JYVÄSKYLÄ STUDIES IN COMPUTING
203

Sukanya Khanom

On Enriching Requirements Engineering Work Using Icons

JYVÄSKYLÄN YLIOPISTO

JYVÄSKYLÄ STUDIES IN COMPUTING 203

Sukanya Khanom

On Enriching Requirements Engineering Work Using Icons

Esitetään Jyväskylän yliopiston informaatioteknologian tiedekunnan suostumuksella
julkisesti tarkastettavaksi yliopiston Agora-rakennuksen Gamma-salissa
joulukuun 10. päivänä 2014 kello 12.

Academic dissertation to be publicly discussed, by permission of
the Faculty of Information Technology of the University of Jyväskylä,
in building Agora, Gamma hall, on December 10, 2014 at 12 o'clock noon.

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2014

On Enriching Requirements Engineering Work Using Icons

JYVÄSKYLÄ STUDIES IN COMPUTING 203

Sukanya Khanom

On Enriching Requirements Engineering Work Using Icons

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2014

Editors
Timo Männikkö
Department of Mathematical Information Technology, University of Jyväskylä
Pekka Olsbo, Ville Korkiakangas
Publishing Unit, University Library of Jyväskylä

URN:ISBN:978-951-39-5974-6
ISBN 978-951-39-5974-6 (PDF)

ISBN 978-951-39-5973-9 (nid.)
ISSN 1456-5390

Jyväskylä University Printing House, Jyväskylä 2014

ABSTRACT

Khanom, Sukanya
On Enriching Requirements Engineering Work Using Icons
Jyväskylä: University of Jyväskylä, 2014, 77 p. (+ included articles).
(Jyväskylä Studies in Computing
ISSN 1456-5390; 203)
ISBN 978-951-39-5973-9 (nid.)
ISBN 978-951-39-5974-6 (PDF)
Finnish summary
Diss.

Requirements engineering (RE) is one of the core elements of software development, directly contributing to its success. The development of effective RE activities depends on the quality of intercommunication between various stakeholders such as users and technical staff. The need for communication between those who have also been influenced by different cultural backgrounds, including differences in spoken language, age, gender, and nationality, has become more relevant.

This thesis introduces an alternative approach for communicating requirements in the RE domain. This approach is built and evaluated following four design science research phases. First, this study seeks to overcome these challenges faced within RE: (1) requirements identification, which refers to the ability of users to express their needs explicitly and concisely, (2) requirements specification, which is associated with difficulty in understanding and reviewing requirements, and (3) managing requirements, which relates to the effort used to manage, monitor, and continue tracing the requirements in the life cycle. Second, artifacts are developed in relation to an icon-based approach to enriching RE work. A piece of RE contexts defined will be represented by icons that correspond to cultural elements. Third, demonstrators portraying the icon-based concept are invented though MediaWiki. Such demonstrators are empirically evaluated in a controlled environment that permits participants to study the built artifacts within two iterations: (1) the formative evaluation, which is handled by students and the data from which is carried to the later iteration for further improvement, and (2) the summative evaluation, in which experts in the field and PhD students are involved. Fourth, the results are delivered to the target audiences.

The results of evaluations show that icons enable the enrichment of RE work. Users’ perceptions and satisfactions toward the representation of a piece of a requirement artifact by icons are significantly stronger than average. In addition, different background stakeholders, both novices and experts, are able to interpret and perceive icons with no statistical difference.

Keywords: requirements engineering, icon, design science research

Author's address Sukanya Khanom
Department of Mathematical Information Technology
University of Jyväskylä, Finland
P.O Box 35 (Agora), 40014 University of Jyväskylä, Finland
sukanya.s.khanom@student.jyu.fi

Supervisors **Senior Lecturer** Anneli Heimbürger
Faculty of Information Technology
University of Jyväskylä, Finland

Professor Tommi Kärkkäinen
Department of Mathematical Information Technology
University of Jyväskylä, Finland

Reviewers **Professor** Jyrki Nummenmaa
School of Information Sciences
University of Tampere, Finland

Professor Markku Oivo
Department of Information Processing Science
University of Oulu, Finland

Opponent **Professor** Hannu Jaakkola
Department of Information Technology
Tampere University of Technology, Finland

ACKNOWLEDGEMENTS

It seems that my four-year journey as a postgraduate student has already flown swiftly by. I have discovered that pursuing a postgraduate degree is not easy, as it is all about questions and answers. This was particularly true for me, as I was living in a completely different environment, with a different culture and language, making the journey even more complicated. Yet there were always people who inspired me with mental and physical support, who spent huge amounts of time just to pull me up when I felt like I was drowning. I met several people who did this for me along my way, and I would like to thank all of them for their inspiration, motivation, and encouragement.

First, I would like to give sincere thanks to my supervisor, Senior Lecturer Anneli Heimbürger, who provided me with encourage, guidance and support from the initial stages to the completion of this study and who showed enthusiasm and faithfulness to my work even when I made mistakes. When I first began to pursue this degree, I studied a topic about which I had very little background knowledge. As I made the decision to select a new topic of study, outside the realm of mathematics, Anneli's name kept coming to my mind. I got to know her in Requirements Engineering class. My perspective of her, even before becoming her student, was that she is a dynamic person who always understands her students' problems and situations. More and more, it became apparent that she could understand how multicultural students must be feeling while pursuing studies in a foreign land. After deciding to ask her to be my supervisor, I have never regretted the decision. She is a cheerful and positive supervisor who has been a source of constant support, offering invaluable assistance and guiding me in the right direction. I would like to thank Senior Lecturer Anneli Heimbürger for all the suggestions and discussions and for providing me with great opportunities throughout my study. In addition, I would like to thank Professor Timo Hämäläinen, who initially gave me the chance to visit Finland as a PhD student by accepting me to the program. Even though after my first year I was not under his supervision any more, I want to thank him for the opportunity he provided to me.

I would like to express my sincerest gratitude to another supervisor of mine, Professor Tommi Kärkkäinen, who always directed me onto the right track. He always arrived at meetings with great, insightful ideas that opened the window to academia for me. Professor Tommi Kärkkäinen produces ideas at such a rate that it sometimes took me many days or weeks to digest. Yet our conversations always stimulated novel directions for my work. Thank you for all the thought and care you invested in helping me along in my academic journey.

I would also like to take the opportunity to thank M. G. S. Sriyananda. Without him to help me through the first stages of my PhD student's life, it is possible that I would have headed straight back to Thailand. At that time, my attempts at academic research were akin to an infant's first endeavors to walk; I did not know how to construct articles, and I was unaware of even simple research methods. He taught me everything from the basic rules of research to complex methods required of graduate students, and I must give a million thanks to my friend.

I was very fortunate to meet Porjai Laithongkam, who has never left me behind, helping me adapt to my new surroundings. He was always patient with me and all the trials and triumphs I experienced throughout my PhD. The life as a PhD student can be difficult, and Porjai was always there to comfort me in difficult times and celebrate with me during times of success. Porjai contributed greatly to ensuring balance in my life, a task that he performed very well.

I would especially thank to P. Chanoknet Commong and Piyanuch Pokeaw for the cover painting of this thesis and Yuwarat Srisupawong for her assistance in database design. Many thanks also go to my master collaborator, Tapio Keränen, who developed the first demonstrator. My warmest thanks to reviewers (Professor Markku Oivo and Professor Jyrki Nummenmaa), and opponent (Professor Hannu Jaakkola).

Special thanks go to my dear friends, Patcharin, Mingmie, Piyanuch, Yuwarat, Ratanaporn, Surachai, Yuwathida, Patima, Manu, Wichanan, Watinee, Thitiporn, Apinya, Puttinan, Amornrat, Jakapan, Kanoktip, Cindy, Peerapong, and countless others for keeping me cheerful, listening to my problems, and celebrating with me in my success. You guys have always been awesome friends and are always there for me. I would also thank those who supported me both mentality and physicality during my stay in Finland, P. Noi, P. Wilai, P. May, P. Jin, P. Juk, and P. Sorn.

Finally, my heartfelt thanks go to my parents, Prayoon and Suwalee Khanom; my two dear brothers, Khanchit and Patiwathakarn Khanom; my sister-in-law, Wisansaya Khanom; my niece, Thanyamai Khanom; and my aunt, Nantanut Wuttiwong. Thank you for your unconditional love and support and for believing in me for my entire life. I do not know where I would be without them. They created such a nurturing environment and cared for my whole life. Since I was a child, I have heard my father say, "*It would be great if there could be at least one in our family achieve a PhD, the highest level of education.*" This motivated me to pursue a PhD, and I am glad to be able to now say to my father, "*I have achieved it!*" Thank you for your inspiration.

Jyväskylä 17.09.2014
Sukanya Khanom

ACRONYMS

BR	Business Requirement
CORE	Control Requirements Expression
DSR	Design Science Research
GORE	Goal-Oriented Requirements Engineering
GUI	Graphical User Interface
HCI	Human-Computer Interaction
IDV	Individualism
IS	Information System
IT	Information Technology
LTO	Long-Term Orientation
MAS	Masculinity
OO	Object-Oriented
OWL	W3C Web Ontology Language
PDI	Power Distance Index
PR	Product Requirement
RAS	Requirements Attribute Schema
RD	Requirements Development
RE	Requirements Engineering
RM	Requirements Management
RQ	Research Question
SADT	Structured Analysis and Design Techniques
SD	Strategic Dependency
SDLC	Software Development Life Cycle
SR	Strategic Rationale
SRS	Software Requirements Specification
UAI	Uncertainty Avoidance Index
UFM	User Preference Modeling
UML	Unified Modeling Language
UR	User Requirement
VORD	Viewpoint-Oriented Requirements Definition Model

FIGURES

TABLES

CONTENTS

LIST OF INCLUDED ARTICLES

AI. Khanom, S., Heimbürger, A. & Kärkkäinen, T. 2012. Icon-based Language in Requirements Development. In Y. Kiyoki, T. Tokuda & N. Yoshida (Eds.) 22nd European Japanese Conference on Information Modeling and Knowledge Bases. Prague, Czech Republic, June 4-9, 20.

AII. Khanom, S. 2013. Icon-based Language in the Context of Requirements Engineering. In R. B. Svensson, D. M. Berry, M. Daneva, et al. (Eds.) 19th International Working Conference on Requirements Engineering: Foundation for Software Quality (REFSQ). Essen, Germany, April 8-11, 215.

AIII. Khanom, S., Heimbürger, A. & Kärkkäinen, T. 2014. Icon Representations in Supporting Requirements Elicitation Process. In T. Tokuda, Y. Kiyoki, H. Jaakkola & N. Yoshida (Eds.) Information Modeling and Knowledge Bases XXV. IOS Press, Amsterdam, 133-145.

AIV. Heimbürger, A., Duzi, M., Kiyoki, Y., Sasaki, S. & Khanom, S. 2014. Cross-Cultural Communication with Icons and Images. In T. Tokuda, Y. Kiyoki, H. Jaakkola & N. Yoshida (Eds.) Information Modeling and Knowledge Bases XXV. ISO Press, Amsterdam, 306-322.

AV. Khanom, S., Heimbürger, A. & Kärkkäinen, T. 2013. Icon-based Language: Auxiliary Communication for Requirements Engineering. International Journal of Engineering Science and Technology 5, 1076-1082.

AVI. Khanom, S., Heimbürger, A. & Kärkkäinen, T. 2013. Icons: Visual Representation to Enrich Requirements Engineering Work. Journal of Software Engineering and Applications 6 (11), 610-622.

AVII. Khanom, S., Heimbürger, A. & Kärkkäinen, T. 2015. Icons Recognition and Usability for Requirements Engineering. In Information Modeling and Knowledge Bases XXVI. ISO Press, Amsterdam, (in press).

AVIII. Khanom, S., Heimbürger, A. & Kärkkäinen, T. 2014. Can Icons Enhance Requirements Engineering Work? Journal of Visual Languages & Computing (submitted).

Sukanya Khanom has contributed as the main author for each of these articles except [AIV]. Co-authors made substantial contributions to the conception, design, and empirical analysis, actively drafting and revising all the joint articles. For [AIV], while Anneli Heimbürger is the main author, Khanom conducted one section pertaining to two examples of application domains for icons.

1 INTRODUCTION

1.1 Background and Motivation

Requirements Engineering (RE) is one of the core aspects of the software development process and stands at the heart of enterprises that successfully make software-intensive systems (Hofmann & Lehner, 2001). RE itself entered into research communities just a few decades ago (Hansen & Lyytinen, 2010) and is a communication-rich and user-involved process (Bano & Zowghi, 2013). Principally, it deals with recognizing, articulating, and understanding needs to arrive at a definition of requirements. Within the realm of RE, it systematically exploits certain standards (e.g., IEEE) and techniques (e.g., Unified Modeling Language (UML), and goal-oriented models) for requirements elicitation, analysis, negotiation, specification, and management (Pohl, 2010; Wiegers & Beatty, 2013).

1.1.1 Why is Requirements Engineering Important?

Nowadays, software systems have penetrated into large segments of our lives, such as financial services, medical sectors, the automobile industry, mechanical and electrical engineering, and even crisis management (Pohl, 2010; Yang et al., 2014). This makes the development of software-intensive systems even more of a necessity and a challenge. The main challenge facing this field is that users increasingly demand innovative features. Thus, the complexity of software systems has significantly surged due to the increasing number of integration variants to fulfill users' desires and an attempt to reduce cost while increasing productivity.

Considerable effort has been devoted to improving the development of software-intensive systems. Acting as a branch of software engineering, RE is the most essential influencer to the success or failure of a system. Various studies state that the problems that result from of incomplete RE can be propagated through a system and affect other components, even those that are independent in the system (Mathiassen et al., 2007; Moody, 2009; Pohl, 2010; Wiegers & Beatty, 2013; Zave, 1997).

Figure 1 depicts the different reasons for which failure occurs in software intensive projects and the frequency (%) at which each of these occurs, providing a tangible way to understand the most common challenges faced (The Standish Group, 2013). This data is based on the results of a survey asking IT executive manages their opinion about why a project was impaired. The reasons listed (grey pies) can often be attributed to insufficient or poor RE, accumulated up to 44 %. The success or failure of a software-intensive project, therefore, depends on RE processes; in these processes, the stakeholders should be actively engaged. Further, interviews with software companies conducted by El Emam & Koru (2008) reveal that one of most common reasons for the cancelation of a project is inadequate requirements at the delivery decision. This is again confirmed in a study by Cerpa & Verner (2009), who showed that the main factor of project failure is owing to the effect when performing requirements engineering activities.

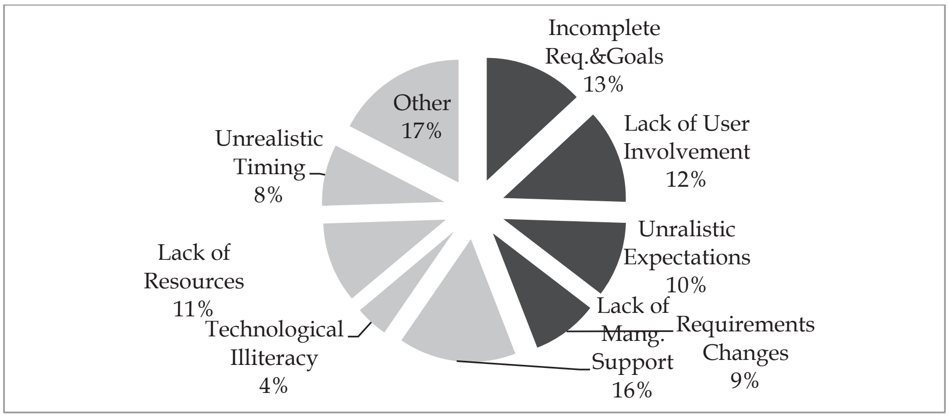

FIGURE 1 Reasons for failures in software-intensive projects.

Another way to understand the significance of RE is by observing the cost of correcting errors. Frequently, requirements defects are detected soon after the system development phase. Using the plan-driven process (e.g., a waterfall model) if requirements defects are uncovered during requirements development (RD) phases, the relative cost of fixing the defect is nearly 100 times less expensive than it is for finding and correcting them after delivery (Boehm & Basili, 2001; Boehm, 2006). Certain studies (e.g., Hall et al., 2002) estimate that the percentage of errors occurring in the release of software projects as a result of requirements challenges is nearly 48% of all problems detected. Hence, software development teams aim to find and fix problems as early as possible.

