has been noted intuitively attractive, there are little empirical studies conducted to examine the SECI Model as a process model in practice (Rice & Rice 2005) such as in applications development. On the contrary, one of the key subjects explored in the development of the SECI Model was an applications developer. There are then debates that arose. One of which is the Universalist-Pluralist debate that criticizes the applicability of the model where the level of cultural influence, based on Japanese car manufacturing companies, is the core of the debate (Hong 2010; Andreeva & Ikhilchik 2012; Cayaba 2012). Other criticisms or debates on the SECI Model point out the existence or non-existence of a cycle and the unidirectional or multidirectional property of the conversions. Since there is no strong empirical basis, the model is conceived by some as seriously flawed. Consequently, there are studies that say the conversion modes are not coherent, which then argue that the paradigmatic status of the theory is unwarranted (Gourlay 2003; Rice & Rice 2005).

There is also an ongoing debate involving technology, such as applications development tools, as being either value-laden or value-neutral (Brey 2009; Flanagan et al. 2008). This consequently has

implications to the process of learning, interactions, and conversions of knowledge. Adding to this, given that there are significant improvements in applications development such as the so-called “best practices” and defined methodologies, there is still a significant failure rate of these projects (Morisio et al. 2007) even those large and planned ones (Humphrey 2005) due to several reasons such as mishandled requirements during the requirements elicitation and gathering stage (Kaur & Sengupta 2011). This stage primarily consists of tacit-to-tacit interactions, which remains less explored compared to interactions involving explicit forms (Mohamed 2008; Mohamed 2010).

To explore the role of knowledge creation in applications development and vice versa, we framed the question: What are the implications of applications development tools and practices to the knowledge conversions, and consequently, to the success or failure of applications development projects?

To answer this, we immersed into the offshored applications development team in an organization and examined applications development tools and practices because they provide rich tacit and explicit interactions, which are necessary for the knowledge conversions in the SECI Model to mobilize. In effect, we used a qualitative case study. This is more appropriate given our research question because it called for studying the interactions and discursive activities involved among human beings. Specifically, while the members of the applications development team perform their duties, we immersed into the environment and our notion towards the subject becomes an expression or reflection of how we make sense of the situation. We gathered data how tacit and explicit knowledge are converted through the employed tools and practices by conducting interviews, document reviews, and observations involving key entities such as the developers, online status posts, emails, and chats. We then analyzed the data using phenomenological, and discourse analysis techniques. The use of a variety of data gathering methods allowed us to triangulate data, which then helped address ethical and reflexivity issues concerning observable assumptions, biases, and vague areas.

This study has made contributions to theory, method, and practice. First, in answering the research question, we hope to have enriched the discussions on the debates revolving around the Theory of Knowledge Creation. Since there is also less empirical work that studied the SECI Model as a process model in modern domains such as in the field of Information and Communications Technology (ICT), this study becomes a useful reference in examining the extent of many factors including the aspect of the theories’ stability and applicability. For the second contribution, this study may also provide an added value to methodological understanding of the SECI Model in sustaining or straightening the claims of the SECI Model as a process model by performing qualitative investigation that critics and the proponents would suggest. Finally, we also contribute to practice by presenting the value of theory-grounded know-how as opposed to mere gut-feel in decision-making. By presenting the implications of the applications development tools and practices to knowledge conversions, ICT professionals become more aware of their vulnerabilities that are caused by internal (self) and external (environment) factors, which then become possible areas for improvements.

## **2 LITERATURE REVIEW**

Since our study involved knowledge, we start by giving an overview on knowledge management, which already gained an impact in modern research (Aggestam 2006). We then build up the discussion towards knowledge conversions and applications development, which are the core of our study.

King (2009) defined knowledge management as the “planning, organizing, motivating, and controlling of people, processes and systems in the organization” in order for knowledge assets to be improved and used effectively. For Aggestam (2006), knowledge management is the process that aims to organize knowledge assets, which then yields organizational learning (Alipour et al. 2011). These knowledge assets may be in the form of documents or in electronic repositories such as databases that are managed through many computer-based communications and information systems called knowledge management systems (King 2009). However, early knowledge management paradigms tend to simply focus on technological innovations such as the use of document management systems

and collaborative technologies. It is the birth of the Theory of Knowledge Creation that gave emphasis on human factors (Khalili et al. 2011). Primarily, the Theory of Knowledge Creation distinguishes two types of knowledge being tacit and explicit, which are enriched and mobilized through several conversion modes. These conversions are carried out in organizations working on a particular domain (Alipour et al. 2011).

However, relevance of knowledge management academic research works were questioned (Fergusson 2005) and that there is a necessity to translate the findings to practice (Booker et al. 2008). We therefore focused on two important things: the Theory of Knowledge Creation for the theory, and Applications Development for the domain. We also present several criticisms on these two where our research question was based.

## **2.1 Theory of Knowledge Creation**

The Theory of Knowledge Creation by Nonaka was highly influenced by the work of Polanyi (1966) where the distinctive aspects about tacit and explicit knowledge were presented. Briefly, tacit knowledge is subjective, experience-based and context-specific while explicit knowledge is objective, rational and context-free (Nonaka 1994). Nonaka and Takeuchi (1995) argue that these tacit and explicit knowledge create new knowledge in the process defined in the SECI Model, which is an acronym for Socialization, Externalization, Combination and Internalization. The first entails transfer of tacit knowledge in one person to tacit knowledge in another person (usually between individuals) through a series of direct interactions. The second is making tacit knowledge explicit between individuals within a group allowed by sharing beliefs, ideas or images in words, metaphors and analogies, and articulate thinking through instant feedbacks and exchange of ideas. At the third phase, the explicit knowledge acquired then is further transferred to become tangible and thus transfers explicit knowledge among groups within an organization by conveying them in documents, email and databases. Last, internalization is when the explicit knowledge is understood and absorbed into tacit knowledge by the individual (Nonaka 1994).