Many scientists put much effort into determining why RE activities are not yet fulfilled by existing methodologies. For instance, Boehm and Basili (2001) found that a less formal (semi-formal or informal) method consistently makes software systems more efficient while simultaneously tackling requirements correctly earlier than later. Similarity, other authors (Moody et al., 2010; Moody & Hillegersberg, 2009; Morris & Spanoudakis, 2001) claim that languages whose graphical syntax is easily understood are less likely to cause requirements problems. In recent studies (e.g., Bera & Evermann, 2014; Hadar et al., 2013; Horkoff & Yu, 2013; Li et al., 2014; Svahnber et al., 2013), researchers intricately examined RE models to understand their comprehensibility when specifying requirements. The heuristic intent is to advance the models upon which most notations are customarily constructed by abstract formalization wherein different interpretations can occur. Creating effective visual notation that is easily-recognizable may foster simple, intuitive, and executable visual notations to support systems that are user-friendly for both technical and non-technical users (Moody et al., 2010). Icons are recommended for use because they make diagrams more intuitive and understandable. Through these advantages, icons may help to overcome the challenges identified in the research and remove the barriers caused by technical-rich methods, such as misinterpretation and misunderstanding (Yu & Junda, 2010).

1.1.2 Why are Icons Essential?

As the world becomes smaller through sophisticated technology, the need for communication among people of different backgrounds has become even more critical. Languages afford us the ability to communicate with each other in a relatively unproblematic and straightforward way. However, the use of native languages is not ideal, as a single language may not be comprehensible to all stakeholders, who come from myriad backgrounds (Nakamura & Zeng-Treitler, 2012).

The use of icons, therefore, has become an intuitive mechanism for human communication. As an example, traffic lights and their colors (green = go, yellow = prepare to stop, and red = stop) are now recognized in almost all countries. This system of symbols has become a simple language stemming from the desire to communicate among different people of different backgrounds knowledge bases (Aykin, 2005; Heimbürger et al., 2012).

Traditionally, icons have been used for formulating and recording thoughts and ideas (Barker, 2000; Marcus, 2003; Fitrianie et al., 2007). Because of their communicative power, icons are now utilized as a commonplace (McDougall & Isherwood, 2009) in a wide variety of domains to inform people about particular conditions (Salman et al., 2012). Over the past few years, the growing sophistication of computer-intensive system has given rise to the birth of graphical user interfaces (GUIs) (Marcus, 2003). A computer interface language is often composed of icons, which can considerably reduce the amount of information that needs to be conveyed. Icons are increasingly used in software packages (e.g., Macintosh and Windows). Nevertheless, being able to understand and communicate using icons has not yet been realized in the area of RE.

When hundreds of members engage in developing software applications, simplicity is vital. In particular, within the RE process, it is impossible to sit down with users and figure out what they want a system to do. Making use of software languages and tools would naturally allow for the production of higher-quality software that meets stakeholders real needs, nonetheless there continues to be much room for improvement, as the RE techniques being used are still not in perfection (Rolland et al., 1998; Carrillo et al., 2012). Bearing this in mind, existing and new RE techniques must constantly be improved. This need has prompted the present research, which examines the ways in which icons can enrich RE work.

1.2 Scope of Study

There has been growing interest in solving RE difficulties because inappropriate RE can be costly and cause other problems in later phases of software development (Brun et al., 2013; Hansen & Lyytinen, 2010; Kaiya et al., 2005). Industries have gradually allocated more weight to the formulation of requirements in the early phases of system development (Pitula & Radhakrishnan, 2011; Pohl, 2010). Widely accepted, effective requirements can be achieved through effective communication among a vast number of stakeholders (Al-Rawas & Easterbrook, 1996; Fitrianie & Rothkrantz, 2007). Therefore, this study investigates icon notation to assist in communicating requirements between users of different backgrounds. Figure 2 explicates the research areas of this dissertation.

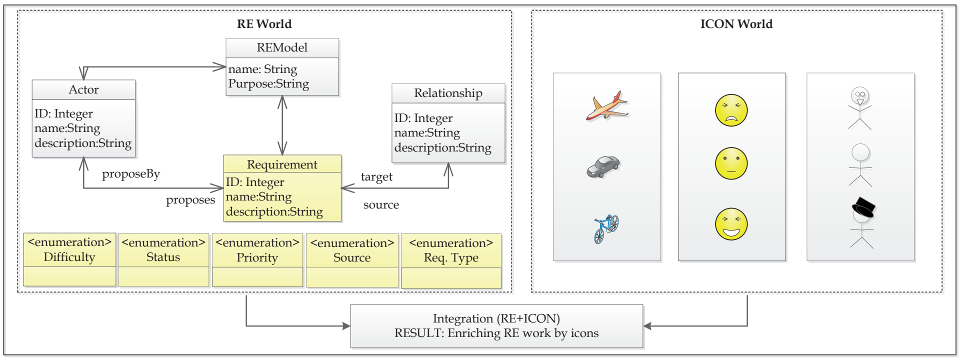

FIGURE 2 Research areas of this dissertation.

The objectives of this study are as follows: (1) to construct artifacts associated with enriching RE work using icons and (2) to evaluate these defined artifacts for the practicality of their applicability. The research into RE and icon-based information, together with an evaluation of the research, will contribute (1) conceptual models designed for an icon-based approach and (2) formative and summative results that verify whether or not an approach is able to enhance participants' comprehension.

To perform the work of this study, a design science methodology (Hevner et al., 2004; Peffers et al., 2007) is followed, consisting of four phases: (1) understanding the environment, (2) building artifacts, (3) evaluating artifacts, and (4) commutating outcomes into publications. Table 1 designates the main phases of the research and questions studied in this dissertation.

TABLE 1 Main research phases and questions studied in this dissertation, including related articles.

Research Phase	Research Question (RQ)	Publications (Phase 4)
Phase 1: Understanding the environment	• What are the most common challenges in performing RE work based on existing literature? • How can RE work be supported by iconic communication?	AI, AII AI, AII
Phase 2: Building a conceptual model to enrich RE work using icons	• How specific tasks of RE could be supported by icons? • How could multi-culturality be supported in icon design for RE?	AIII, AV, AVI AIV, AVI
Phase 3: Evaluating the constructions	• How to evaluate a proposed solution to confirm if it is usable for multifaceted stakeholders?	AVI, AVII, AVIII

1.3 Dissertation Structure

The overall organization of the dissertation is depicted in Figure 3.

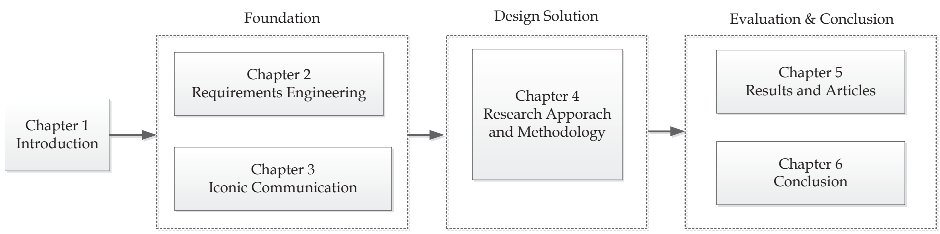

FIGURE 3 Structure of this dissertation.

Chapter 1 introduces why RE and the use of icons within this field are imperative areas of study.

Chapter 2 provides further information about RE, first detailing general definitions of relevant terms and then outlining the RE process from elicitation to management. Next, it points out the current challenges facing RE and various approaches that have proven useful in tackling them; the chapter ends with a summary.

Chapter 3 introduces an icon-based language, which is a promising tool that has already been used in a variety of domains. In this chapter, the current iconic communication environment is presented and the potential applicability of icons for supporting RE is described.

Chapter 4 deals with questions, problems, and approaches that support this research. The research methodology employed to obtain the research concept is also described. Its relevant artifacts are also expressed, in sequence, and an evaluation and demonstration of the concept of icon-based language are provided.

Chapter 5 presents a summary of the included articles and results, while Chapter 6 gives a conclusion and notes the limitations of the work. Further research is also suggested.

2 ON REQUIREMENTS ENGINEERING

RE is a fundamental part of every software project, defining what all relevant stakeholders in a system need and what the system needs to handle in order to fulfill stakeholders' desires (Agarwal et al., 2010; Ballejos & Montagna, 2011).

2.1 Requirements Engineering Definition

The following definitions describe relevant terms, from what a requirement itself means to what RE means.

2.1.1 Definition of a Requirement

Definition by (IEEE Std 1220-1998, 1998): *"A statement that identifies a product or process operational, functional, or design characteristic or constraint, which is unambiguous, testable or measurable, and necessary for product or process acceptability (by customers or internal quality assurance guidelines)"*.

Definition by (IEEE Std 610.12-1990, 1990): *"(1) A condition or capability needed by a user to solve a problem or achieve an objective. (2) A condition or capability that must be met or processed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents"*.

2.1.2 Definition of a Stakeholder

Stakeholders are the primary source of requirements. The key stakeholders in a project are usually customers, sponsors, or end-users including regulatory bodies and other external parties.

Definition by (ISO/IEC/IEEE 29148, 2011): *"Individual or organization having a right, share, claim, or interest in a system or in its possession of characteristics that meet their needs and expectations"*.

2.1.3 Definition of Requirements Engineering

RE deals with understanding external conditions, determining what capabilities the proposed system must possess in order to conform to these external conditions, and documenting those capabilities as requirements for a system.

Definition by (ISO/IEC/IEEE 29148, 2011): *"RE is an interdisciplinary function that mediates between the domain of acquirer and supplier to establish and maintain the requirements to be met by the system, software or service interest"*.

Definition by (Zave, 1997): *"RE is a branch of software engineering concerned with the real-world goals for functions of and constraints on software systems. It is also concerned with the relationship of these factors to precise specification of software behavior, and to their evolution over time acorss software families"*.

2.2 Requirements Engineering Process

RE involves all life-cycle activites, which include discovering, eliciting, analyzing, communicating, documenting, validating, and managing requirements (Hofman & Lehner, 2001; Cheng & Atlee, 2007; Svahnberg et al., 2013). Figure 4 is a portrayal of RE, which fundamentally consists of requirements development (RD) and requirements management (RM).

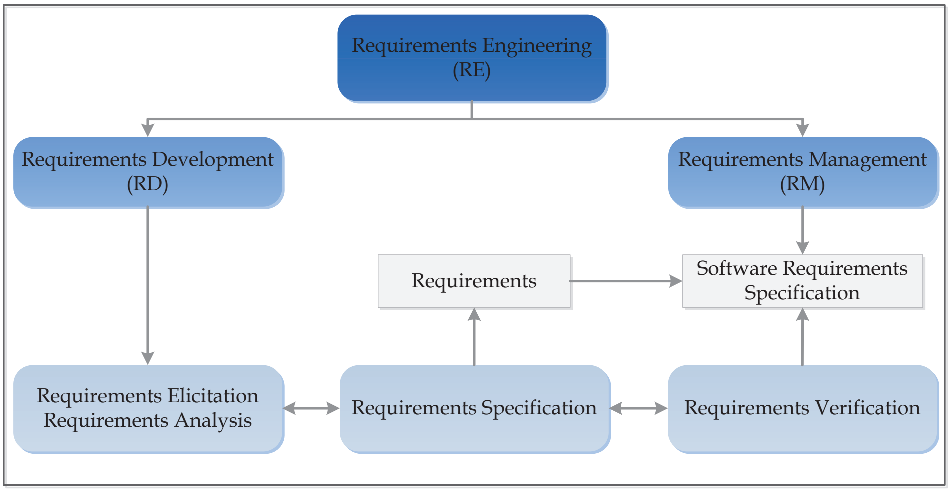

FIGURE 4 Subcomponents of the RE domain.

2.2.1 Requirements Development

The objective of RD is to capture a set of requirements that are explicit enough for all team members on the project (Wiegers & Beatty, 2013). RD can be further described in four distinct tasks, which will be outlined below (Agarwal et al., 2010; Pohl, 2010).

2.2.1.1 Requirements Elicitation

Requirements elicitation is the shared activity of realizing and understanding the stakeholders' desires. This activity requires involvement with multifaceted stakeholders (Luna et al., 2011) and is therefore perhaps one of the most complex and communication-intensive aspects of software development (Coughlan & Macredie, 2002; Sutton, 2000). It is a collaborative process between users and development teams. The process of elicitation itself is frequently composed of iterative tasks depending on the communicative skill of engineers and the commitment and cooperation of all the stakeholders regarding the requirements (Agarwal et al., 2010; Pohl, 2010). The goals of requirements elicitation are to identify relevant requirement origins, to elicit existing requirements, and to develop new and innovative requirements. Many techniques are used extensively in requirements elicitation and are constantly being put into practice; such techniques include questionnaire, introspection, observation, scenarios, brainstorm, focus group, joint application development, workshops, prototypes, viewpoints, task analyses, domain analyses, and knowledge acquisition techniques (Sommerville, 2011; Zowghi & Coulin, 2005).

2.2.1.2 Requirements Analysis

Requirements analysis is a vital activity, performed after elicitation to analyze the needs of users to obtain details about the requirements. In other words, requirements analysis is an activity that deepens the understanding of needs gathering from the elicitation phase. Each requirement is analyzed to achieve completeness, consistency, and feasibility for a firm understanding of the specifications (Agarwal et al., 2010; Wiegers & Beatty, 2013). In additions, a requirements analysis seeks to determine a system's functionality and non-functionality. Analyzed requirements may be used in negotiations to resolve any conflicts that might arise among stakeholders regarding requirements. It includes the activity of deciding which of the requirements can be accepted or rejected and setting requirements' priority levels. In this phase, it is often the case that visualizations are utilized to aid in the portrayal of certain aspects of the requirements.

2.2.1.3 Requirements Specification

Requirements specification is the activity that structures the collection of information gathered in the requirements analysis process into a document that outlines a set of requirements (Sommerville, 2011, Wiegers & Beatty, 2013). When inputs from several stakeholders are gathered, as is more often the case, these inputs are likely to be inconsistent. Requirements specification, then, translates the ideas from the various users into a set of requirements that are eventually contributed as a part of the software requirements specification (SRS). A good SRS, as defined by the (IEEE Std 830-1998, 1998) must conform to the following eight criteria: correctness, unambiguity, completeness, consistency, verification, prioritization, modification, and traceability. During requirements specification, formal and informal techniques can be applied, depending on the circumstances.

2.2.1.4 Requirements Verification

Requirements verification is the process of assuring that requirements are in compliance with the actual needs of the stakeholders. Moreover, it examines requirements specification to ensure that all system requirements have been stated unambiguously, resulting in consistent, complete, and realistic agreements. It iteratively reviews the variant of the following question "Are we building the right system?" (Pohl, 2010; Sommerville & Kotonya, 1998; Wiegers & Beatty, 2013). The verified requirements enable the development team to build a solution that satisfied the business goals.

2.2.2 Requirements Management

In principle, requirements management (RM) refers to the management activities in the RE process (Pohl, 2010; Sommerville & Kotonya, 1998). A vast amount of requirements are elicited, negotiated, documented, and validated during RE. In addition, the stakeholders' understanding of the system is constantly evolving. The system requirements must be changed, thus, to reflect this evolved view. Each requirement artifact has a number of basic properties, including priority, status, source, and difficulty and is related to other requirement artifacts. Because of this, the planning and controlling of the requirements in the RE process are inevitably needed. Typically, RM also continuously traces the relationships of all requirements and can be summarized as follows (Carrillo de Gea et al., 2011; Carrillo de Gea et al., 2012; Pohl, 2010; Wiegers & Beatty, 2013).

- Requirements change, versioning, and tracking management: the evolution of requirements should be maintained. During the life cycle of the system, the requirements are adapted as new requirements are added and existing requirements are deleted or modified. The versioning of requirements authorizes stakeholders to access or track the specific changes in the state of requirements over the system life cycle. It is also possible that stakeholders can assess the benefits of implementing new requests in relation to their cost and time of implementation (Pohl & Rupp, 2011).
- Requirements traceability management: the nature of requirements is somewhat dependent, which means that one requirement can relate to other requirements to a certain degree. Hence, recorded traceability information supports stakeholders in the discovery of possible relationships between requirements (Gotel & Finkelstein, 1994).

2.3 Current Requirements Engineering Challenges

During the past few decades, software project failure syndrome has not been overcome, despite the many techniques that have been introduced and developed for RE in software development projects (Cerpa & Verner, 2009; El Emam & Koru, 2008; Hansen & Lyytinen, 2010; Hofmann & Lehner, 2001; The Standish Group, 2013). However, while there is an intensifying attentiveness to the essence of RE, the development of a conclusive means of handling requirements is left primarily to intuition and remains largely unaccomplished. In RE, domain experts, who might not have a software engineering background, simply do not know what they want until they see or interact with the final result (Cheng & Atlee, 2007). There is currently an augmented stress on agile method that strongly engages users in the development process to receive rapid feedback on requirements and minimize documentation. The fundamental expectations of all development processes, plan-driven such as waterfall or agile method, are to derive a complete and explicit set of requirements from users and to detect unidentified needs or desires as soon as possible (Sommerville, 2011). It is plausible to contour RE challenges into these classifications.

2.3.1 Challenges to Requirements Identification

A significant goal of executing RE early in the software development life cycle (SDLC) is to establish the scope, boundary, and context of the system. Imprecise definition is a general problem faced when setting up system boundaries (Cheng & Atlee, 2007; McGee & Greer, 2012; Sateli et al., 2013). This is because unnecessary or irrelevant information can be provided to all stakeholders. For instance, stakeholders may fail to define a system based on their acquired or shared knowledge. The process of scope identification is often a difficult task because, for example, stakeholders' ambitions can be varied and conflicted, relying on their perspectives of the work circumstances or the tasks to be achieved (Hansen et al., 2010; Katina et al., 2014; Mathiassen et al., 2007).

Furthermore, RE is human-centric, involving communication and interaction between stakeholders to different degrees (Bano & Zowghi, 2013; Coughlan & Macredie, 2002; Pohl & Rupp, 2011). This communication directly influences the ability of system stakeholders to express their desires exactly and succinctly. Well-known obstacles to communication occur due to the diversity of professional expertise and organizational roles that separate individuals' views and expectation about the boundary of the system-to-be (Helming et al., 2010; Sutton, 2000). In sophisticated software project environments, development teams delegated to the process are widely diverse; for example, a development team includes requirements engineers, developers, and users. The involvement of stakeholders throughout the requirements process significantly improves the final system (Pitula & Radhakrishnan, 2011; Thanasankit & Corbitt, 2002; Yu et al., 1997). However, particular needs can be embedded within ambiguities, uncertainties, and gaps in knowledge (Hofman & Lehner, 2001; Van Lamsweerde, 2000). Thus, requirements must be declared accurately before they can be developed correctly.

2.3.2 Challenges of Requirements Specification

Specifying requirements is document-centric and can be very complicated to perform, especially for complicated systems in which a large number of requirements have been gathered and in which requirements relate to complex information (Carrillo de Gea et al., 2012). Requirements specification relies primarily on the capability of generating acquired information into a concrete and understandable representation. Practically, individual requirements engineers or system analysts synthesize the needs communicated by users and model them into the system requirements using informal, semi-formal, or formal means (Saiedian & Dale, 2000; Zhang et al., 2010). The model developed is normally based on the requirements engineers' personal understanding of the requirements they have synthesized (Hofman & Lehner, 2001). It is frequently the fact that when requirements engineers create a model within the context of RE, the structure of the model is typically readable only by requirements engineers themselves (Nuseibeh, 2001; Nuseibeh & Easterbrook, 2000). In contrast, people who are not technical professionals, unlike requirements engineers, might interpret such a model very differently (Vassev & Hinchey, 2013). Because of this, completing requirements without misunderstanding and misconception is not easy (Katina et al., 2014).

2.3.3 Challenges to Managing Requirements

Managing requirements during the course of the SDLC is one of the most critical aspects of the RE process, as failure in this stage has many adverse consequences that jeopardize the development of the software (Pohl, 2010; Wiegers & Beatty, 2013). Ideally, requirements for a software development are complete, consistent, and unambiguous before the first line of coding is begun. However, in reality, requirements change over time as users' needs evolve. One reason for these changes is that requirements are the outcome of the merits of multifaceted stakeholders, and those stakeholders often have conflicting desires and objectives (Ncube et al., 2013). If requirements changes happen repeatedly, a project can have a high rate of delivery delay or it may never be delivered at all. To avoid this, RE should be executed iteratively and progressively so that requirements can be reflected on and reworked to account for new knowledge acquired for entire activities. It is therefore important to gain insight into management's ability to accomplish visibility and accessibility (Kelanti et al., 2013) that allows (1) monitoring the progress of requirements, (2) managing and tracking the status of requirements, as well as their changes and version control, and (3) tracing individual requirements to their corresponding components or to other related requirements (Gotel & Finkelstein, 1994; Pohl, 2010; Wiegers & Beatty, 2013).

2.4 On Coping with Requirements Engineering Challenges

Modeling languages, which exploit some kind of graphical notation, have generally come to serve as a visual means for supporting RE tasks. Commonly, the model is used for recurrently and iteratively building graphical semantic representations of the concepts, attributes, and relationships (Costagliola et al., 2004). In the RE domain, models support the analysis and design process for requirements elicitation, analysis, negotiation, and specification. This section considers a number of modeling languages that can be found in RE arena.

2.4.1 Goal-Oriented Requirements Engineering

Goal-oriented RE (GORE) models have become a standard method for eliciting, refining, analyzing, negotiating, validating, and documenting requirements (Van Lamsweerde, 2001; Lapouchnian, 2005; Pohl, 2010; Rolland et al., 1998). It provides the kind of abstraction that addresses the problems and issues of RE. Usually, stakeholders understand better their overall goal they want to achieve than the functionality that should be employed by the desired system (Pohl, 2010). Thus, it is imperative to understand the structure of goals in deriving requirements from a starting point. The intention of proposing a GORE model is to bridge the gaps in understanding of the system requirements between requirements engineers and non-technical stakeholders (Dardenne et al., 1993; Duboc et al., 2013; Horkoff & Yu, 2013; Lapouchnian, 2005). Also, goal modeling offers the fundamental utilities for conflict detection and management among requirements and supports a notable way of transferring requirements to users (Ur Rehman et al., 2010; Van Lamsweerde, 2001). Table 2 summarizes two remarkable GORE frameworks. KAOS and *i** relate to GORE in that they provide a framework to consider organizations' and actors' goals as the source of requirements and deal with intentionality and dependencies among different actors.

TABLE 2 Summary of relevant frameworks of goal-oriented model.

Framework	Description
KAOS	A multi-paradigm language with a rich set of formal analysis techniques. KAOS provides several sub models for eliciting, specifying, and analyzing goals, requirements, scenarios, and responsibility assignments (Dardenne et al., 1993; Pohl, 2010; Van Lamsweerde, 2001). In KAOS, a goal is defined as a prescriptive statement of intent that the system should fulfill.

Continued

TABLE 2 (Continued).

i*	One of the most modern visual notations in RE. The *i** framework is an inclusive model for documenting and analyzing goals and goal dependencies. The *i** model is defined only by diagrams, rather than diagrams with supplemental text. It uses two diagram types to document requirements at different levels of abstraction: strategic dependency (SD) and strategic rationale (SR) diagrams (Dardenne et al., 1993; Pohl, 2010; Van Lamsweerde, 2001). The *i** framework is a comprehensive approach for documenting and analyzing *who* and *why* (the actors and the goals to be achieved). It focuses on dependencies between actors in order to accomplish the goal.

2.4.2 Object-Oriented Model

During the past three decades, plentiful object-oriented (OO) models have arisen, contributing to two chief purposes: OO analysis and design. OO analysis, in similar purpose to OO design, was first announced by Coad and Yourdon as a structured design method to guide developers who seek to implement complex software systems (Booch et al., 2007; Kaindl, 1999). The fundamental building blocks of OO analysis and design are class, object, and relationship. A drawback to OO models is that in order to do OO analysis and design, the users need to be rigorously trained in how to use object orientation to cope with different kinds of systems (Booch et al., 2005). The Unified Modeling Language (UML) is the OO method predominantly employed in RE (Bera & Evermann, 2014). UML is a modeling language that attempts to bring together OO approaches and that has been widely accepted (OMG, 2006). UML notation is made up of several models, which together explain the system requirements. Each model serves different purposes for distinct phases. Not all the diagrams within UML are used as, frequently, a small subset of diagrams is enough to model a system. Class diagrams and use case diagrams are often employed, as is summarized in Table 3.

TABLE 3 Summary of two relevant UML diagrams in OO model.

Framework	Description
Class diagram	Class diagrams are the basic notation of UML and are often used to record the object classes pertaining to the application domain. Basically, class diagrams express information about real-world objects in a particular problem space. Class diagrams consist of a set of classes of objects and relationships amongst classes. Akin to entity-relationship diagrams in many views, class diagrams assist how objects of a specific class relate to other objects of the same or different classes.

Continued

TABLE 3 (Continued).

Use case diagram	Use case diagrams are an RE technique that have been implemented in the objectory method (Jacobson et al., 1999) for the first time. A use case diagram contains information about the actors and use cases and about the relationship between the two. Actors can be human, agent, position, or other parties that are representative of roles. Each use case defines functional requirements for the system. Use case combines main scenarios with the related alternative and exceptional scenarios (Cockburn, 2001). The scenario functions as a design to initiate the requirements capture process (Seyff et al., 2009). It also provides detail-rich events and examples with existing memory that fosters an easy understanding of the requirements problems (Carrizo et al., 2014; Sutcliffe, 2003; Sutcliffe et al., 1998). Scenarios themselves have been proposed as a method to be employed during early activities of RE to produce models that are more familiar to users.

2.4.3 Viewpoint-Oriented Model

Viewpoint-oriented models aim to construct courses of different perspectives to develop a complete and reliable understanding of the information for the resulting system (Kotonya, 1999; Sommerville & Sawyer, 1997). Different methods based on viewpoints are the Viewpoint-Oriented Requirements Definition model (VORD), Structured Analysis and Design Techniques (SADT), and Control Requirements Expression (CORE). These types of methods are especially applicable for projects in which the system entities contain a large number of details that intricately relate to each other. They are also helpful in organizing and prioritizing requirements. One of the downfalls of viewpoint-oriented models is that they are absent of features that enable non-functional requirements to be represented straightforwardly (Pozgaj, 2000). A key character of VORD, however, is the way it structures the requirements to represent the perspectives of different stakeholders to discover points of conflict (Regnell et al., 2000). There are two genres of viewpoints in VORD (Sommerville, 2011; Sommerville & Sawyer, 1997):

- Direct viewpoints represent people or other parties that interact directly with the system.
- Indirect viewpoints represent those who do not directly use the system themselves.

Diagram notation is used to represent VORD. A viewpoint is represented using a rectangle. Direct viewpoint are unfilled rectangles and indirect viewpoint are in greyscale. Each viewpoint may contain an identifier, label/attributes, and type.

2.5 Summary

Globalization has made traditional RE techniques even more challenging, as it takes place on a massive scale, requiring adjustments to be made to accommodate openness and diversity (Boehm, 2006; Cheng & Atlee, 2007; Li et al., 2014). The RE process is composed of the interwoven sequences of communicative, collaborative, and managerial activities with the overall goal of discovering, specifying, and understanding a software system (Ferreira & Silva, 2009; Svahnberg et al., 2013). RE methods, therefore, should foster communication that distributes collaboration in synchronizing activities (Pohl & Rupp, 2011).

Many researchers have performed studies that engage modeling languages that can be easily understood by the various stakeholders in the activities. Figl et al. (2010) built and evaluated the visual instructional design language for the instructional domain. Ballejos & Montagna (2011) constructed an integrated model for representing stakeholders in information system (IS) design with all related attributes. Vassev & Hinchey (2013) proposed an approach to autonomy RE in which a goal-oriented model is merged with special generic autonomy requirements. Li et al. (2014) created a tree-overlay-based visual notation to support enterprise service modeling and generation using a more end-user-friendly metaphor. Huang et al. (2014) tried to improve graphs considering aesthetic importance in graph design and found that larger crossing angles make graph notations easier to read. Similar to these and other approaches, including Carrillo de Gea et al. (2012), the motivation of the present research came from an attempt to derive an approach to enrich communication processes among RE stakeholders. Notably, this study uses icons to communicate about the content of the requirements.

Recent studies have also pursued the development of visualized tools that enable end-users without technical experience to transmit their requirements. Aghaee & Pautasso (2014) launched a visual mash-up tool that enables non-professional users without any knowledge of programming languages and skills to create feature-rich, interactive, and useful mash-ups. Ardito et al. (2014) invented a flexible composition platform that supports the activities of the various stakeholders involved through heterogeneous visual templates. Danado & Paternò (2014) implemented a visual framework that permits end-users without information technology (IT) background to create, modify, and execute applications. Unfortunately, none of these studies employs icons. The present study is therefore intended to involve stakeholders, aiding them in the process to communicate through the capability of icons.

General-purpose modeling languages like UML and goal-oriented models have a well-established set of visual notations and constructs. While they are sufficiently expressive of model requirements, they are often inadequate in terms of interoperability between amateur and expert users (Gotel et al., 2008). Graphical complexity plays a decisive role in the usability of any visual language (Moody, 2009). In particular, novices are much more influenced by graphical complexity than experts because they must consciously retain the meaning of visual vocabularies in the memory. The graphical notations of these dominant methods can inadvertently be prejudiced by individuals' own responsibilities and backgrounds (Cheng, et al., 2001; Cheng, et al., 2014). Because of the overarching use of natural languages, UML, and goal-oriented models, several researchers have appraised the pros and cons of such methods, and the outcomes and improvement possibilities were found to be the following.

First, formal and natural language may frequently be innately ambiguous and problematic if stakeholders are unfamiliar with these notations (Falessi et al., 2013; Ferreira & Silva, 2009; Pohl & Rupp, 2011). Problem owners and domain experts might be unable to comprehend a formal specification thus cannot determine whether it perfectly represents their requirements. Requirements engineers, who are proficient with the formal specification, may not understand the solution domain, so they too cannot be sure that the formal specification is an accurate reproduction of the system requirements. One possible method for preventing the ambiguity inherent in such cases is to write requirements in a particular form using other RE visual languages (Moody, 2009; Sommerville, 2011).

Second, scenarios require a large amount of labor to make use when gathering and recording requirements. Besides, the intuition about how scenario information can be simplified into a consistent specification that can be exploited in RE has rarely been existed (Sutcliffe et al., 1998). With scenarios, it is hard to validate the invisible errors, resulting in the risk of overwhelming a specification (Boehm, 2006; Van Lamsweerde, 2000).

Third, UML provides limited traceability, addressing only certain aspects, such as process modeling and use case diagrams, and it is very poor at expressing the nature of interactions (Morris & Spanoudakis, 2001). It does not offer an explicit grammar and regularly allows for the rational linking of everything to everything; thus, the dependencies and the execution orders are omitted (Bendraou et al., 2010; Moody & Hillegersberg, 2009). Complexity, understanding, and transforming requirements into a computer model are other problems that have arisen due to the use of UML (Hadar et al., 2013; Helming et al., 2010; Meziane et al., 2008; Luna et al., 2011). In fact, the requirement artifacts have to be easily understandable to all participants in order to avoid future misunderstandings. UML is useful for capturing important aspects of requirements; however, it manifests the inconsistency of models across different viewpoints and the inconsistent interpretation of the models (Budgen et al., 2011). Even among experts, ill-defined syntax in UML can be a major problem.

Finally, although goal-orientation is a visual language and it seems like a simple task for practitioners to use graphs as a computational data structure, several shortcomings remain when putting it into practice (Duboc et al., 2013; Horkoff & Yu, 2013; Moody et al., 2010; Ur Rehman et al., 2010). For example, the direction of some dependency links is not explicitly denoted and the semantic transparency is not obviously provided. Moreover, stakeholders may be uncomfortable with expressing their needs in an abstractive pattern (Kaiya et al., 2005; Van Lamsweerde, 2000). Some researchers concentrate on the use of goal model visualization in innovative ways without considering a typical visual representation that is easy to recognize and that does not require a considerable investment of effort to comprehend (Hadar et al., 2013).