The study of Nonaka and Takeuchi (1995) involved examination of the sharing of knowledge that existed in Japanese organizations. In their study, the conversion of knowledge between the members of the organization is coursed by continuous involvement in learning-by-doing and shared experiences, which originate from the individual's tacit knowledge. With an emphasis in the tacit-to-tacit knowledge in the Socialization Phase, Hong (2010) presents that the tacit knowledge possessed by the individual will not be of value when it is not made explicit and shared with the other members of the organization through language, images, and other concrete means, which constitute the process of Externalization. Metaphors and other use of analogies are therefore used to deliver difficult ideas and images in mind (Nonaka 1991; Pablo 2007). The use of these metaphors and other analogies among Japanese managers allow them to characterize abstract ideas, which may be difficult to discuss, in order for others to develop concrete forms out of their implicit understanding and personal insights for shared interpretation (Nonaka 1994). Moving on to the next mode is Combination, where explicit forms of knowledge are combined and shared with others forming the integration of unique perspectives (Hong 2010). These shared views then create a shared environment (Nonaka & Kono 1998) for the members of the organization to transform the explicit knowledge into tacit knowledge through action and reflection during the internalization process. The entire process starts again after the enriched tacit knowledge of individuals are shared with others.

Hong (2010) presents two enablers for the knowledge conversions to happen. These are the requirement for a more proactive role in the knowledge creation process, and the maintenance of the shared context. The first promotes having the "knowledge activists" role (Hong 2010) and for the management to uphold knowledge creation throughout the organization by cascading hard-to-understand and experience-grounded concepts down to the members of the organization. This leads to the second enabler, which takes its concept from Nonaka and Kono (1998) in saying that the shared context becomes a room or space for innovations to materialize. Nonaka and Toyama (2003) defined

“ba” as “an existential place where participants share context and create new meanings through interactions” either physically, socially or virtually (in nature). It is the continuous participation in “ba” that allows members to: first, develop a shared goal; second, overcome limited perspectives; third, promote further interaction; and fourth, therefore, create new knowledge (Nonaka et al. 2000). Since Japanese organizations are organized in teams, less attention is given to individual contributions. It is in these groups where formal and informal participations are imposed that help establish and strengthen relationships with co-workers. It is through the notion of being engaged in group dialogues that members are able to bring out and share their tacit knowledge and be explicated (Hong 2010).

### *2.1.1 Applicability of the SECI Model: Universalism versus Pluralism*

It is now clear that human interactions are critical for the carrying out of the processes involved in the knowledge creation theory. However, this depends largely on the willingness and the capacity of the individuals to be able to participate in activities that involve interactions and collaborations. The debate now arises whether the so-called continuous collaborations that exist in Japanese organizations (Nonaka & Takeuchi 1995) would still be applicable for entities outside the premises of the Japanese cultural conditions. Some, therefore, question whether the SECI Model may or may not be adapted in other cultural contexts. The theory is argued to have potential cultural inclinations, thus affecting its applicability only to Japanese organizations, which brings out the Universalistic-Particularistic debate, where the main argument against the claim for universality centers on the trace of Japanese influence.

First, the socialization process happens when there is a series of interactions that allow one’s tacit knowledge to be shared with another. The Japanese have long been associated to be predominantly collectivists (Hofstede 1980; Singh et al. 2003). With this mentality, they are ruled by a behavioral disposition call “*giri*”, which describes the unconditional obligation to protect and look after social relationships and prefer strong personal tie among them (Hong 2010). It is through direct experience, face-to-face interactions and observations that they learn and make sense (Nonaka & Kono 1998). Second, the process of externalization, in which tacit knowledge is transformed into explicit knowledge, entails the use of metaphors in order for their subjective, tacit knowledge is comprehended to explicit forms (Nonaka & Toyama 2003). Since the Japanese culture is driven by high uncertainty avoidance, they have developed the inclination to collect and share information necessary to avoid the impact of uncertainties (Hofstede 1980). Third, the process of combination entails group learning that requires the need for the involved groups to develop a strong motivation to speak up what they know and share them to others. The Japanese tradition of having uncensored access even to important information regarding their work practices, technology and other relevant issues that arise in the workplace allow the Japanese workers to create a shared knowledge of resources (Hong 2010). Last, the process of internalization, which is adopting explicit knowledge as one’s own tacit knowledge involves a high level of adjustment and flexibility in order for the self to permit a new understanding or concept to be embodied (Hong 2010). The Japanese is known for not making bold moves and effecting drastic changes, associating them to having high level of uncertainty avoidance (Hofstede 1980). As noted by Nonaka et al. (2000), Japanese managers participate in sessions that allow them to share their experiences with the other members and partake in activities that allow them to make new connections through experience and over everyday practice, thus enriching their tacit knowledge.

### *2.1.2 Properties of the SECI Model: Existence versus Non-existence of the SECI Cycle and the Unidirectional versus Multidirectional Property of Knowledge Conversions*

Although the Theory of Knowledge Creation has been cited by several authors in knowledge management (Choi & Lee 2002; Khalili et al. 2011), there are still criticisms to the properties of the SECI Model and question its “paradigmatic status”. Gourlay (2003) mentioned that the empirical basis of the SECI Model is highly unsatisfactory and the knowledge conversion modes are not coherent and therefore, the model may be flawed. Furthermore, the knowledge conversion modes had been studied by other fields that the proponents overlooked (Adler 1996). Jorna (1998) also believed that it is a

failure to have omitted many important learning theory philosophers and even noted that there are some misreadings of important organizational writers. In addition, as knowledge conversion modes depict changes in the knowledge, a framework that dealt with “signs” to these changes is required, but is absent. There are also studies that say each of the knowledge conversion modes is dependent on the type of tasks (Becerra-Fernandez & Sabherwal 2001), which consequently suggests that depending on the characteristics of the task at hand, workers learn with different levels of experiences in these modes and learning may not necessarily need to go through the SECI cycle. Nonaka also cautioned that the survey they used made generalization to other cultures difficult, however, it is criticized that they still arrived at the conclusion about the four modes of knowledge conversion going through a cycle that starts from socialization. The use of survey questionnaires to study the complexity of tacit and explicit knowledge conversions is doubtful as it was designed to study content but is not enough to validate the SECI Model as a process model; evidence for the strictness in the process of the SECI cycle is weak or non-existent (Gourlay 2003).

As the surveys and case studies may be flawed, these are used as arguments against the existence of the SECI cycle and the unidirectional property of the conversions. Nonaka’s accounts on tacit and explicit conversions may generally be called into question and that there may be significant theoretical shortcomings (Tsoukas 2003). The concept of “ba”, which was used to explain the context in which the SECI modes can transcend smoothly, also had little empirical grounding (Rice & Rice 2005). Gourlay (2003) suggests that instead of knowledge creation, his theory could have been called Theory of Semantic Information; as Nonaka and colleagues made a distinction between information and knowledge in their previous works, they, however, did not equate semantic information with knowledge.