In summary, RE is a collaborative process that engages technical and non-technical stakeholders alike to communicate requirements. For the proper transmission of requirements information from one individual to another, a common code is needed, especially among those who have different cultural backgrounds and experiences (Pohl & Rupp, 2011). Therefore, it is desirable that the visual representation can be cognized by both groups. The nature of notation in visual language has always been susceptible to misrepresentation and misinterpretation. Until now, there has been a discrepancy between the attention paid to RE techniques and their actual usages. In the research community, these techniques purposefully sought to enrich users in coping with RE activities. However, in practice, unskillful stakeholders find it difficult to use them due to their unfamiliarity and the complexity of the languages themselves (Costagliola et al., 2004; Cheng et al., 2014; Lang & Duggan, 2001). Because of the pervasiveness of the outlined methods, the stakeholders who model requirements must become acquainted with the modeling languages and techniques used to document the requirements. Likewise, stakeholders who read model requirements must be accustomed with the modeling languages and techniques in order to understand the requirements correctly. Unfortunately, in reality, different stakeholders have varied knowledge and expertise, which is often insufficient; for example, business and technical users frequently have different competencies when collaborating in RE. This peril may be reduced by means of careful paying attention to the graphical syntax design (Morris & Spanoudakis, 2001; Moody, 2010).

It is worth seeking out new solutions that allow stakeholders from any background being able to cooperate in RE with a lower investment of effort (Gotel et al., 2008; Hadar et al., 2013). To make this happen, this study attempts to reduce the complexity of approaches and the level of technical know-how required to put them into practice. Likewise, another visualization method, icons, which are more likely to work well in practice, has been utilized to enhance comprehensibility. It is expected that the use of icons whose meaning can be perceived directly and learned easily may be foster the ease of use and understanding of RE work for stakeholders as they communicate their requirements.

3 ON ICONIC COMMUNICATION

In contrast to other writing systems, icons are often intended to communicate information in a non-verbal manner (Moody, 2009; Salman et al., 2012). Icons are symbols that possess a perceptual resemblance to the concept they represent (Chandler, 2007; Peirce, 1931). Icons often incorporate various aspects of concept they represent, using shades of meaning to speed up recognition (Nakamura & Zeng-Treitler, 2012). Recognition depends upon the representation and the intended meaning of such representation (Quispel & Maes, 2014). Peirce (1931) viewed a sign (icon) as a production of three-way interaction: the *Representamen* (i.e., the representation), the *Object* (i.e., the referenced object, concept, or idea) and the *Interpretant* (i.e., the procedure of interpretation).

As an example, Figure 5(a) contains a context (the level of difficulty), the representation (a *smiley* metaphor), the object (the level of difficulty = *easy)*, and the interpretation (the *smiley* indicates an *easy* level of difficulty). Figure 5(b) expresses the example of a traffic light, which uses the symbol *red* to implicate the prohibition of any traffic proceeding.

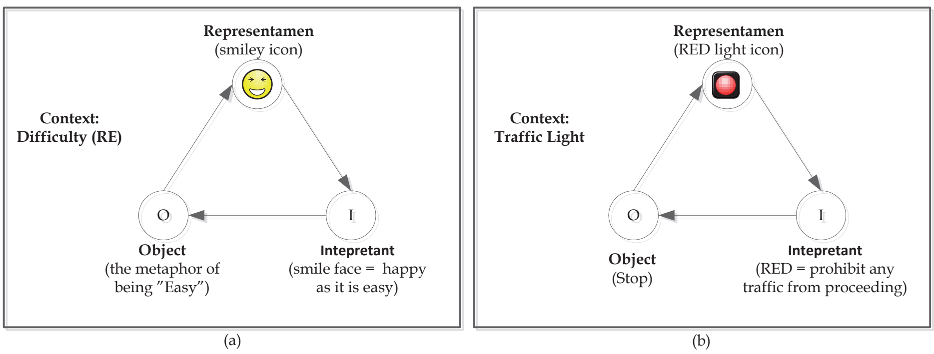

FIGURE 5 Components related to the interpretation of an icon (a) in the RE context and (b) in the traffic context.

The adoption of icons to represent specific information can promote an accurate mental model (Isherwood, 2009; McDougall et al., 2001). The relationship between the icon itself and its meaning makes visual representation self-explanatory and, therefore, easier to learn than textual language (Nakamura & Zeng-Treitler, 2012). However, when individual readers interpret icons, they do so personally; each reader understands icons based on his/her own culture, knowledge, and familiarity with the icons. This means that it is not easy for designers to determine the relationship between *Intepretant* and *Object*, as this is an intrinsic function of an individual (Heimbürger et al., 2011; Isherwood et al., 2007; McDougall & Curry, 2007).

3.1 Icon Characteristics

Icons are small images that perceptually resemble a particular function and/or object (Chandler, 2007). In a visual world, it is found that users are likely to communicate with pictorials (Lodding, 1983; Moody et al., 2010). Please note that, throughout this dissertation, the term icon includes pictures, symbols, and signs. In practice, icons are categorized into three types, based on their styles and usage (see Figure 6): concrete, abstract, and arbitrary (Lodding, 1983; Mack et al., 2002; McDougall & Curry, 2007; Microsoft, 2008).

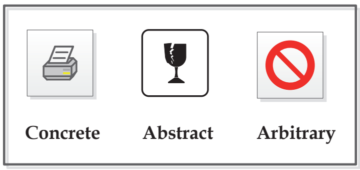

FIGURE 6 Example of concrete, abstract, and arbitrary icons.

Concrete icons are also known as representational icons, and their design generally reflects the object or action they represent. A common example is the icon of a printer seen on a computer interface. The printer icon does not display all the components of a real printer. Rather, it carries the relevant qualities: the shape of the basic structure of a printer. Concrete icons are often very effective because their design style is easy conceived, taught, learned, and retained.

Unlike concrete icons, abstract (concept-related) icons are based upon an example or concept of a real-world object or action. This icon style conveys an example of the meaning; the design is not intended to show the object itself. For example, the icon of a cracked glass intends to transmit a high-level concept, an abstraction of "*fragile*". Abstract icons tend to be more difficult to comprehend as they focus on a particular concept instead of on the direct object.

Neither concrete nor abstract, arbitrary icons were invented for a special purpose. Encompassing simple geometric shapes and colors at hand, the design of arbitrary icons seeks to foster easy learning processes. The meaning of this type of icons is the most complex and must be specifically taught, differing from other types of icons. The icon in Figure 6 is one of the most well-known traffic signs, conveying the message "*no-entry*" to its viewers. This design concept may lead to confusion if meaning is not taught, but the main idea behind arbitrary icons is that, although they often make sense in context, they must also be learned to be understood.

3.2 The Current Iconic Communication Environment

Recent developments in icons have heightened the need for the ease of communication (Britton & Jones, 1999; Bongshin et al., 2012). Communication systems based on icons originally introduced by Peirce (1931) have been reproduced in recent language work (e.g., Deacon, 1997). Icons can be found in practically any domain: hospitals, shopping centers, airports, transportation, computer systems, and many more (Aykin, 2005; Heimbürger & Kiyoki, 2010; Marcus, 2003).

Since the arrival of the Xerox Star computer in the 1970s, the use of icons in human-computer interaction (HCI) has been constantly growing. The introduction of icons to the desktop metaphor began with the onset of Apple Macintosh in the mid-1980s (Apple Computer Inc, 1996). From there, icons branched out, being extensively used in other software packages and workstation platforms (Marcus, 2003). For instance, Microsoft became a fan of icons by adopting many of them into Windows system and software packages. With the blossoming of the World Wide Web era, icons rapidly became the central ingredient of GUIs (Bongshin et al., 2012). Most web browsers use the graphical attributes of icons, such as shape and color, to convey their key functionality in a quick, comprehensible form (Galitz, 2007).

Cartographic visualization has also benefited from value-added iconic variables, which complement the existing set of text and line variables used in traditional design (Edsall, 2007). It can now be see that geography has, over time, transcended ordinary drawn text and has become somewhat iconic (Orford, 2005). Icons that are embedded in maps typically indicate points of interest or other discrete object classes (Heimbürger et al., 2014). In the new paradigm of a globalized world, geo-communication between different cultures has become more pervasive. Barriers ascending from this new situation are typically solved with a visual approach, which is often a standardized symbolic set that has to be learned by those who are involved.

Icons have also significantly benefitted crisis situations, being put into operation as an auxiliary communication amongst various users (Tatomir & Rothkrantz, 2005; Fitrianie et al., 2007; Fitrianie & Rothkrantz, 2007; Salman et al., 2012). For instance, in a disaster management system, the focus is frequently on solving the problems raised by the communications breakdowns. Being intuitive and effective, icons allow people to communicate with each other in a relatively uncomplicated way. In research conducted by Fitrianie et al. (2007) and Fitrianie and Rothkrantz (2007), the authors devised iconic communication tools to represent concepts and ideas in crisis environments. These tools are generally composed of icon strings or geometrical features, such as arrows, lines, ellipses, rectangles, and triangles. These crisis observation interfaces carry effortless respondent IS to effectively cope with natural or man-made disasters, such as explosions and fires, to avoid further catastrophes. Most of these iconic constructions are created to conform to ontology-based knowledge; within W3C web ontology language (OWL) (McGuinness & Harmelen van, 2004), for example, knowledge about a crisis context is stored in the system's ontology and is represented using graphs for data modeling. An example of icons in crisis management can be found in Figure 7. These icons reflect the situations they represent.

FIGURE 7 Example of icons used in crisis management application.

3.3 The Effects of Culture on Icon Preferences

Iconic communication systems that transcend language barriers depend on the degree of cultural background (Shen et al., 2006). Culture influences perception, which is vital to communicating via icons; therefore, this raises the question, "What we have to know about culture to understand its encouragement on user perception about icons". Cultural background is becoming even more central in the use of information and communication technology (Lindberg et al., 2006). It has long been understood that cultural diversity can prevent the successful use of IT (Aykin, 2005). The primary consensus within the existing literature about cultural differences (Hofstede, 1997; Callahan, 2006; Leidner & Kayworth, 2006; Reinecke, & Bernstein, 2011; Reinecke, & Bernstein, 2013) is that meaning is constructed through a social contract and that different cultures have different preferences. Cultures are divided according to their way of communicating. Whereas some cultures express much information implicitly, others might transmit nearly everything explicitly. Communication problems and conflicts are becoming more severe as people from diverse cultures are increasingly sharing knowledge with each other (Heimbürger et al., 2011).

The main traits of a culture's preference are distinguished based on cultural theories (e.g. Ackerman, 2002; Aykin, 2005; Heimgärtner & Holzinger, 2005; Hofstede, 1997). Hofstede has classified five dimensions on national level: power distance (PDI), individualism (IDV), masculinity (MAS), uncertainty avoidance (UAI), and long-term orientation (LTO). PDI describes the extent to which hierarchies exist and are accepted by the members in a society. Within countries that have been assigned a high PDI score (e.g., Thailand), inequalities are believed to be much more acceptable in society than in low-PDI countries (e.g., Finland). The people in highly individualist (IDV) countries (e.g., Finland) are usually seen as more independent, while people in collectivist countries (e.g., Thailand) often see themselves as a part of a group. The third dimension, MAS, refers to the high preference for competitiveness (high masculinity) versus low preference for competitiveness (femininity). The degree to which the members of society tolerate uncertainty and ambiguity is inversely reflected by UAI, that is, people from high UAI countries prefer less ambiguity than those in low UAI countries. The fifth dimension, long-term orientation, measures how people perceive time. In LTO countries, people are comfortable with sacrificing for long-term benefit, but in countries with short-term orientation, people are more focused on immediate results. In Table 4, the rules for user preferences that will be used in designing or selecting icons have been summarized. For example, if a user has a high score in the dimension of UAI, then provide a dual-code pattern, giving an icon with a caption around it.

TABLE 4 User preference aspects of icons based on cultural dimensions.

Cultural Aspect	**Design Aspect**	**Low**	**High**
IDV	Color	Colorful icons, referring to the color palette in Reinecke and Bernstein (2013).	Monotone icons, referring to the color palette in Reinecke and Bernstein (2013).
PDI	Spatial/ Orientation	Less structured data; space, between each element can be narrow, with each element placed close to the others.	Restricted structure; clear (wide) space between elements.
UAI	Support/ Guidance	Individual icons and self-explanatory.	Icon displays with dual-code (text and icon).
LTO	Shape/ Texture	Low icon detail and limited number of shapes.	High icon detail and many kinds of shapes.
MAS	Vividness of colors	Pastel colors, referring to the color palette in Reinecke and Bernstein (2013).	Brightly colored icons, referring to the color palette in Reinecke and Bernstein (2013).

3.4 Summary

Natural and textual languages are complicated on a number of levels. Linguistic communication requires participants to learn and perform complicated skills such as forming and analyzing speech (Deacon, 1997). In addition, thousands of words, complex grammatical rules, and syntax operations must be remembered. The visual languages that do not use words, but rely on symbols or icons, are therefore vital, bearing in mind that (1) different people interpret and understand natural language differently and (2) not all of the world's population is able to read or understand one specific language, such as English. As presented in Section 3.2, a large range of fields are now employing icons as a means of communication. Despite this, little attention has been paid to leveraging the potential of icons in the RE field.

In the RE domain, the main challenge faced when seeking to integrate icons has been determining the underlying visual notation that should be used to represent a specific concept to minimize ambiguity (e.g., Ballejos & Montagna, 2011; Bera & Evermann, 2014; Hadar et al., 2013; Horkoff & Yu, 2013; Li et al., 2014; Svahnber et al., 2013; Yu, 1997). Flowcharts, the first pictorial notation, have long been developed to support software engineering and are the predecessor for all modern visual notations such as UML and goal-oriented models (Agarwal et al., 2010). Visual notations are able to be employed in all stages of RE from RD to RM. They are supposed to transmit information in a clearer and more straightforward fashion to all stakeholders (Lodding, 1983; Moody, 2009). It seems an enigma, then, that with the use of these visual notations, intuitive languages that do not rely on the technical knowledge of users have not yet been developed (Boehm, 2006; Moody et al., 2010; Morris & Spanoudakis, 2001).

Accordingly, to facilitate RE communication, this research uses icons to enrich RE work and make notations more effective (Figl et al., 2010; Petre, 1995), especially for often-misunderstood aspects of RE such as attributes and relationships. Attributes highly generalize requirement characteristics and require further management activities. The prime ambition of relationships is to continue tracing all elicited requirements artifacts with other relevant attributes as well as their evolution. To envision a sound visual representation for this purpose intuitive, this study bases its principles on Deacon (1997) who has claimed that a tiny vocabulary of mnemonic icons and a modest syntax will suffice to achieve this goal. Syntax simplicity, in this sense, implies the use of a restricted number of symbols but underlining on combining such restricted visual symbols for new meanings.

Although this study initially presents icons only to represent attributes and relationships, there are other possibilities for using icons to further generalize RE, and these are shown in Table 5. Although icons offer many advantages, there are also a number of concerns relating to their application to RE. For example, individuals can interpret icons uniquely, lying on his/her culture, knowledge, and familiarity with the icons. Therefore, the meaning of an icon is theoretically based on the arbitrary knowledge and biases of the person who makes the interpretation (Heimbürger et al., 2011; Isherwood et al., 2007; McDougall & Curry, 2007). In addition, making use of icons can also be difficult regarding the extent degree to which particular icons can be relied upon to carry a specific message. Inappropriate icons not only preclude communicating the intentional message, but instead convey different messages that result in misunderstanding (Lodding, 1983; King, 2000).

TABLE 5 Opportunity to use icons to improve RE visual notations.

Possibilities for improving RE	Opportunities for using icons
Requirements must be communicated. For the transmission of information from one individual to another to work properly, a common code is needed, especially among those who have different cultural backgrounds and experiences (Nakamura & Zeng-Treitler, 2012; Pohl & Rupp, 2011).	Icons are not just images, but they are also pictorials utilized by computer graphic designers to help improve man-man and man-machine interaction (Lodding, 1983).
Notations with a meaning that can be conceived directly and learned easily can enrich the cognitive effectiveness of RE visual notation (Moody, 2009; Moody et al., 2010).	The adoption of icons to represent information for specific matters can promote accurate mental models that are likely to allow the readers to recognize an icon's meaning or function quickly and easily (Isherwood, 2009; McDougall et al., 2001).
The convergence of RE models and other visualization methods may enhance comprehensibility (Horkoff & Yu, 2013).	Icons' capabilities can be capitalized to degrade the learning curve in both time and effort for novice audiences and to expedite user performance while lessening errors (Isherwood et al., 2007). Icons are able to communicate large amounts of information succinctly and have the potential to reduce the severity international communication barriers (McDougall & Curry, 2007).