## **2.2 Applications Development**

ICT research works are often seen as to how these studies can contribute towards productivity and other positive benefits (Miller 1996). Specifically in the applications development, which is a field under ICT, several research works have been conducted that are often in the line of creation of applications as tools that offer several benefits ranging from personal to societal. For instance, Lim et al. (2007) created an application that automatically generates use case diagrams that are necessary for requirements gathering and analysis. This acts as a guide for business analysts when trying to understand the requirements and resolve ambiguities. Another, Henninger (1997) presents the use of case-based knowledge management tools for applications development. However, there are other research works that focus on ICT for its other value. Pablo (2007) found the value of studying metaphors on websites. There are also other research works that link knowledge, specifically, tacit and explicit knowledge with applications development (Bailin 1997; Avram 2007; Endres et al. 2007).

In Mohamed’s earlier study (2008), he argued that much of the knowledge management systems available are designed in a formal structure while there is less effort to capitalize into capturing the tacit information from individuals. Mohamed (2010) then studied and characterized the tacit knowledge acquisition from the requirements gathering stage of the software lifecycle model. According to him, since requirements gathering involves intensive discussions and deliberations, it is identified as having the richest tacit knowledge interaction. He asserted that the representation of tacit knowledge through requirements specification is a complex process since it may take many forms such as gestures in order for the requirements to be understood (Mohamed 2010). The richness of these discourse elements (Pablo 2007) are hardly mapped into formal media.

There are also some research works that involve applications development practices such as studies in pair programming, code reviews and applications development lifecycle models. First, pair programming is a social skill that takes time to learn. It works as a cooperative way to work with partners regardless of corporate status (Extreme Programming 2012). There is a series of collaborations and discussions involved and a lot of transformation of one’s tacit knowledge to explicit knowledge by allowing each member of the pairs to transcribe their ideas into code. In pair

programming, the two applications developers produce codes through a series of close working relationship. Code Review is similar, however focuses on identifying bugs, encourages collaboration, and keeps code more maintainable. SmartBear (2012) conducted the world's largest-ever published study on code review, which includes two thousand five hundred code review sessions for the fifty programmers and three million two hundred thousand lines of code at Cisco Systems, which allowed them to come up with eleven best practices for performing code reviews used in various applications development lifecycle models. Aside from the requirements analysis and coding that were mentioned earlier, other research works involve other primary stages such as application design (Yamamoto 2008), testing (Dai & Chen 2007; Xu et al. 2007), and design patterns (Dong & Zhao 2007).

### 2.2.1 *Value-ladenness versus Value-neutrality of Tools*

The idea that technology is value-laden or value-neutral has its roots from early interdisciplinary works involving science, technology, and society. Williams and Edge (1996), for instance, believe that technology is socially shaped. However, though the debate on value-ladenness and value-neutrality is deemed important, what is value-neutral or value-laden has not been totally defined. For one, the notion of value and valuable are ambiguous and may often be described in both concrete and abstract levels such as injustice and other valued possessions (Brey 2009). Several authors then tend to come up with several criteria by studying biases, moral norms, consequences, which are then associated to observable personal and societal values.

Friedman and Nissenbaum (1996) analyzed how three types of biases emerge in computer applications. First, they argued that pre-existing biases emerge from the values and attitudes of the designers of the applications; the second type involves values embedded into computer applications known as technical biases; and the third type is the emergent bias that comes out when the intended design resulted in other social contexts. For example, Johnson (1997) claimed that the Internet is inherently a democratic technology, and therefore promotes this by empowering individuals, and facilitating dialogues and decision-making process. She argued however that this inherent democratic property may however be suppressed by humans such as by filtering access to select individuals. Therefore, these technologies may then reflect both democratic and non-democratic potentialities (Camp 1999). On the other hand, Barlatier et al. (2006) believe that ICT tools are seen as enablers for people where they felt more encouraged to give their opinions, more free and less observed inhibiting fears of expressing oneself in public, thereby increasing participation and commitment. Nevertheless they noted that while it is easy to determine the level of interaction such as through the number of emails, for instance, it is difficult to determine the level of interest of people using these tools. They argued that tools are valueless when they are not shared. Others believe that there are no inherent consequences to computer applications and technological artifacts, but argue that there are multiple ways in which these may be used and each use has its own sets of consequences (Brey, 2009). The understanding of embedded values becomes distant because the views of the applications developers and designers who work on these systems may be distant from the views of the users. Thus, there are many applications development practices that are morally opaque because they operate in difficult to understand ways for laypersons. This consequently makes these embedded values also hidden from the view of the average users.

More studies analyze embedded values in technological artifacts and computer applications such as privacy and trust. For instance, Introna (2005) studied values existing in face-detection applications. Flanagan et al. (2005) studied values in computer games, where characters revealed racial prejudices or patriarchal values. Tavani (1999) analyzed the impacts of data mining to the issue of privacy. While some computer applications and technological artifacts impose significant constraints to the environment including the way people act and behave merely by their presence, there are those that may even behave autonomously and are capable of exhibiting behaviors on their own such as robots and artificial agents (Brey 2009). These applications and artifacts therefore have consequences, oftentimes social consequences that reflect embedded values (Latour 1992).

### 2.2.2 *Significant Failure Rate of Projects despite “Best Practices” and Defined Methodologies*

Software process models aim to represent architecture, design, and definition of software processes that are intended to make significant advantages to cost, time, quality, and handling requirements changes of clients (Humphrey & Kellner 1989). Significant efforts have been made in this field but still, applications development projects have resulted in failures in terms of cost and schedule slippage, and not meeting expectations from users (Mohamed 2010; Kaur & Sengupta 2011). Humphrey (2005) suggests that before the invention of the Capability Maturity Model (CMM) and Integration (CMMI), the main problem was the lack of plans in these projects (Chrissis et al. 2003). The methods defined in CMM and CMMI then emphasized the importance of applications development plans leading to more sound project management practices (The Standish Group International Inc. 2001; Humphrey 2005). However, the achieved success rate is still an insignificant amount. Several reasons were found to account for these failures. Tilmann et al. (2004) suggest these are rooted in the project management style employed in the applications development process while Kaur and Sengupta (2011) identified several factors, which include requirements gathering, lack of user involvement, team size, testing, and poor quality management. Requirements gathering is the first stage of applications development, which is usually conducted through a series of meetings and discussions, and exchanges of documents containing the agreed requirements. However, there are requirements that have been hardly written down into documents because most of them were not initially thought of during the requirements stage (Mohamed 2008). Great part of this knowledge, such as in the case of requirements gathering, is realized in tacit form, which is volatile and hardly captured in a formal way; managing this type is therefore considered challenging (Mohamed 2010). Kaur and Sengupta (2011) believe that larger-sized teams tend to break proper communication and tend to be less flexible over smaller-sized teams as teams with more members require more interactions, sharing of ideas, feedbacks, and knowledge. They also added that it is more difficult to call meetings as well as feedbacks, and tends to result in bad communication, and poor testing, which may eventually lead to poor quality of the applications.

The use of the Internet, for example, made emails a more convenient method for trying to capture tacit knowledge. But this resulted in the neglect of the value of face-to-face interactions where richer tacit experience would more likely to surface (Mohamed 2008). Becker (1997) explains that the hesitation to meet for face-to-face interactions for requirements gathering may be due to shyness, low status, and even laziness. Mohamed (2008) suggests that the explication of tacit knowledge among the members of the applications development team is necessary to be integrated into the knowledge development processes such as applications development practices. Unfortunately, he points that the importance of such integration has not been fully understood yet. Furthermore, both academic and practitioners seems to fail in creating efficient methods for measuring knowledge (Barlatier et al. 2006).

## **3 RESEARCH QUESTION**

As presented earlier, collaborations and interactions among humans give consent to the movement of the processes in the model. However, since the model was patterned from Japanese organizations, there is a suspected cultural bias, which then leads to several debates. Noting that in the discussion of “ba”, physical or virtual interactions are emphasized, it is also important to examine the importance of studying modern tools and practices that enforce cooperation and group effort, which consequently aid in knowledge conversions. In our study, this is specific to the field of applications development.

There is also an ongoing debate involving technology, such as applications development tools, as being either value-laden or value-neutral. This consequently has implications to the process of learning, interactions and conversions of knowledge. Adding to this, given that there are significant improvements in applications development such as the so-called “best practices” and defined methodologies, there is still a significant failure rate of these projects (Morisio et al. 2007) even those large and planned ones (Humphrey 2005) due to several reasons such as mishandled requirements during the requirements elicitation and gathering stage (Kaur & Sengupta 2011). This stage primarily

consists of tacit-to-tacit interactions, which remains less explored compared to interactions involving explicit forms (Mohamed 2008; Mohamed 2010).

To further explore the role of the knowledge conversions in applications development, we framed our research question as follows: What are the implications of applications development tools and practices to the knowledge conversions, and consequently, to the success or failure of applications development projects?

Having presented our research question, we now proceed to the discussion on our methodology.

## **4 METHODOLOGY (RESEARCH FRAMEWORK AND METHOD)**

This section presents why qualitative approach is used, the justifications for the case study, and the discussion on the data gathering and analysis techniques.

### **4.1 Qualitative Approach**

We used qualitative approach for three primary reasons. First, Nonaka recognized that generalizability to other cultures is difficult and recommended the use of more qualitative investigations (Nonaka et al. 1994; Gourlay 2003). Critics also believe that the SECI Model claimed as a process model lacked empirical basis in practice (Gourlay 2003; Rice & Rice 2005). Furthermore, our research question was also framed to develop the theory, in which qualitative studies are also designed for (Ospina 2004). Therefore, looking at the knowledge conversion process in practice by qualitative investigation can provide more empirical proof to the conceptual or theoretical claim of the SECI Model as a process model. The domain that has been chosen in this study is applications development where knowledge interactions and conversions are rich.

Second, our view of knowledge is socially-constructed (Williams & Edge 1996) and consequently, we take the interpretivist stance (Kelliher 2005). The interpretivist worldview believes that realities cannot be understood in isolation from their contexts and that the knowledge of the world is constructed through lived experiences by people (Weber 2004; Johari 2009); conducting qualitative approach in this study's natural setting is important. In immersing ourselves to the discursive activities involved among human beings, based on this paradigm, we are studying subjective aspects and our findings are affected by our interactions with the study participants, and therefore, it is also our own interpretations that can best grasp and evaluate the meanings of these interactions (Weber 2004). In addition to these, Orlikowski and Baroudi (1991) suggest that the positivist tradition, or the view that there is just one universal truth (Weber 2004), dominates ICT research and that much would be gained if other approaches are studied. Pablo (2007), for instance, used discourse analysis in studying the discursive activities such as images involved in websites, which then allowed her to suggest to further examine how metaphor as a cognitive device influences the way in which people engage with web portals.

Third, there are debates on applications development tools being value-laden or value-neutral (Brey 2009). There are also debates on having significant failure rate of projects due to less explored tacit knowledge studies (Mohamed 2008; Mohamed 2010), which is seen as being "hard to capture" (Polanyi 1966). These debates involve subjective aspects, which create a compulsory consequence to base our analyses and interpretations from experiential notions, and which may be conducted through qualitative study. The qualitative data contains more than the data spoken or written down by the participants. There is much to gain that is at a much more abstract, and may even be at a nonverbal level (Klein & Myers 1999). There is value in this when trying to understand and coming up interpretations to the depth of interactions between us and the participants, as well as the interactions among participants themselves especially those involving tacit knowledge. By doing so, we are gaining an insider's perspective of this subjective phenomenon through immersion, also known as the case study (Walsham 1995).

## 4.2 Case Study

In this study, we immersed in one of the largest dental distributors in the United States. It has now over thirty thousand customers and is home to over one thousand three hundred employees spread across the fifty states. In 2010, the company ventured into outsourcing, choosing the Philippines for its first international location with the goal of obtaining offshore support for its telemarketing and ICT needs. This organization is interesting to study for the following reasons:

First, the company is a non-Japanese organization. Specifically, it is a United States (US)-owned company. According to Hofstede (1980), the US is the exact opposite of Japan in terms of High and Low Uncertainty Avoidance, and Collectivism and Individualism. This makes the organization an important case study subject since the arguments in the debates discussed in the literature review dwelled on the cultural specificity of the model in Japanese organizations. It is also important to study an organization that did not only originate from a non-Japanese organization but should also have relevantly established a different culture from Japanese. While the parent company employs Americans, the international location employs Filipinos. There is a wider aspect of complexity in terms of cultural differences. The mixed culture adds more value in this research since the model is alleged to be only applicable where there is a more proactive interaction (Hong 2010). Although the Philippines and Japan are both Asian countries, there are still some differences in culture. Hofstede (1980) specifies the difference in terms of Weak and Strong Uncertainty Avoidance. Nixon (2005) pointed out difference in social practices such as bowing for Japan and handshakes for the Philippines as well the Philippines' use of English as one of its official languages is traced back from western influence. Second, the company reflects the knowledge conversions defined in the SECI Model through their applications development tools and practices, which mobilize collaborations and a series of tacit and explicit interactions, which is the most basic characteristic of the SECI Model. Last, the members of the applications development team of both locations come from different backgrounds. Aside from the difference in location and culture, the members have different educational degrees (computer science, mathematics and journalism). This difference requires the members to interact most often. Given the concept of "ba", the case subject, having members of the team with different backgrounds, create a "room or space" that becomes a key interest in the research as much collaboration and interactions are expected in order for the team to deliver the applications.