4 DESIGN SCIENCE RESEARCH APPROACH

This section describes the research method used to study the research questions (RQ) as follows.

- RQ1: What are the most common challenges in performing RE work based on existing literature?
- RQ2: How can RE work be supported by iconic communication?
- RQ3: How specific tasks of RE could be supported by icons?
- RQ4: How could multi-culturality be supported in icon design for RE?
- RQ5: How to evaluate a proposed solution to confirm if it is usable for multifaceted stakeholders?

This study relies heavily on design science research (DSR) (Hevner et al., 2004; Peffers et al., 2007) as it is at the confluence of building and evaluating processes in the construction of new artifacts (see Figure 8).

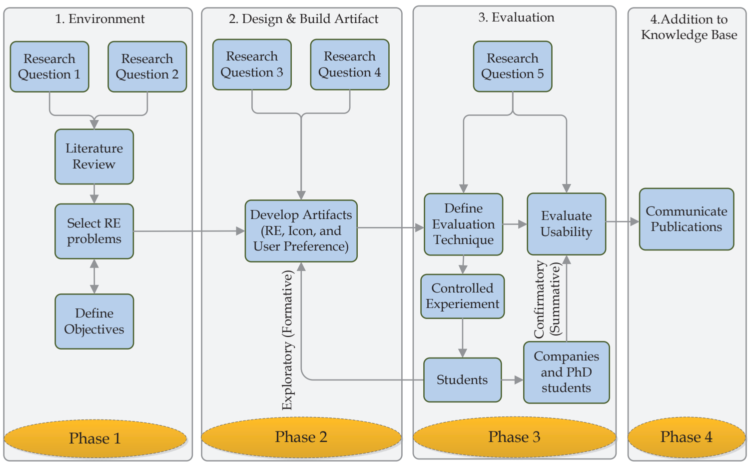

FIGURE 8 Design science research approach.

4.1 Phase 1: Understanding the Environment

Regarding the field study of the environment mentioned in Section 2.3, RE challenges can be classified into three problem spaces (addressing RQ1): requirements identification, requirements specification, and managing requirements. 1) Requirements identification mainly relates to the degree of shared understanding about requirements between business users and technical experts. To the extent that their backgrounds differ, users and developers are viewing requirements from different perspectives, drawing different conclusions about meaning. 2) Requirements specification relates to the difficulty of understanding, and reviewing requirements. Even though agreed requirements are perhaps presented, the notion of understanding is an approximation. 3) Managing requirements relates to the capability of stakeholders to manage and control requirements throughout the SDLC. As the level of complexity of products increases, RE methods that encourage management from the initial phase of elicitation to the end phase of deployment are keenly desired.

The primary objectives of a solution must be identified to diagnose the individual problem that contributes to the overall strength of a proposed solution. In this dissertation, the focus is on three insightful objectives (addressing RQ2). 1) It attempts to determine a means of communication that is tailored to stakeholders who have been influenced by different backgrounds. 2) It aims to accelerate communication to alleviate the efforts required to discover and analyze requirements. Using icons that allow for the flexible composition of various attributes and rationales may increase the number of model elements to enrich stakeholders in tackling complex information. 3) It is an intention to allow RE stakeholders to manage and continue tracking requirements. By attaching icons to each attribute and relationship, stakeholders are given the ability to visually communicate and review requirement information.

4.2 Phase 2: Design and Build Artifact

Specific elements have been defined and shown in Figure 9. Trawling for requirements begins with the elicitation activity. The business requirements (BRs) are first elicited to determine all sources of requirements that can further be used to understand the problems to be solved and the characteristics of the solution. The process of elicitation itself is habitually iterative, which means requirements can be repetitiously captured and refined to make clear understanding of the problems to be solved. Once business requirements have been gathered, users and analysts can then generate user requirements (URs), which are commonly presented in terms of a scenario to portray a set of interactions between parties, typically between actors and a system. Scenarios normally describe or exemplify a concrete example of satisfying or failing to satisfy a requirement; thereby, they contribute more detail about one or several requirements. Then relevant teams categorize and organize BRs into detailed subsets of product requirements (PRs): they explore each requirement in relation to others, examine requirements for consistency, and rank requirements based on the needs.

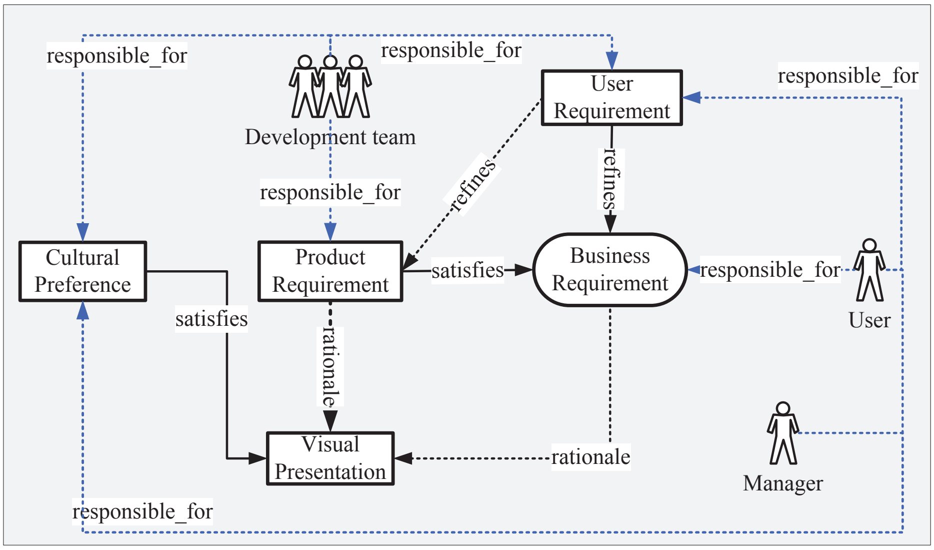

FIGURE 9 High-level components of a proposed solution.

Icon visualization is then created as a consequence of RE tasks by taking into consideration all the relevant artifacts: BRs, URs and PRs. As a result, stakeholders can review and ensure that every entailed requirement has consistently been stated. Icons are presented adaptively to the user cultural preferences that have been created as the rule-based pattern. To participate in this adaptive process, all stakeholders are required to initially provide information about their own backgrounds.

Three key artifacts, RE modeling, icon-based modeling, and user preference modeling (UFM), are needed to produce a supportive solution to reflect all the components in Figure 9 (addressing RQ3 and RQ4). The necessary requirements for every artifact are inferred from the theoretical foundations in existing knowledge bases of RE, iconic communication, and cultural ontology. Knowledge and techniques from the research field of modeling language can further be used to make design decisions that shape the concentration of the research approach. Modeling is used to help visualize the aspects of the developed artifacts. Modeling provides a way of generating the aspects' representations, through e.g., class diagram, that serve as a medium for understanding and communicating the thought or concept. A number of loops to refine and validate the artifacts are typically required during the process of designing and constructing, before the final design artifacts are engendered.

4.2.1 Requirements Engineering Modeling (Artifact 1)

All necessary textual attributes that help in understanding and managing the requirements are pre-defined in the requirements attribute schema (RAS). Figure 10 shows the overall format of RAS and its relevant constituents. The RAS includes entities such as *Requirement*, *Stakeholder*, *Source,* and *Relationship*. This information is used to model the general characteristics of requirements artifacts.

In principle, there are several genres of attributes and classifications; however, this study follows ISO/IEC/IEEE 29148 (2011), Pohl (2010), and Wiegers & Beatty (2013) concentrating on only important aspects of requirement characteristics. More than giving information about requirement, the RAS also contains requirement management attributes, such as a unique identifier, a name, a priority, a status, an origin, and a difficulty of each requirement. The class *Stakeholder* generally describes the source of requirements. Each requirement is originated by an actor, which is either a person in the *Stakeholder* class or other sources in the *Source* class; thus, it is essential to record data about the originator.

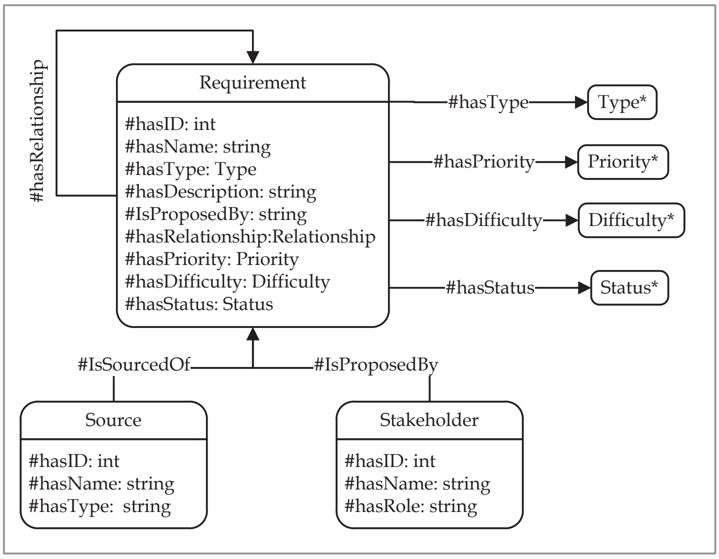

FIGURE 10 Requirements attribute schema (RAS).

4.2.1.1 Requirements Attribute

An attribute is an inherent possession of an entity that can be discriminated quantitatively and qualitatively by human or automatic means (ISO/IEC/IEEE 29148, 2011). Following is a list of the various attributes considered and a description of each.

- Priority: due to resource constraints and restrictions (time, technology, human), not all requirements can be contemplated during the development life cycle. The importance and urgency (priority) assigned to every individual requirement helps moderate imprecise requirements and assists the development team in identifying the core requirements. A requested requirement must be assigned one out of three priority levels. A "high" priority is both important and urgent, a "medium" priority is important but not urgent, and a "low" priority is neither important nor urgent.

- Status: the classification of several statuses is more meaningful for stakeholders in monitoring the progress of each single requirement throughout development process: "propose" is a status indicating when a requirement has been initialed by an authorized source; "accept" is a status indicating when a the requirement has been analyzed and key stakeholders have agreed to incorporate the requirement; "reject" is a status indicating when a requirement has been proposed but it is not planned for implementation; "implement" is a status indicating when a designing, writing, and testing of the source code that implements a requirement is underway, and "verify is a status indicating when it has been ascertained that a requirement has been implemented and is functioning properly.
- Difficulty: assuming the difficulty of each requirement (easy/nominal/difficult) offers additional context in terms of requirement's breadth and affordability. "Easy" is assigned when proposer assumes that a requesting requirement is easy to implement. "Nominal" is assigned when proposer assumes that a requesting requirement is not easy to implement but is not difficult either. "Difficult" is assigned when proposer assumes that a requesting requirement is complicated to implement.
- Origin: according to this meta-model, each requirement is originated by either a person's role or a document, and hence, it is essential to record data about the originator.
- Relationship: the existence of a relationship between two or more requirements artifacts presents and requires traceability information. The requirements can be associated with each other through the link types (see Figure 9). First, "responsible_for" is a relationship type used to document that a stakeholder (actor, role, agent, or position) is responsible for delivering the associated artifacts. Second, "refines" is a relationship type used to document that one artifact clarifies the other artifact in greater detail. Third, "satisfies" is a relationship type used to correlate one artifact (*depender*) to another (*dependee*); unless the *depender* artifact is satisfied, the *dependee* artifact cannot be satisfied. Fourth, "conflicts" is a relationship type used to explain two artifacts that foster each other in realization, yet removing one of them does not preclude the realization of the other. Finally, "rationale" is a relationship type in which one artifact aids in the construction of another artifact.
- Requirements Type: see Section 4.2.1.2 for details.

4.2.1.2 Requirements Type

A requirement's type is defined based on the properties it represents. This necessitates collecting requirements into classifications for analysis and management. The main types of requirements are received from literature and RE standards (e.g., ISO/IEC 9126, 1996; ISO/IEC/IEEE 29148, 2011; Wiegers & Beatty, 2013; Sommerville, 2011). In this dissertation, requirements have been divided into three chief classes.

Business Level

- Business requirements represent high-level or abstract objectives of the organization, so are defined in this study as goals. Business requirements are the reason for developing software systems in the first place.

User Level

- User requirements detail business requirements or inter-related tasks that the users must be able to execute using the product. Proven ways to determine the user requirements are use cases or scenarios.
- Business rules contain organization policies, laws, industry standards, and accounting practices that limit certain freedoms in delivering a solution.

Product Level

- Functional requirements are statements of services the system should provide, how the system should react to particular inputs, and how the system should behave in particular situations.
- Quality requirements define the quality properties that must be matched by the entire system or by a system component, service, or function. There are many types of quality requirements. Pohl recommended not using the term "non-functional requirements" because quite often, non-functional requirements document quality requirements in details and thereby help to refine underspecified functional requirements (Pohl, 2010).
- Interface requirements designate how the system interacts with external systems and how internal system components interact with each other.
- Constraints also important to consider because they can limit the options open to the development teams in constructing the software product.

4.2.2 Icon-Based Modeling (Artifact 2)

Icons have been brought to represent a piece of RE artifacts. The capabilities of icons may mitigate the learning curve of amateurs with regard to both time and effort (Isherwood et al., 2007). Icon surfaces are formalized to the manner and context they represent. Two contexts, attributes and link types, must first be sharpened by icons corresponding to the RAS. As portrayed in Figure 11, an icon library has been prepared by maintaining each variable to comply with the rule-based principles that take five cultural aspects into consideration. The rule-based principles can be found in Table 4.

Other sources such as cognitive principles (e.g., Ainsworth, 2006; Moody, 2009) and design guidelines (e.g., ETSI EG 202-048, 2002; ISO/IEC 11581, 2002) can also be utilized as design principles for creating effective visual syntax. A main element is the class *Icon*, which defines general characteristics such as colorfulness, vividness of color, orientation, coding, and density. To gain the characteristics defined in the model that are suitable for a person's background, icon elements are connected to the class *CulturalValue*. This class stores the score of the cultural dimensions in five corresponding sub properties. *CulturalValue* could also be extended to include further aspects of a person's cultural background, such as education, language, or experience.

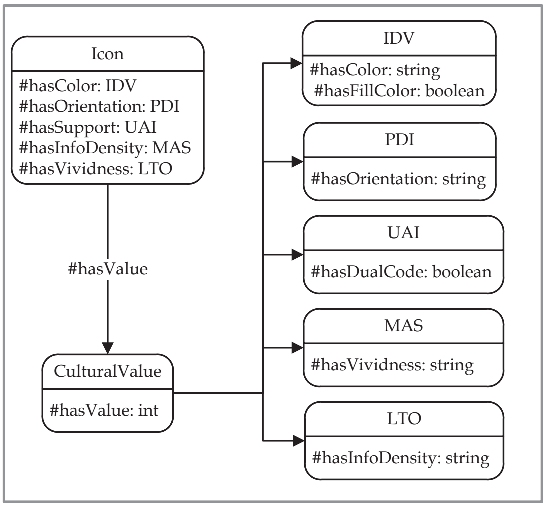

FIGURE 11 Icon-based information modeling.

This dissertation communicates the framework for icons' construction. In fact, all icons used in this study are gathered from existing sources and resemble simple concepts and ideas. This is because the design perspective must be undertaken by designers who are experts in the area. When choosing icons to refer to the RE artifact, concrete and abstract icons are given first priority. However, certain artifacts cannot be portrayed by either concrete or abstract icons, and in such cases, arbitrary icons were used.

Arbitrary icons are structured through combining concepts to yield a composite meaning. Once concepts are combined in this manner, rules of syntax govern how and where they may be compounded. This study consequently carries out the simple principles of building an icon grammar as follows:

- A noun form—concrete or abstract— is used to resemble a subject and object.
- A verb form—e.g., an arrow or a mathematic sign— is used to resemble an action or a modifier.