## 4.3 Data Gathering Techniques

We performed three primary data gathering techniques, namely, participant observation, document reviews, and interviews. Initial observation of more than two years in the organization has been made. Literatures suggest the importance of having preliminary observations before the formal research is conducted, making it more iterative (Iacono et al. 2005). In our case, during this period, we were able to get significant information about the employed tools and practices, the participants, and the processes involved. We spent about six months more to formally conduct well-documented participant observation on the daily routine of the applications development team members. The team worked four times a week usually Mondays through Thursdays with the flexible work shift usually from twelve noon until eleven in the evening. Participant observation is a type of observation where researchers are immersed together with the study subjects, which is believed to be beneficial in terms of gaining well-interpreted knowledge for MIS phenomena where meanings and practices are deemed important to be understood (Nandhakumar & Jones 2002) in case studies. The preliminary observation process allowed us to formulate more informed and specific questions, which were crucial for further data collection and interpretation. During the participant observation, we had more opportunity to profile our participants by understanding further their backgrounds, their behaviours, their styles in applications development, and the practices they are accustomed to. Involvement in several applications development projects that entailed direct and indirect interactions and collaborations were also helpful as well as those information that we have observed during the weekly applications development meetings held on Wednesdays at ten until eleven in the evening. As a summary, the

observations during the immersion allowed us to explore the four modes of knowledge conversions by involvement in major meetings and discussions, actual applications and research development involving several stages such as requirements gathering, analysis, and database design. To be more systematic during the process, we kept track of these observations by writing the narratives on a daily basis and were synthesized weekly. Aside from the narratives, our interpretations and reflections about the data as well as the possible questions that we needed to find out for the upcoming days were also written down. Having both narratives of what transpired and interpretations at the early stage allowed us to achieve an evolving analysis as part of the overall inquiry process (Maxwell 2008).

Second, as this study dealt with conversions involving explicit knowledge, several reviews of this form of knowledge were necessary. The SECI Model did not only mention actual interactions among people as driving forces for knowledge creation, but also the transcription or the explication of tacit into documents, for example. In this study, we examined the following: chat history, emails, and intranet site discussions for several reasons. Chat history and emails contained several conversations involving requirements as well as directives for the completion of the projects. The online discussions were heavily used by the applications development team to inform the entire team about the other projects' statuses and issues as well as put in their comments for further discussions. As suggested by Letts et al. (2007), document reviews are often employed for historical research studies involving data about past events. Since our study involved how knowledge converts to other forms, it is important to review how these documents have matured or not and consequently, to understand how they reveal or not reveal the knowledge conversions; these generally store information, which merited scrutiny.

Third, as we also did participant observation, it would be less natural to conduct formal interviews; instead several informal, conversational interviews were performed. While we discussed about specific tasks in applications development that we needed to analyze, code or generally work on, we were also gathering data by informal, conversational interviews with them, making the setting more natural. Questions were spontaneous, not necessarily pre-determined, and we were relatively not bound with time for interview schedules (Turner 2010). Several informal, conversational group discussions were also employed. Group discussions were useful in brainstorming and expounding on matters that may not arise during one-on-one discussions. The data gathered from these informal, conversational interviews were included in our weekly observations documents that included the narratives and interpretations. Gall et al. (2003) summarize that the use of informal, conversational interviews are effective when relying on the spontaneous formulation of questions in a natural setting. We have also employed what is known as "member checking", which involved discussion of our interpretations back to the participant (Carlson 2010). Participants may or may not agree with our early interpretations so this process allows for some give and take with the belief that the results will be a more accurate reflection of reality (Krefting 1990).

#### **4.4 Data Analysis Techniques**

We conducted a variety of analysis approaches appropriate to analyze the data, specifically we used phenomenological, and discourse analysis techniques. First, phenomenological perspective seeks to understand the phenomenon by studying people's daily life experiences (Groenewald 2004). This "lived experience" requires us to enter into an individual's life world focusing on the participants' perspectives on how the event was experienced and use the self to interpret the individual's or group's experience (Letts et al. 2007). Specifically in this study, our research question aims to understand the implications of the tools and practices to knowledge conversions of the applications developers. Primarily, the data that were examined came from the weekly observations documents that contain both the narratives and interpretations. We also triangulated our interpretations based on the data that are found during the informal, conversational interviews. For example, when looking at Socialization (tacit-to-tacit), we conducted analysis as to how application developers gather requirements through discussion meetings; for Externalization (tacit-to-explicit), we looked at the experience such as through pair programming of writing the requirements into program codes or designs; for

Combination (explicit-to-explicit), we looked at practices such as code review that allow the program codes or designs to mature by doing enhancements to these codes and designs; and last, for Internalization (explicit-to-tacit), we also looked at practices such as code review and analysis of existing designs in which applications developers use to increase their tacit knowledge.

Second, discourse analysis is usually used to analyze spoken and written human communications in linguistics context. However, there are other research studies that are beginning to explore its value in other contexts such as what is now a growing research analysis technique called visual discourse, where images are interpreted according to the context of what is being studied. We used discourse analysis to analyze the conversations that are part of the data gathering, specifically document reviews. Specifically, we analysed the data from the chat history, email conversations, and online discussions from the online intranet site. With these analysis perspectives in mind, we followed a thematic analysis, which we patterned from the suggestion of Boaduo (2011). First, we coded the data by assigning labels to the raw data to link bits of data to ideas. Second, we categorized the codes by identifying how the codes will be grouped together to reflect the general idea of the data generally known as the categories/themes. By prioritizing and coming up of hierarchy to the categories/themes, we determined the relationships that exist between them, which constituted the findings section.

## **5 FINDINGS AND CONTRIBUTIONS**

This study confirms that knowledge matures and converts from one form to another; however we do not agree that knowledge conversions go through a unidirectional cycle defined in the SECI Model, in which the transitions from one mode to another are believed to be “smooth”. Instead, knowledge conversions happened due to several conditions and through non-straightforward steps in the process.

Tools are generally beneficial in guiding the participants in self-exploration of the tasks. For instance, the use of “breakpoint” that may be performed through the Visual Studio Editor versus the traditional Notepad++ allows them to trace the logic of the program code that they want to understand. This speeds up the acquisition of knowledge from program codes into one’s tacit understanding. The Participants also took advantage of the technical frameworks in the Subversion, a tool used for repository of documents, and the installation of Virtual Private Network access that allows developers to telecommute making resources available where there is Internet. The use of laptop being portable and with the permission to install other tools allows them to work offline even outside work hours. However, there are practices such as meetings that may not necessarily lead to a better understanding of requirements and eventually to successful delivery of software applications by the Participants. It is notable that there are Participants that acquired knowledge by trial and error method through self-study techniques and considered that meetings are not necessarily helpful, which is contrary to what the SECI cycle describes. It is therefore conclusive that while the effect of various factors such as cultural difference and preferences are strong at the initial phases, it is the same reason why combination and internalization were primarily focused on by the Participants leading to an argument that unlike the original model, knowledge creation does not necessarily start at the Socialization Phase.

A closer look at these tools in applications development indicates that these interactions overlap definitions within the entire SECI model. For one, due to non-availability of the members, being located in different areas and time zones, there is a lesser opportunity for the direct, face-to-face interactions to happen. Therefore, there is the need for the team members to construct their thoughts via email and online discussions most of the time. In the process, ideas need to be explicated in the forms of user interface prototypes and source codes, which fall under the Externalization phase. Again, the idea that knowledge creation starts with converting tacit-to-tacit knowledge does not necessarily hold true in this case. Certain features of tools such as in the online discussions show interesting effects too. The use of “like” similar to social networking sites like Facebook shows several implied meanings. It could mean agreement to one’s thoughts on ongoing debates. For some, it can mean approval or supportive of one’s arguments too. Others also use the like feature in order to get notified about the progress of the thread discussions, and even to show politeness. As others have liked their

posts, they return the favor by liking others' posts too. Another, metaphors are also used by the developers to convey their ideas relating to technical ideas. The use of "check in" and "commit", for instance, are used to mean that the files will be placed in the repository for reference. Metaphors help in articulating ideas and with the help of certain features of the chat tool where developers can share the screen. Ideas may therefore be demonstrated by visual aids too aside from simply having to specify all the details via phone alone. It is also evident that management support and flexibility in schedule play important roles in knowledge conversions. The management designated a US point person for the developers in the Philippines to raise concerns to. Due to geographical and time difference, the four-day workweek and flexibility in schedule allows the developers to have more overlaps in schedule.

The interface of emails compels a formal setting and is seen inappropriate for long threads of discussion while chat does not only offer instant messaging, which may also be saved, but also screen sharing capability. The intranet site may be considered somewhere in between being casual and formal. It is notable that posts from the Philippines mostly include completion of tasks and less on difficulties, which bring the aspect of Uncertainty Avoidance. There are also posts by the US developers that indicate the requirement of a code review to the works of developers in the Philippines. Though on paper, developers from both countries have the same level, in terms of position, there is a conscious or unconscious thought that US developers are better "knowledge experts", which bring evidences of Power Distance. Consequently, as code review sessions transform codes, for instance, this indicates that knowledge conversions in terms of explicit-to-explicit need to be validated by US developers.

In addressing the research question, this study has made contributions to theory, method, and practice. First, we hoped to have enriched the discussion on the Universalist-Pluralist debate that has been revolving around the Theory of Knowledge Creation and to the arguments on and/or against the SECI Model, specifically regarding the existence or non-existence of a cycle and the unidirectional or multidirectional properties of the conversions. There is also less empirical work done on how the Theory of Knowledge Creation may be applied, specifically the SECI Model as a process model, to modern domains such as in the field of ICT. Empirical studies on applications of traditional theories to modern domains, where they were not possibly designed and thought to work, such as in ICT, become useful references in examining the extent of many factors including the aspect of the theories' stability and applicability. This also paves the way for other theories to be formulated and for existing ones to be redefined to fit modern domains. Also, since much of the ICT research works are dominated by the positivist tradition, which is the view that aims to identify, generalize, and measure social structures through quantitative measures, the views we can get from other approaches provide significant knowledge in understanding and finding meanings about the subjective experience involved in examining ICT phenomena (Orlikowski and Baroudi 1991; Pablo 2007). Also, as stated earlier, much of the studies delved in the general application of the theory's conceptual model but there are little empirical studies conducted to examine the SECI Model as a process model, which leads to literatures suggesting that this can be better explored through qualitative investigation. This study may then provide an added value to methodological understanding of the SECI Model to re-examine the claim as a process model since even the proponents would suggest more qualitative investigations. Third, we also contribute to practice by presenting the value of theory-grounded know-how as opposed to mere gut-feel when faced to come up with decisions to address minor to critical issues. Specifically, this can be done by presenting the knowledge conversion concept defined in the Theory of Knowledge Creation as applied in applications development may be used and/or revised to effectively manage knowledge between ICT professionals such as application developers, quality assurance testers, and other possible sources of knowledge (tutorial videos, technical documents, etc.). By understanding the implications of the employment of tools and practices to the conversion of knowledge, ICT professionals become more aware of their vulnerabilities caused by internal (self) and external (environment) factors, which then become possible areas for improvements. The discussion of these implications also aims to give a richer understanding as to why even when there are the so-called "best practices" and defined methodologies, there is still a significant failure rate of projects, as well as a richer understanding of the other possible effects of the tools used as they are currently seen as being value-laden or just mere artifacts, and thus being value-neutral.