- An adjective form—orientation— is used to resemble information property (e.g., if one symbol is touching another, it means the two relate to each other).

4.2.3 User Preference Modeling (Artifact 3)

The UFM, in accordance with RQ4, acquires background information about each individual stakeholder, as displayed in Figure 12. This information is used to discern user preference. A *Stakeholder* is characterized by name, age, language of speaking, education level, and gender. Two different types of gender, female and male, can be defined. To demonstrate the effect of nationality upon preference, the framework also includes *Nationality,* which is the main derivation of five national dimensions: MAS, UAI, PDI, IDV, and LTO. Each index aids in arranging the icon pattern in the second artifact. The user preference framework has also been accompanied with the class *RE Experience*, which includes role, year, and month, which afford information pertaining to a user's competency. It is worth noting that the UFM draws upon Reinecke and Bernstein's (2013) user cultural ontology theory because its concept has proved useful in adapting preferences to suit personal background.

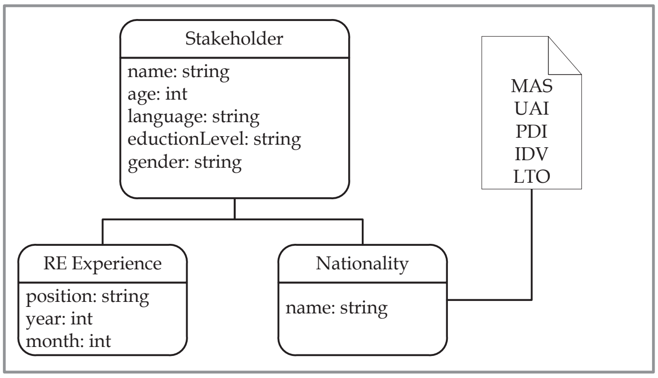

FIGURE 12 User preference modeling.

4.2.4 Integrating the Three Artifacts

This section describes the integration of the three artifacts, icons that symbolize RE artifacts, and the icon concept that bases on cultural aspects. Icon-based modeling is associated to the RE modeling in a number of ways, such as the way in which icons are assigned based on how they have been defined in the RAS; attributes and relationships are two examples used in this study. Icons' appearance can be designed and adapted according to the cultural patterns. While icon-based modeling is as dynamic as cultural preference, RE modeling is static and is not affected by UFM. Figure 13(a) describes how icons, requirements artifact, and cultural aspects can be intertwined. Attributes and link types in RE modeling can be embedded by icons or symbols.

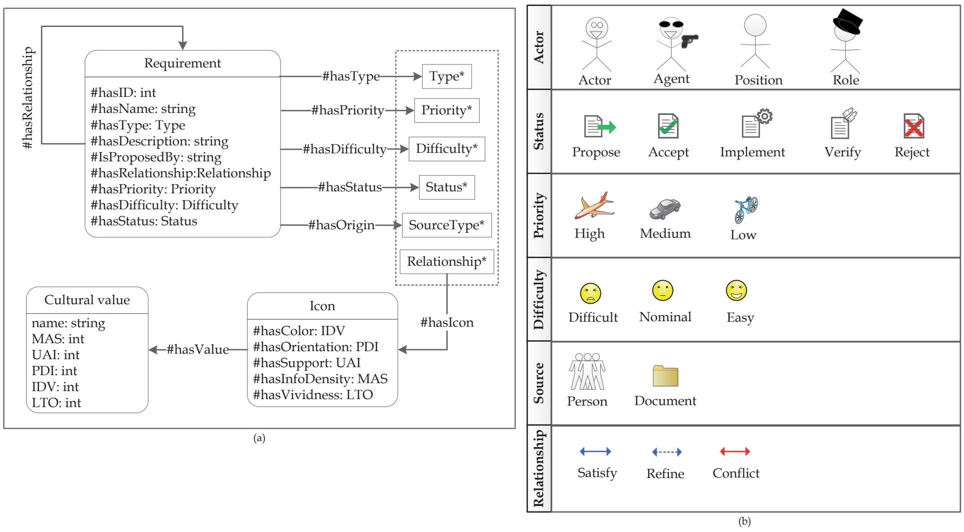

FIGURE 13 (a) Meta-class of icon, RE, and cultural artifacts and (b) examples of icon notations of attributes and relationships for Finnish culture.

Depending on the rule-based principles, outlined in Table 4, an icon library containing iconic symbols is produced. Icons are either abstraction, concreteness or arbitrary that their displayed appearances are varied from nationality to nationality. Figure 13(b) gives an example of icon design characteristics for the Finnish nationality, in which IDV = high, MAS = low, PDI = low, UAI = high, and LTO = low.

- Because of a high IDV, color preference color tends to be monotone (see color palette in Reinecke & Bernstein (2013)).
- Due to a low MAS, the vividness of color tends to be bright (see color palette in Reinecke & Bernstein (2013)).
- Owing to a low PDI, objects can be positioned close to/overlapping each other (see icons for status kinds).
- As a consequence of a high UAI, icons are encoded with caption, namely, they are dual-code.
- In conjunction with a low LTO, icons are presented in low density, unlike real, detailed pictures.

Note that some visual vocabularies used in this study were gathered from existing sources but were only used to represent concepts and ideas. At this stage, it is not a concern how their appearance relates to what they represent. Rather, the reference notions of icons are extracted from their metaphor.

Different types of actors can be distinguished using stick figures with dissimilar variations. This view of using stick figures to simplify mnemonic effectiveness has also been studied (Moody, 2009) to make languages more discriminability.

The velocity metaphor of vehicles is exploited in complying with three priority levels: the fastest vehicle, a plane, represents "high," the moderately fast vehicle, a car, represents "medium," and the slowest vehicle, a bicycle, represents "low."

It seems plausible to portray levels of difficulty using smiley face representation. For example, being smile face obviously conveys happiness, therefore describing the requirements as "easy."

Source attributes use object metaphors: a pictorial person icon illustrates that a requirement is proposed by a stakeholder, while a folder icon illustrates that a requirement is rooted in a standard, policy, or specification.

Link types are represented using geometric metaphors such as lines, arrows, and shapes, which are internationally accepted (Aykin, 2005; Huang et al., 2014).

For status kinds, arbitrary icons are taken to play, such as a piece of paper representing a requirement, arrows and signs representing an action. These signifiers do not resemble the objects they depict; however, it is not difficult to interpret their semantic meaning based on surrounding elements.

4.3 Phase 3: Demonstrator and Evaluation

This section (which addresses RQ5), first discusses the demonstrator, which utilized the MediaWiki platform, together with empirical evaluation settings and their target audiences.

4.3.1 Demonstrator through MediaWiki

To illustrate and evaluate the proposed solution, three incremental web applications were developed. These three versions permit participants to access and manage their test task and questionnaire online. Demonstrators are developed based on the MediaWiki platform (Decker et al., 2007; Majchrzak et al., 2013). MediaWiki was selected because it provides a flexible and open source platform for collaboration to create application content. Also, MediaWiki is applicable in other development phases, such as testing. MediaWiki is a collaborative tool that can be used by developers without the need for infrastructure integration.

4.3.1.1 On Top of the Basic MediaWiki

The basic components of MediaWiki are illustrated in Figure 14(a); these components are basically built around the following layers.

- Data layer: consists of a host file system and object-relational database, that is, MySQL.
- Logic Layer: formed by MediaWiki's PHP script and the PHP interpretation engine.

- Network Layer: provides a network-level interface between clients and the server. The MediaWiki network layer is built upon Apache web server.

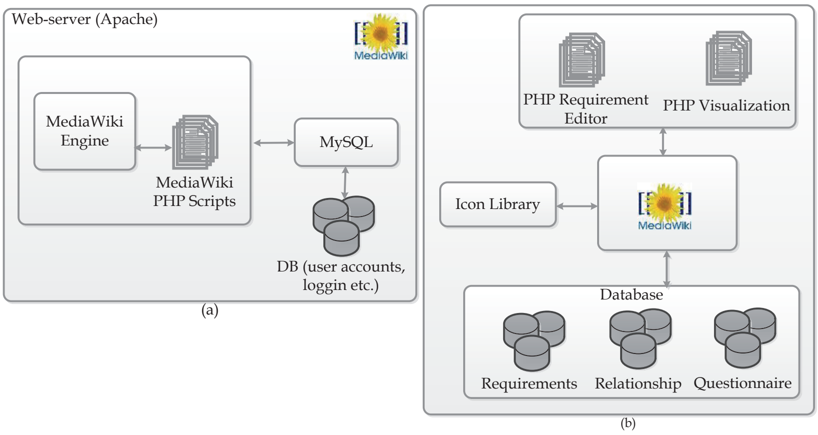

FIGURE 14 (a) Basic MediaWiki components and (b) incremental MedialWiki for icon-based demonstrator.

The basic model does provide for collaboration and distribution, but not to the extent required in RE, especially requirements creation, requirements modification, and requirements visualization. Furthermore, the basic model does not have details of attributes and properties that should be stored within the model. Therefore, those features have been customized to intensify the demonstrator (see Figure 14(b)). On top of the standard architecture, new PHP scripts for requirements editors and for requirements visualization had to be established. Three database tables were constituted to store requirements, relationships between requirements, and replies to questionnaires. Additionally, an icon library was made in which to keep all relevant icons that will be the representative of requirements attributes and relationships. The requirement editor allows users to transfer their needs into requirements forms, whereas the visualization form transmits requirements submitted by users into iconic form.

4.3.1.2 Demonstrator Interfaces

The demonstrator in Figure 15 is the requirements management, which assists users to organize their desires into requirements and to transform those requirements into iconic representations.

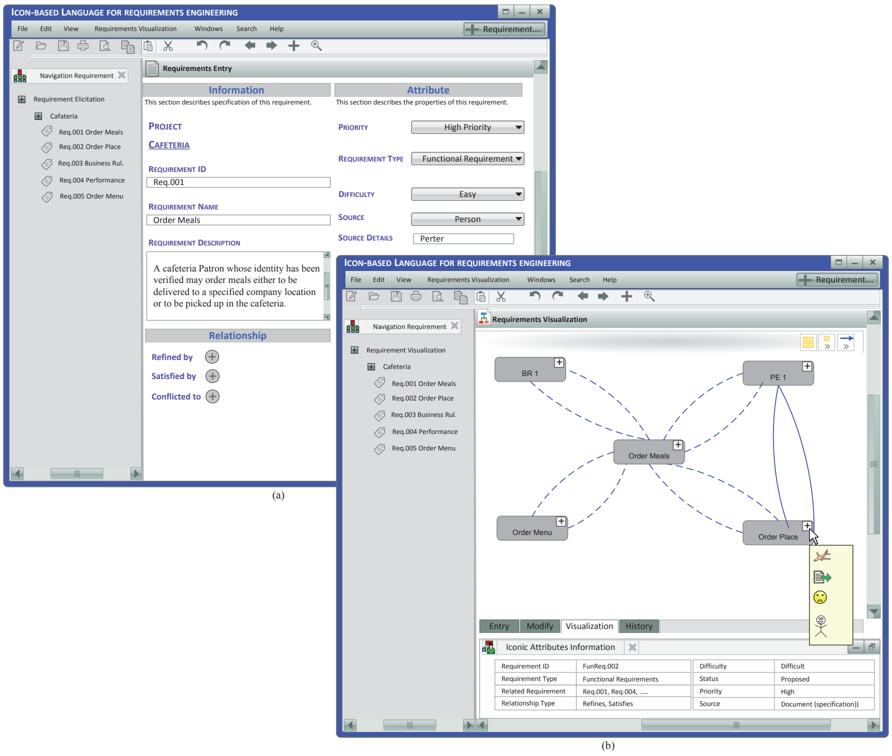

FIGURE 15 Examples of (a) requirements entry screen and (b) the transition of textual requirements into icon visualization.

In this study, demonstrators build on previous research, which is the specification of cafeteria ordering system (Wiegers & Beatty, 2013). Rudimentary for such specification, the RAS is elaborated to contain related attributes and relationships as a requirements editor template, as shown in Figure 15(a). As specified in the delivered RAS, users are first able to enter a new requirement that all materials are restricted to such a schema template. In this stage, users can comprehend how their needs can be formulated into textual requirements. Screens for updating, reading, and deleting the requirements are also provided. Afterward, practitioners can query to display textual requirements in the form of icon visualization (see Figure 15(b)). All requirements are transitioned into a short form, and all attributes are embedded by icons, together with their relationships. Users can configure whether to visualize an icon individually or in conjunction with several options.

4.3.2 Empirical Iteration

In this dissertation, usability experiments are set through which practitioners can study artifacts in a controlled environment (Hevner et al., 2004; Venkateshet al., 2003). People living in Finland were chosen as the target groups. Table 6 describes empirical iterations and participants encompassed in this study.

TABLE 6 Empirical evaluation iterations and target groups.

Empirical Evaluation	Target Groups	Number
Formative	Students attending RE class at the Department of Mathematical Information Technology at the University of Jyväskylä	48
Pre-testing	Experts in software companies	2
Summative	Experts in software and computers enterprises	9
	PhD students under the Faculty of Information Technology at the University of Jyväskylä	5

4.3.2.1 Exploratory (Formative) Evaluation

The formative evaluation is to gain preliminary result on how to improve the design artifacts. Formative evaluation is performed by the 48 students in the RE course in the Faculty Information Technology at the University of Jyväskylä. These students were in the group assigned to participate in the first empirical evaluation session, as they fit the selection criteria i.e., those practitioners should be naïve, amateurs, and have multicultural characteristics.

4.3.2.2 Pre-Test Evaluation

Before distributing the demonstrator to the post-test, a pre-test is pivotal to ensure that the demonstrator is properly implemented and that it can be understood and used. This evaluation is executed by few testers who possessed considerable expertise in the RE field. The results of this evaluation were utilized to improve the next demonstrator. The series of components that seem to disrupt pre-testers and those which are changed for the final mostly relate to the look-and-feel of the demonstrator. For instance, some practitioners cannot comprehend how an icon-based approach could support their work as a whole. Because of this reason, the later version of the demonstrator is altered to contain background information about the icon-based approach so that every participant can read to understand the approach in an overview; a guidelines page about what needs to be done and how to interact with the demonstrator is also provided. Besides, the visual presentation is updated for more user-friendly.

4.3.2.3 Confirmatory (Summative) Evaluation

The summative evaluation aids in obtaining the confirmed utilities and efficacies of design artifacts. Within the summative evaluation, two sub-iterations are delineated. One phase, which is performed first, is conducted by nine experts in software and computer companies located in Finland. The condition for selecting this type of candidate is that, in all enterprises, some of their duties have to relate to RE activities. Otherwise, the participants themselves must have experience in an RE role. The second phase is later conducted by five PhD students. PhDs were selected rather than experts in software/computer organizations to weigh how highly educated people perceive the use of icons to enrich RE work. To balance their understanding about RE, the PhD students must currently be under the Faculty of Information Technology at the University of Jyväskylä.

4.4 Phase 4: Publications Related to the Research Phases

As described, this study follows DSR and consists of four phases (shown in Figure 8). This section expresses the fourth phase, that is, to communicate the results derived by this study to the knowledge base. Table 7 summarizes the communication basics of articles related to this dissertation and explains their relationships to the research phases. More than analyzing the literature, other documents such as public reports published on the Internet were used in all the studies. There were particularly used to determine what solutions have been presented earlier in answer to the questions studied.

TABLE 7 Original articles and their relationship to research phases.

Phases / Article	**Identification of Problems and Objectives**	**Design and Development**	**Empirical Evaluation**
[AI]	Theory base • Literature & state-of-the-art analysis		
[AII]	Theory base • Literature & state-of-the-art analysis		
[AIII]		Artifact-building • Literature & state-of-the-art analysis • Construction	
[AIV]	Theory base • Literature & state-of-the-art analysis		
[AV]		Artifact-building • Literature & state-of-the-art analysis • Construction	
[AVI]		Artifact-improving • Construction	Artifact-evaluating • Evaluation (students)
[AVII]			Artifact-evaluating • Evaluation (students)
[AVIII]			Artifact-evaluating • Experts • PhD students

4.5 Summary

Almost every element in our IS environment is much more artificial, therefore, a theory of design in IS can be stated as a scientific revolution (Hevner et al., 2004; Samon, 1996). One key concern that cannot be neglected in DSR is how it differentiates from routine design. While the routine design gives an attention to the application of existing knowledge or organization's problems, the DSR contributes to the solution for unachieved problems in innovative manners (Hevner et al., 2004). The information in Table 8 summarizes the conclusions of the post-evaluation. This measures how thoroughly this research makes use of DSR processes in delivering a solution (Hevner & Chatterjee, 2010).

TABLE 8 Evaluation of icon-based approach to DSR checklist.

DSR questions	Icon-based approach
What is the research question (design requirements)?	This research has explicitly identified five research questions to be studied (presented in Table 1).
What is the artifact? How is the artifact represented?	DSR produces a variable artifact in the form of a construct, a model, a method, or an instantiation. In this study, artifacts are composed of models (RE modeling, icon-based modeling and user preference modeling), demonstrators (MediaWiki), and data-gathering templates (questionnaire and interview).
What design processes (search heuristics) were used to build the artifact?	This study attempts to develop many artifacts, and each artifact can be designed by different means. (1) RE modeling is designed by performing a thorough analysis of requirements in the RE process and identifying features provided by leading RE activities. The basic routines of RE modeling define the characteristic of the requirement. (2) Icon-based modeling is designed with regard to the rudiments of icon design conventions. The findings from icon design conventions lead an understanding of *what* and *how* to produce icons that represent RE. (3) User cultural modeling is designed based on the existing ontology. This ontology provides basic elements to derive cultural models for an icon-based approach. (4) Demonstrators are designed by applying fundamentals of web-based knowledge base. (5) Evaluation templates, including questionnaires and interview, are designed by systematically following existing evaluation methods.
How are the artifact and the design processes grounded by the knowledge base?	This research draws upon literature and knowledge in modeling, web-based technology, and evaluation methods. Class-based ontology is taken into account for designing RE modeling, icon-based modeling and user preference modeling artifacts. MediaWiki, a web-based technology, is used in demonstrator implementation. Usability experiments with control variables are the main base for evaluation and data gathering.

Continued

TABLE 8 (Continued).

What evaluations are performed during the internal design cycles?	The empirical evaluation in a controlled environment permitting users to study artifacts for usability has been done in three phases. The first phase is as a formative evaluation, done by students in an RE class. The second phase is managed by experts who possess long expertise in the RE field. The results of this evaluation were utilized to make improvement to the demonstrators. The third phase occurs in an environment in which experts and PhDs completed test tasks and provided feedback via questionnaires and interviews.
How is the artifact introduced into the application environment and how is it field-tested?	Artifacts have been introduced into the application environment through the demonstrators where all materials of demonstrators are restricted to the concept of each artifact. In accordance with this, artifacts are reiteratively reviewed, refined, and reconstructed.
What new knowledge is added to the knowledge base and in what form?	Peer-reviewed articles pertaining to new proposed artifacts, proof-of-concept, proof-of-value analyses, and data gathering templates on the icon-based approach were added to the knowledge base in multiple disciplines such as RE, modeling and visual language.
Has the research question been satisfactorily addressed?	Each research question is studied in individual DSR phases. Using the results of Phase I, research gaps and objectives of the solution (RQ1 and RQ2) are realized. Through Phase II, RE tasks, icons, and cultural influence are known (RQ3 and RQ4). Through Phase III, the validation of a proposed approach is confirmed (RQ5). The satisfaction of individual RQs can be confirmed using the related articles: • RQ1 and RQ2 are discussed in [AI] and [AII] • RQ3 is discussed in [AIII], [AV] and [AVI] • RQ4 is discussed in [AIV] and [AVI] • RQ5 is discussed in [AVI], [AVII] and [AVIII]

DSR has been a notable theory in IS research, which involves the construction of a wide range of artifacts (constructs, methods, models, or instantiations). Not only does DSR design an artifact, it must also be able to provide evidence that the designed artifact solves the real problems. Throughout the process of DSR, the fundamental questions are "What utility does the new artifact provide?" and "What demonstrates that utility?" While the current section addressed the utility that new artifacts aim to provide (former question), the next section outlines the evidences of the appraised artifacts' utility (latter question).

5 SUMMARY OF INCLUDED ARTICLES

This section consists of an overview of the results of this dissertation and the original articles constituting its basis. The intent of the research into enriching RE work with icon-based language is (1) to understand the current status of RE and its use of icons, (2) to introduce RE compositions that can be represented by icons, and (3) to evaluate whether such a conceptual model is usable from users' points of view.

5.1 Understanding the Environment

5.1.1 Article [AI]: Icon-based Language in Requirements Development

This article was published in 2012: Y. Kiyoki & T. Tokuda & N. Yoshida (Eds.) 22nd European Japanese Conference on Information Modeling and Knowledge Bases. Prague, Czech Republic, June 4-9, 20.

Objective of this Study

The aim of article [AI] is to determine both the importance of RE and its current status in the industrial and research communities. In addition, the weaknesses of existing visual representations are discussed, and the adoption of icon representation is proposed. This article briefly discusses requirements development (RD) and the potential ways in which iconic communication can enrich RD. This would help practitioners and researchers understand what needs to be improved in RE practice.

Findings and Contribution

The authors of this paper researched a series of likely challenges in RE and focused on RD. RD requires significant efforts to identify stakeholders' desires, and to analyze them into system requirements. The overall success of a software project is mainly influenced by the contribution of RE. The most outstanding cause of failure is due to the changes in requirements and scope. The main reason for this is poor communication between stakeholders who vary greatly in terms of background, nationality, culture, experience, level of education, age, and gender.

Indeed, a number of diagrammatic notations exist that represent sets of requirements and their relationships. The authors of this study identified that there is one basic issue that arises when reviewing available visual techniques such as UML or use case diagrams. This issue is the complexity of depiction for both modelers and interpreters. Visual languages generally have one main advantage over non-visualized mechanisms. This advantage stems from the fact that the capabilities of visual languages make them less difficult for those who have no experience.

This study is dedicated to improving the understanding of RE and the ability of icons to enhance RD work. The contributions of this study are 1) the identification of RE challenges in practice for which solutions to be developed, improved, or introduced and 2) the identification of icons (and their capabilities) as an alternative way to facilitate RD tasks and help solve the challenges.

5.1.2 Article [AII]: Icon-based Language in the Context of Requirements Engineering

This article was published in 2013: R. B. Svensson, D. M. Berry, M. Daneva, et al. (Eds.) 19th International Working Conference on Requirements Engineering: Foundation for Software Quality (REFSQ). Essen, Germany, April 8-11, 215.

Objective of this Study

The aim of article [AII] is to introduce an icon-based approach to help illuminate the general characteristics of requirements. This article presents concept as the entire whereas article [AI] positions the proposed solution for a fragment of RD. As a portion of Phase I (in Table 1), the authors analyzed the current state of RE and the potential shortcomings of existing and well-known methods to find a potential solution. This study provided an overview of the requirements world, icons world, and their integration. The research problems, questions, and methodology are included in the study.

Findings and Contribution

This study identified five research questions (RQs). In addition, DSR was used to determine the research outcome. Starting with problem identification, three primary research problems that remain to be solved are outlined. Next, concrete objectives to obtain a possible solution are defined. This is followed by a solution design and an empirical evaluation. Based on the solution design, a meta-model is introduced. This meta-model includes all relative entities, such the actors and requirements, the relationship between the two, and their attributes.

This study contributes to the research by 1) identifying RQs, 2) pursuing a research methodology to achieve RQs, and 3) introducing designed artifacts to address the problems. The result of this study clarifies to the concept of iconic communication that goes beyond the traditional natural languages.

5.2 Designing and Building Conceptual Models

5.2.1 Article [AIII]: Icon Representations in Supporting Requirements Elicitation Process

This article was published in 2014: T. Tokuda, Y. Kiyoki, H. Jaakkola & N. Yoshida (Eds.) Information Modeling and Knowledge Bases XXV. IOS Press, Amsterdam, 133-145.

Objective of this Study

In line with article [AII], this paper explains the requirement model in great detail. This paper considers the model in terms of addressing the difficulties faced by stakeholders in the general requirements elicitation process. This paper addresses the question of how the tasks of software developers and other stakeholders in RE can be facilitated by the use of icons in interpreting and modeling.

Findings and Contribution

This study identified the general characteristics of requirements elicitation by means of a model. Taxonomy of requirements types is included in the requirements elicitation model to potentially help stakeholders understand and communicate requirements more effectively. This study also classified requirement artifacts, together with a set of relevant actors, attributes, and relationships. Two attributes (status and priority) and two link types (dependency and parent-child) were first refined to be represented by icons. This study explains the possible syntax and semantics when designing icons to incorporate. This article also suggests the evaluation pattern: a series of potential tasks and questions, a test scenario, and a target audience.

5.2.2 Article [AIV]: Cross-Cultural Communication with Icons and Images

This article was published in 2014: T. Tokuda, Y. Kiyoki, H. Jaakkola & N. Yoshida (Eds.) Information Modeling and Knowledge Bases XXV. ISO Press, Amsterdam, 306-322.

Objective of this Study

The aim of this paper is to discuss icons and images in the context of cross-cultural communication. It embraces a visual vocabulary that any one from any culture, any country, and in any context can understand. The collections of visual symbols usually are context-specific and cross-cultural.

Findings and Contribution

Article [AIV] attempts to further the understanding of professional application domains for icons. The finding of this study is that icons are a good means of communication within a certain application domain and in a certain context. Culture is an embodiment of how we interact with other individuals and how we interpret things in different situations. Therefore, the way people understand icons is affected by culture through learned meanings of phenomena, items, and actions, such as reading direction and symbolic meanings. Context also has a central role in cultural interpretation. Different contexts lead people to associate different meaning with the same icons.

5.2.3 Article [AV]: Icon-based Language: Auxiliary Communication for Requirements Engineering

This article was published in 2013: International Journal of Engineering Science and Technology 5, 1076-1082.

Objective of this Study

After the research presented in article [AIII], the practical usage and precision of prime artifacts for solution design were seen as essential in order to inform the approach objective. Article [AV] therefore provides a more detailed solution. A design solution is given for a set of artifacts. This paper constructs two artifacts that can be turned into a means of requirements communication: a definition of the requirements world and a definition of the icon world. To facilitate better organization and management, a good requirement must be well-established. Consequently, this paper initially proposes plausible requirements attributes and relationships that match the easily understood and verifiable characteristics. It is critical to define the icons as they relate to the requirements attributes

Findings and Contribution

This study contributes to the development of two key insights in the design solution stage. The contributions of this study are 1) generic characteristics in the context of requirements and 2) icon characteristics that relate to requirement attributes and relationships. This study identified the necessary characteristics of well-formed requirements that meet the stakeholders' intentions, as follows:

- Requirements attribute: individual requirements must contain essential properties that assist understanding, analyzing and managing, that is, attributes. The attribute information in this study is associated with priority and status. As appropriate, a simple scheme such as "high", "medium", and "low" can be used to identify the priority of each requirement. Each requirement includes an attribute that indicates the evolution from "propose" to "verify".
- Requirements traceability: relationships commonly link lower-level requirements to higher-level requirements and vice versa. The dependency between requirements has to be defined so that if the dependent requirement is removed or modified, the supporting requirement can also be detected.

Icons referencing these aspects are identified and presented. During this process, it is the case that metaphors are used to resemble requirements context that is impossible to represent using concrete objects. The velocity metaphor is taken for granted to represent priority; for instance the very low speed of a bicycle could help describe a low state priority. The action metaphor of compounded icons denotes status. The direction of arrows follows a particular convention to represent dependency that can be traced to requirements' relationships.

5.3 Developing and Evaluating the Construction

5.3.1 Article [AVI]: Icons: Visual representation to enrich requirements engineering work

This article was published in 2013: Journal of Software Engineering and Applications 6 (11), 610-622.

Objective of this Study

The aim of this paper is to examine and determine how people interpret icons that represent requirements attributes and relationships. In compliance with evaluation purposes, icons are divided into three series: individual icon interpretation, multiple icon interpretation and compound icon interpretation. In line with article [AIV], this paper incrementally details already devised artifacts and inserts another crucial artifact, cultural aspects, to broaden the use of a solution for different stakeholders with different backgrounds.

Findings and Contribution

Article [AVI] contributes to the empirical evaluation of icon representation in the context of requirements attributes and relationships. The contributions are 1) identifying and defining a group of testing tasks and questions that can be utilized in assessing the concepts of a solution and 2) developing demonstrator to contain testing tasks and questions that allow practitioners to perform interactively.

The findings of the empirical evaluation suggest the interpretation of each icon test as follows. First, the individual and multiple icon interpretations have a strong connection to interpreters' conventional knowledge. Second, with compound icon interpretation, it is revealed that the use of dual-code icons with label captions help interpreters to easily understand icons' meaning. Last, and especially interesting, it is shown that novices are able to interpret icons in some tasks more accurately than experienced users. Therefore, the interpretation of icons does not rely on the degree of experience.

5.3.2 Article [AVII]: Icons Recognition and Usability for Requirements Engineering

This article was published in 2014: B. Thalheim, H. Jaakkola & Y. Kiyoki (Eds.) 24th International Conference on Information Modeling and Knowledge Bases. Kiel, Germany, June 3-6, 248. A re-edit paper will be published in 2015: Information Modeling and Knowledge Bases XXVI by ISO Press, Amsterdam.

Objective of this Study

When introducing new methods, it must be taken into account that what users perceive as recognizable and usable strongly depends on their background. Therefore, evaluation to determine users' perceptions is needed. This article aims at evaluating the icon approach to explore its usability in enhancing RE work. In order to achieve this aim, the authors of this paper develop test tasks and questionnaires. These tasks were tested by 48 students in RE course at University of Jyväskylä.

Findings and Contribution

This study evaluates icons to confirm that the use of icons to represent textual attributes improves recognition and that the use of icons to represent things increases satisfaction. 1) Concerning icon recognition, experiments show promising results, which generally support the research direction. Icons proved to be easily recognized by participants when all relevant icons are presented together. In particular, dual-coding icons help the viewers to interpreted icons more accurately. However if icons are displayed individually, it appears that the readers cannot guess their meaning correctly. 2) Concerning icon usability, three measurement types were gauged. First, satisfaction based on the degree to how participants beneficially perceived icons received popularity at satisfied level. Second, attitude toward using the system obtained a higher positive feedback than negative response Nevertheless, no opinion on attitude with regard to a system gained a majority. Third, effort expectation to judge the degree of ease associated with the use of a system. Unfortunately, because this demonstrator is developed just for presenting a small fragment of a solution concept, it results in failure to serve ease of use in users' perspective. Besides, this study is not emphasized on icons design thus the portrayal icons were selected from existing icons. Otherwise they were created only for the reference purpose without comprehensive design. As a consequence this may make icon representation in this study fail in fulfilling users' expectation.

5.3.3 Article [AVIII]: Can Icons Enhance Requirements Engineering Work?

This article was submitted to Journal of Visual Languages & Computing.

The objective of this study

The aim of this article is to examine the usability and aesthetic satisfaction of the proposed approach. The results are obtained by empirical evaluation of three measurement types. First, Moody (2009) recommends that visual communication must reflect users' perceptions. Therefore, this study assesses the effectiveness of the icon-based approach by indicating whether or not it offers positive and usable means for RE stakeholder to communicate their requirements. Second, considering ergonomic quality (EQ) (Hassenzah et al., 2000), this study tests the quality aspects of the products for effectiveness and efficiency to see if users can reach task-related goals related to function or design issues. Third, regarding hedonic quality (HQ) (Hassenzah et al., 2000), this study checks the quality aspects that users desire in the novel proposal associated with the items of pleasure. This paper also attempts to compare the reaction to the icon approach of different groups of practitioners.

Findings and Contribution

This article goes on to explore the applicability of the icon-based approach from the perspective of a number of practitioners. For the first measurement of usability, the study revealed that the artifacts proposed in this research make the method to enrich RE work more interesting. For the second measurement of aesthetics satisfaction the icon-based approach, the study revealed that requirement artifacts represented by icons can be easily understood by stakeholders of different backgrounds when they communicate requirements. If a set of representative icons are sufficiently appropriate and everybody interprets them the same way, this approach may be a fast and easy method that minimizes misunderstanding. Regarding the overall preference of icon metaphors, it appears that metaphors can offer a positive choice when designing or selecting icons to refer to some contexts. However, it might not be possible to find a metaphor that corresponds to every context. Last but not least, this study revealed that different people interpret icons differently. Even though those people have the same nationality and perceive the same pattern of icons, they might interpret things differently. The final and most important discovery in this study is that experienced and inexperienced stakeholders do not have vastly differing opinions about the icon-based approach.