## References

- Adler, P. (1996). The Dynamic Relationship Between Tacit and Codified Knowledge: Comments on Ikujiro Nonaka's Managing Innovation as an Organizational Knowledge Creation Process. G. Pogorel and J. Allouche (eds.) *International Handbook of Technology Management*, Amsterdam: North-Holland.
- Aggestam, L. (2006). Learning Organization or Knowledge Management - Which Came First, the Chicken or the Egg? *Information Technology and Control*, 35 (3A), 295-302.
- Alipour, F., Idris, K. and Karimi, R. (2011). Knowledge Creation and Transfer: Role of Learning Organization. *International Journal of Business Administration*, 2 (3), 61-67.
- Andreeva, T. and Ikhilchik, I. (2011). Applicability of the SECI Model of Knowledge Creation in Russian Cultural Context: Theoretical analysis. *Knowledge and Process Management: The Journal of Corporate Transformation*, 18 (1), 56-66.
- Avram, G. (2007). Knowledge Work Practices in Global Software Development. *The Electronic Journal of Knowledge Management*, 5 (4), 347-356.
- Bailin, S. (1997). Software Development as Knowledge Creation. *International Journal of Applied Software Technology*, 3 (1).
- Barlattier, P., Vidou, G., Jacquemart, S. and Latour, T. (2006). Don't Value the Valueless: Toward a Model of Evaluation of Knowledge within E-Communities of Practice. *Proceedings of the 2nd International Workshop on Building Technology Enhanced Learning solutions for Communities of Practice*, 43-57.
- Becerra-Fernandez, I. and Sabherwal, R. (2001). 'Organizational knowledge management: a contingency perspective'. *Journal of Management Information Systems*, 18 (1), 23-55.
- Becker, K. (1997). Integrating Voting Techniques into a Discussion Rationale Model-based GDSS. *ISDSS'97*.
- Boaduo, N. (2011). Systematic Analysis and Interpretation of Collected Data for a Research Study: A Practical Methodological Framework for Writing Research Report. *Educational Research and Review*, 6 (2), 140-146.
- Booker, L., Bontis, N. and Serenko, A. (2008). The Relevance of Knowledge Management and Intellectual Capital Research. *Knowledge and Process Management*, 15 (4), 235-246.
- Brey, P. (2009). Values in Technology and Disclosive Computer Ethics. *The Cambridge Handbook of Information and Computer Ethics*, Ed. L. Floridi, Cambridge: Cambridge University Press.
- Camp, J. (1999). Democratic Implications of Internet Protocols. *The Information Society*, 15, 249-256.
- Carlson, J. (2010). Avoiding Traps in Member Checking. *The Qualitative Report*, 15 (5), 1102-1113.
- Cayaba, C. (2012). The Role of ICT in Knowledge Creation. *International Conference on Software and Information Management (ICSIM)*. Thailand.
- Chrissis, M., Konrad, M. and Shrum, S. (2003). *CMMI – Guidelines for Process Integration and Process Improvement*. Reading, MA: Addison Wesley.
- Choi, B. and Lee, H. (2002). Knowledge Management Strategy and its Link to Knowledge Creation Process. *Expert Systems with Applications*, Elsevier Science Ltd.
- Dai, Z. and Chen, M. (2007). Automatic Test Generation for Database-Driven Applications. *19th International Conference on Software Engineering and Knowledge Engineering (SEKE'07)*. Boston, USA.
- Dong, J. and Zhao, Y. (2007). Classification of Design Pattern Traits. *19th International Conference on Software Engineering and Knowledge Engineering (SEKE'07)*. Boston, USA.
- Endres, M., Endres, S., Chowdhury, S, and Alam, I. (2007). Tacit Knowledge Sharing, Self-Efficacy Theory and Application to the Open Source Community. *Journal of Knowledge Management*, 11 (3), 92-103.
- Extreme Programming, 2012. Pair Programming. Retrieved April 23, 2013, from the Extreme Programming website: <http://www.extremeprogramming.org/rules/pair.html>
- Ferguson, J. (2005). Bridging the Gap between Research and Practice. *Knowledge Management for Development Journal*, 1 (3), 46-54.

- Flanagan M., Howe, D. and Nissenbaum, H. (2008). *Embodying Values in Technology: Theory and Practice*. Information Technology and Moral Philosophy (eds. J. van den Hoven and J. Weckert). Cambridge: Cambridge University Press.
- Friedman, B. and Nissenbaum, H. (1996). Bias in Computer Systems. *ACM Transactions on Information Systems*, 14 (3), 330-347.
- Gall, M. D., Gall, J. P., & Borg, W. R. (2003). *Educational Research: An Introduction* (7th ed.). Boston: Allyn & Bacon.
- Gourlay, S. (2003). The SECI Model of Knowledge Creation: Some Empirical Shortcomings. 4th European Conference on Knowledge Management, Oxford, England.
- Groenewald, T. (2004). A Phenomenological Research Design Illustrated. *International Journal of Qualitative Methods*, 3 (1).
- Henninger, S. (1997). Case-Based Knowledge Management Tools for Software Development. *Journal of Automated Software Engineering*, 4 (1).
- Hofstede, G. (1980), *Culture's Consequences: International Differences in Workrelated Values*. Sage, Beverly-Hills, CA.
- Hong, J. (2010). Nonaka's Knowledge Creation Model: Universal or Particularistic? *International Conference on Organizational Learning, Knowledge and Capabilities (OLKC'10)*, Boston, US.
- Humphrey, W. (2005). Why Big Software Project Fail: The 12 Key Questions. *CrossTalk, J. Def. Software Eng.*, 18 (3), 25-29.
- Humphrey, W., & Kellner, M. (1989). *Software Process Modeling: Principles of Entity Process Models (CMU/SEI-89-TR-002)*. Retrieved April 23, 2013, from the Software Engineering Institute, Carnegie Mellon University website: <http://www.sei.cmu.edu/library/abstracts/reports/89tr002.cfm>
- Iacono, J., Brown, A. and Holtham, C. (2005). Research Methods – a Case Example of Participant Observation. *The Electronic Journal of Business Research Methods*, 7 (1), 39-46.
- Introna, L. (2005). Disclosive Ethics and Information Technology: Disclosing Facial Recognition Systems. *Ethics and Information Technology*, 7 (2), 75-86.
- Johari, J. (2009). Interpretivism in Information System (IS) Research. *Integration and Dissemination Research Bulletin*, 4, 25-27.
- Johnson, D. (1997). Is the Global Information Infrastructure a Democratic Technology? *Computers and Society*, 27, 20-26.
- Jorna, R. (1998). Managing Knowledge. *Semiotic Review of Books*, 9 (2).
- Kaur, R. and Sengupta, J. (2011). Software Process Models and Analysis on Failure of Software Development Projects. *International Journal of Scientific & Engineering Research*, 2 (2), 1-4.
- Khalili, H., Maleki, A. and Mirzahosseini, M. (2011). Innovation and Knowledge Creation as Parts of Knowledge Management. *International Conference on Innovation, Management and Service*.
- King, W. (2009). *Knowledge Management and Organizational Learning*. Annals of Information Systems 4, Springer Science+Business Media, LLC.
- Klein, H. and Myers, M. (1999). A Set of Principles for Conducting and Evaluating Interpretive Field Studies in Information Systems. *MIS Quarterly*, 23 (1), 67-94.
- Krefting, L. (1990). Rigor in Qualitative Research: The Assessment of Trustworthiness. *The American Journal of Occupational Therapy*, 45 (3), 214-222.
- Latour, B. (1992). Where are the Missing Masses? The Sociology of a Few Mundane Artifacts,' in *Shaping Technology/Building Society: Studies in Sociotechnical Change* (eds. W. Bijker and J. Law), Cambridge, MA: MIT Press.
- Letts, L., Wilkins, S., Law, M., Stewart, D., Bosch, J. and Westmorland, M. (2007). Guidelines for Critical Review Form: Qualitative Studies (Version 2.0). *Qualitative Review Form Guidelines*.
- Lim, N., Rodil, J. and Cayaba, C. (2007). Automatic Generation of Use Case Diagrams from English Specifications Document. 19th International Conference on Software Engineering and Knowledge Engineering (SEKE'07). Boston, USA.
- Maxwell, J. (2008). *Designing a Qualitative Study*. Applied Research Designs, Corwin: A SAGE Company.
- Miller, S. (1996) *Where is Cyberspace? Civilizing Cyberspace: Policy, Power, and the Information Superhighway*, New York: ACM Press.

- Mohamed, A. (2008). Capturing Software Engineering Tacit Knowledge. Proceedings of the 2nd conference on European Computing Conference (ECC'08:).
- Mohamed, A. (2010). Facilitating tacit-knowledge acquisition within requirements engineering. Proceedings of the 10th WSEAS International Conference on Applied Computer Science (ACS'10).
- Morisio, M., Egorova, E. and Torchiano, M. (2007). Why Software Projects Fail? Empirical Evidence and Relevant Metrics. International Conference on Software Process and Product Measurement (IWSM-Mensura'07).
- Nandhakumar, J. and Jones, M. (2002). Development Gain? Participant Observation in Interpretive Management Information Systems Research. *Qualitative Research: SAGE Publications*.
- Nixon, J. (2005). Impact of National Differences in Work Practices. *The Journal of Learning in higher Education*, 1 (1), 43-52.
- Nonaka, I. (1991). The Knowledge-Creating Company. *Harvard Business Review*. November-December 1991, pp. 96-104. Reprint: *Harvard Business Review: Managing for the Long Term*, July-August 2007, 162-171.
- Nonaka, I. (1994). A Dynamic Theory of Organizational Knowledge Creation. *Organization Science*, 5 (1), 14-37.
- Nonaka, I. and Konno, N. (1998). The Concept of 'Ba': Building a Foundation for Knowledge Creation. *California Management Review*, 40, (3), 1-15.
- Nonaka, I. and H. Takeuchi. (1995). *The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation*. Oxford University Press, Inc.
- Nonaka, I. and Toyama, R. (2003). The Knowledge-Creating Theory Revisited: Knowledge Creation as a Synthesizing Process. *Knowledge Management Research and Practice* 1, 2–10.
- Nonaka, I., Byosiere, P., Borucki, C. and Konno, N. (1994). Organizational Knowledge Creation Theory: A First Comprehensive Test. *International Business Review*, 3 (4), 337-351.
- Nonaka, I., Konno, N. and Toyama, R. (2000). SECI, Ba and Leadership: a Unified Model of Dynamic Knowledge Creation. *Long Range Planning*, 33, 5-34.
- Orlikowski, W. and Baroudi, J. (1991). Studying Information Technology in Organizations: Research Approaches and Assumptions. *Information Systems Research*, 2 (1), 1-28.
- Ospina, S. (2004). *Qualitative Research*. Encyclopedia of Leadership. SAGE Publications, Londo, Thousand Oaks CA.
- Pablo, Z. (2007). The Discursive Construction of Metaphors on Web-Based Portals: the Case of the Development Gateway. *Journal of Research in Science, Computing, and Engineering*, 4 (2), 73-84.
- Polanyi, M. (1966). *The Tacit Dimension*. Doubleday and Company, Incorporated, Garden City, New York.
- Rice, J. and Rice, B. (2005). The Applicability of the SECI Model to Multi-Organisational Endeavours: An Integrative Review. *International Journal of Organisational Behaviour*, 9 (8), 671-682.
- Singh, N., Zhao, H. and Hu, X. (2003). Cultural Adaptation on the Web: A Study of American Companies' Domestic and Chinese Websites. *Journal of Global Information Management*, 11 (3), 63-80.
- SmartBear, 2012. 11 Best Practices for Peer Code Review. Retrieved April 23, 2013, from the SmartBear website:  
[http://support.smartbear.com/pdf/11\\_Best\\_Practices\\_for\\_Peer\\_Code\\_Review.pdf](http://support.smartbear.com/pdf/11_Best_Practices_for_Peer_Code_Review.pdf)
- Tavani, H. (1999). Informational Privacy, Data Mining, and the Internet. *Ethics and Information Technology*, 1 (2), 137-145.
- The Standish Group International Inc. (2001). *Extreme Chaos*. The Standish Group International, Inc.
- Tilmann, G. and Weinberger, J. (2004). Technology Never Fails, But Project Can. *Baseline*, 1 (26), 28.
- Tsoukas, H. (2003). Do We Really Understand Tacit Knowledge?. In Easterby-Smith, M. and Lyles, M.A. (Eds), *The Blackwell Handbook of Organizational Learning and Knowledge Management*. Malden, MA; Oxford: Blackwell Publishing Ltd, 410-427.

- Turner, D. (2010). Qualitative Interview Design: A Practical Guide for Novice Investigators. *The Qualitative Report*, 15 (3), 754-760.
- Walsham, G. (1995). Interpretive Case Studies in IS Research: Nature and Method. *European Journal of Information Systems*, 4 (2), 74–81.
- Weber, R. (2004). The Rhetoric of Positivism Versus Interpretivism: A Personal View. Editor's Comments, *MIS Quarterly*, 28 (1), 3-12.
- Williams, R. and Edge, D. (1996). The Social Shaping of Technology. *Research Policy*, 25, 856-899.
- Xu, D., Xu, W. and Wong, W. (2007). Automated Test Code Generation from UML Protocol State Machines. 19th International Conference on Software Engineering and Knowledge Engineering (SEKE'07). Boston, USA.
- Yamamoto, S. (2008). Special Feature: Knowledge Creation Design Methodology for Service Innovation. (Nippon Telegraph and Telephone Public Corporation) *NTT Technical Review*, 6 (7), 1-5.