6 CONCLUSION, LIMITATIONS, AND FURTHER STUDY

This section concludes the dissertation by summarizing the results and limitations and outlining future research needs. The future research section suggests future research opportunities to complement and continue the work carried out in this dissertation.

6.1 Conclusion

Various methods and techniques such as natural languages and semi-formal or informal approaches are presently utilized in RE. Visualizations are seen as essential for communicating requirements between domain and technical experts. The visual techniques in RE currently come in many different forms, such as charts, graphs, and diagrams. This thesis is concerned with creating icon information that can facilitate RE work. This will help various project stakeholders to deal with requirements. For instance, users would be able to review their proposed requirement by looking up its status progress. This can be beneficial for requirements engineers that they can investigate the association of all the specified requirements through link types. Project managers may be interested in this approach because it allows them to monitor requirements based on their priority and difficulty.

This research draws upon DSR method, which comprises four phases to derive an icon-based approach. The first phase is field study to understand environments and problems. The second phase involves developing solution artifacts to tackle the problems identified in the first phase. Rather than just obtaining artifacts, the third phase consists of the critical evaluation of all artifacts. The fourth phase involves contributing to the knowledge base in the form of conference proceedings and journal articles. Figure 16 illustrates the three research phases and the research questions, including articles relevant to each phase and question.

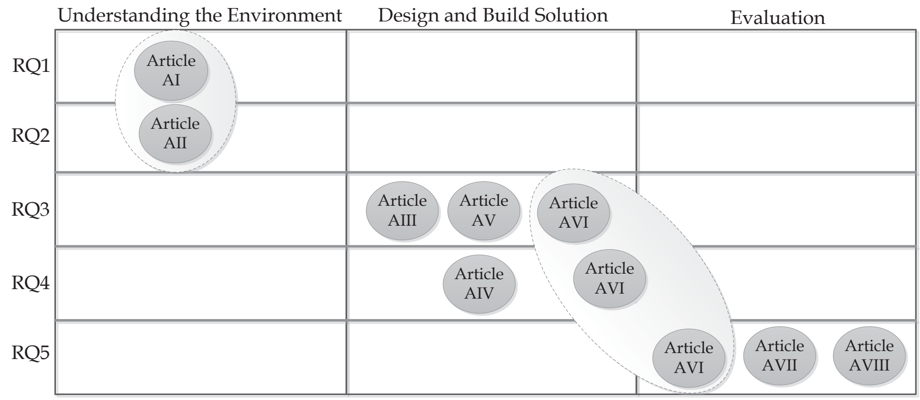

FIGURE 16 Positioning of articles between phases and RQ.

RQ1: What are the most common challenges in performing RE work based on existing literature?

The problems and challenges stakeholders face in RE are identified (Section 5.1 and articles [AI] and [AII]). The results of this study reveal the following challenges:

(a) It is often difficult to make stakeholders with different backgrounds fully understand when communicating requirements.
(b) Much effort is required when eliciting stakeholders' needs and translating those needs into system requirements.
(c) With current visual techniques such as UML and goal-oriented models, inexperienced users experience difficulty.

RQ2: How can RE work be supported by iconic communication?

RE areas that need additional mechanisms for their improvement and the ability of icons to facilitate communication among multiple stakeholders are discussed (Section 5.1 and articles [AI] and [AII]).

Icon-based information as one type of visual form has been introduced and is expected to help multifaceted stakeholders communicate, understand and manage their requirements in a simple and friendly way. Icons that are properly designed have the power to capture users' attention and speed up recognition.

RQ3: How specific tasks of RE could be supported by icons?

Principal requirements artifacts that could be enhanced by icons are identified. In addition, icons artifacts thought to improve RE work have been proposed (Sections 5.2.1 and 5.2.3 and articles [AIII] and [AV]). Besides, some of this study in Section 5.3.1 and article [AVI] helps improve RE work that would be supported by icons.

The results of this study show that (a) requirements properties such as attributes and relationships are possible to be designated by icons and (b) distinguishing activities in each RE process, where icons can enhance understanding, could make tasks more clear cut for stakeholders to deal with requirements.

RQ4: How could multi-culturality be supported in icon design for RE?

Social contracts that relate to the recognition of icons' meanings have been defined (Section 5.2.2 and article [AIV]). The patterns that can be used for the design of icons are produced (Section 5.3.1 and article [AVI]).

The results of this study show that icon conventions can be the basis for designing intuitive icons. Even if icons are strictly designed according to the guidelines and conventions, it is often the case that the interpretation of icons is strongly dependent on interpreters' cultural background age, experience, computer literacy, and education level.

RQ5: How to evaluate a proposed solution to confirm if it is usable for multifaceted stakeholders?

The scenarios for empirical evaluation are determined, such as evaluation methods, a series of test tasks, a group of questionnaires and interview's topics and a target set of participants. The demonstrators and their peripherals are developed. The statistical results obtained from the first iteration by students in the RE course at the Department of Mathematical Information Technology, University of Jyväskylä are presented (Sections 5.3.1 and 5.3.2 and articles [AVI] and [AVII]). The statistical results obtained from the second iteration by software companies and PhD students at the University of Jyväskylä are explained (Section 5.3.3 and article [AVIII]). To be more precise, the obtained results support the following three assumptions.

> ***Assumption I:*** *Enriching RE work with icons improves both usability and user satisfaction.*

In the first iteration of the evaluation by students, the formative results demonstrated that participants were satisfied with the icon-based approach from its usability and satisfaction, with a significantly more than 50 percent. Positive feedback about using icons was greater than the negative feedback

In the second iteration by experts and PhDs, the summative results reveal three findings. First, both experts and PhDs have perceived that icons would be usable in multifaceted environments, at a significant above average. Regarding the aesthetic perspectives (Hassenzahl et al., 2000), ergonomic quality (EQ), and hedonic quality (HQ), experts and PhDs gave EQ and HQ a higher than average rating.

Assumption II: *The icon-based approach is suitable for both novices and experts.*

One of the most important results concerns the ability of both novices and experts to understand icon-based approach. It was revealed that there is no significant difference between those groups in understanding and communicating requirements using icon-based approach.

Assumption III: *It is possible for icons to represent requirements attributes and relationships.*

The experiments showed that icons offer positive cognition from their metaphor, used to represent the concept of attributes and relationships. The formative experiments demonstrated that students interpret the meanings of icons from metaphors at a rate of accuracy of over 50 percent.

The summative experiments exhibited that experts and highly educated practitioners satisfy the use icon metaphors resembling concepts at an above-average level. However, the association between an icon and its meaning can be a many-to-many relationship, which means the same icon may represent different concepts or ideas and vice versa. Given this dichotomy, ambiguity through icon communication may occur. To make the outcome of icon recognition more reliable and decrease ambiguity, every icon must be designed carefully to include three elements: a graphic representation, a referent object, and interpreting procedure.

According to these results, representing RE work using icons appears to have bridged the gap of communication between technical and non-technical users. In addition, it has been perceived to offer an intuitive medium when communicating requirements based on its usability and users' satisfaction. Overall, icons have proven to be a feasible way for enriching RE work. Even though certain aspects of its implementation must be perfected, the viability of such an approach has been demonstrated through the evaluation results, which can be seen in the evaluation sections, and through the included articles. The results of the artifacts tests have shown that it is possible to use this approach to enhance RE work. Furthermore, with self-explanatory icons, the approach can be made even easier to lean and use.

6.2 Limitations

This dissertation study provides the opportunities to understand the challenges in RE area, and to engender and evaluate the new solution in coupling with RE work in specific tasks. Nevertheless, this study has confronted a number of limitations.

First, the challenges and research gaps presented in the literature acted as the primary focus of this study, which sought to engage with the current RE situation. Because other research was the primary platform from which relevant research areas and questions were identified, this study focuses purely on the research community and does not discuss other fields, such as the industrial dimension. In this regard, it would be beneficial to study application to actual practice from the industrial perspective.

Second, DSR examines whether an artifact works and does what it is meant to do (Gregor & Hevner, 2013). However, the limitation here is that one artifact (user preference adaptivity) was not evaluated to prove whether the concept would work appropriately. Another concern is that the RE artifact has been delineated for small segments of attributes and relationships. It would be useful for this artifact to be enlarged to include the generation of requirements with the enhanced icons.

Third, as with most novel approaches, this research has opened up possibilities for a new and exciting future direction. The results of formative and summative tests support the idea of icon-based approach. However, a broader range of participants and applications is needed to further investigate details. In particular, a larger summative group assessment is necessary. In addition, the participants in the summative evaluation were software specialists. Other perspectives and concerns were not analyzed. The evaluation must be expanded to include more subjects. A limited validation of the chosen icons and developed demonstrations during this study can also be seen as a limitation of the results. Further, all participants have a high education level and most also have a high level of computer literacy. In this respect, it would be worthwhile to determine whether the approach can be improved to take into account more aspects that influence culture, such as age, education level, and even computer literacy.

Finally, in this dissertation, cultural study has been restricted only to Finnish culture and the concepts of icons have been generalized based only on Hofstede's cultural dimensions. Although the experiments have demonstrated that proposed icons may enrich RE work, it would be exciting to see whether icon concepts that follow other models or a combination of Hofstede's approach with other variables could result in greater precision for a wider range of cultures.

6.3 Further Research

This study focused on designing an approach that will facilitate RE work while solving the identified challenges. All artifacts have been proposed as a part of a conceptual framework.

Regarding icons' design aspects, in the future, icons used to represent RE artifacts must be explicitly and concretely designed for a specific purpose. What an icon represents may not be obvious when it is first encountered, even though it may be possible to guess the meaning (Salman et al., 2012).

User preference modeling can be extended to support both the personal and cultural preferences of users and it must be carefully evaluated. In particular, a non-adoption and adoption must be made to determine whether cultural background influences the way readers interpret and use icons.

Requirements artifacts have now been identified in the small fragments of attributes and relationships. The next step is to incrementally include other relevant artifacts, such as a classification of requirements types, like UML or goal-oriented models.

Scalability and granularity must also be studied further. This first version of icon-approach artifacts has the potential to compose only simple representations. In the future, all functionalities of the complete language should be possible.

YHTEENVETO (FINNISH SUMMARY)

Ohjelmiston vaatimusmäärittelyn tarkoituksena on selvittää ne ohjelmistotuotteelle asetetut tavoitteet, mitkä valmiin järjestelmän tulisi täyttää. Vaatimusanalyysi tehdään tiiviissä yhteistyössä ohjelmiston asiakkaan kanssa. Usein hankkeeseen liittyy myös muita keskeisiä toimijoita ja sidosryhmiä kuten eri alojen viranomaiset esimerkiksi lääketieteellisissä ja ilmailuun liittyvissä sovelluksissa. Vaatimusanalyysi tuottaa lopputuloksenaan dokumentin, jossa kuvataan ohjelmistoprojektin tavoitteita ja vaatimuksia. Siinä määritellään, miten lopullisen ohjelmiston tulisi toimia ja millä keinoilla nämä toiminnallisuudet saavutetaan. Toiminnallisuuksia kuvataan usein käyttötapausten avulla. Käyttötapaukset kuvaavat käyttäjän ja ohjelmiston välistä vuorovaikutusta. Vaatimusmäärittely jaotellaan pääpiirteissään kahteen osaan eli toiminnallisiin ja ei-toiminnallisiin vaatimuksiin. Ei-toiminnallisia vaatimuksia ovat laadulliset vaatimukset ja resurssivaatimukset. Vaatimusmäärittelytyö on innovatiivista, tieto- ja osaamisintensiivistä toimintaa, jossa asiantuntijaryhmiä johtavilla henkilöillä on keskeinen rooli, erityisesti monikulttuurisissa toimintaympäristöissä. Globaalissa ohjelmistokehitystyössä kohtaavat eri kansallisuudet, erilaiset organisaatiot ja erilaiset tiimikulttuurit. Vaatimusten määrittelytyö on ennen kaikkea kommunikointia eri sidosryhmien välillä. Globaalissa ohjelmistokehitystyössä monitahoinen sidosryhmäympäristö asettaa uusia haasteita kommunikaatiolle.

Tämä väitöskirja käsittelee visuaalista, ikoneihin perustuvaa, kommunikointia, jonka avulla pyritään tukemaan vaatimusmäärittely- ja vaatimusten hallintatyötä erityisesti monitahoisissa sidosryhmäympäristöissä. Tässä työssä esitetään hajautettuun vaatimusmäärittely- ja vaatimusten hallintaympäristöön kehitetty ikonikieli. Työn konstruktiivisessa osassa on kehitetty hajautettuun vaatimustyöhön ikoni-demonstraattori MediaWiki-ympäristössä. Työ sisältää kaksivaiheisen evaluoinnin MediaWiki-ympäristössä. Evaluointiin osallistui kokeneita ja vasta-alkavia vaatimusmäärittelijöitä. Evaluoinnin tulokset ovat hyvin rohkaisevia jatkokehityksen kannalta, jossa kiinnostavana haasteena on ikonien suunnittelu ja tulkinta monikulttuurisessa vaatimustyö-ympäristössä.

REFERENCES

Ackerman, S. K. 2002. Mapping User Interface Design to Culture Dimensions. Proceedings of IWIPS, 89.

Agarwal, B. B., Tayal, S. P. & Gupta, M. 2010. Software Engineering and Testing. The United States of America: John and Bartlett Publishers.

Aghaee, S. & Pautasso, C. 2014. End-User Development of Mashups with NaturalMash. Journal of Visual Languages & Computing 25 (4), 414-432.

Ainsworth, S. 2006. DeFT: A Conceptual Framework for Considering Learning with Multiple Representations. Learning and Instruction 16 (3), 183-198.

Al-Rawas, A. & Easterbrook, S. 1996. Communication Problems in Requirements Engineering: A Field Study. Proceedings of the First Westminster Conference on Professional Awareness in Software Engineering, Royal Society, 47.

Apple Computer Inc, 1996. Newton 2.0 User Interface Guidelines. Addison-Wesley.

Ardito, C., Francesca Costabile, M., Desolda, G., Lanzilotti, R., Matera, M., Piccinno, A. & Picozzi, M. 2014. User-Driven Visual Composition of Service-Based Interactive Spaces. Journal of Visual Languages & Computing 25 (4), 278-296.

Aykin, N. 2005. Usability and Internationalization of Information Technology. Lawrence Erlbaum Association, Inc., Publisher.

Ballejos, L. C. & Montagna, J. M. 2011. Modeling Stakeholders for Information Systems Design Processes. Requirements Engineering 16 (4), 281-296.

Barker, P. 2000. Human Communication Process. In P. G. Barker & M. Yazdani (Eds.) USA: Intellect Ltd, 1-16.

Bendraou, R., Jezequel, J., Gervais, M. -. & Blanc, X. 2010. A Comparison of Six UML-Based Languages for Software Process Modeling. IEEE Transactions on Software Engineering 36 (5), 662-675.

Bera, P. & Evermann, J. 2014. Guidelines for Using UML Association Classes and Their Effect on Domain Understanding in Requirements Engineering. Requirements Engineering 19 (1), 63-80.

Boehm, B. 2006. A View of 20th and 21st Century Software Engineering. Proceedings of the 28th International Conference on Software engineering, 12.

Boehm, B. & Basili, V. R. 2001. Top 10 list [software development]. Computer 34 (1), 135-137.

Booch, G., Maksimchuk, R. A., Engle, M. W., Young, B. J., Conallen, J. & Houston, K. A. 2007. Object-Oriented Analysis and Design with Applications. (3rd edition) The Addison-Wesley.

Booch, G., Rumbaugh, J. & Jacobson, I. 2005. The Unified Modeling Language User Guide. (2nd edition) Addison Wesley Professional.

Britton, C. & Jones, S. 1999. The Untrained Eye: How Languages for Software Specification Support Understanding in Untrained Users. Human-Computer Interaction 14 (1), 191-244.

Brun, Y., Holmes, R., Ernst, M. D. & Notkin, D. 2013. Early Detection of Collaboration Conflicts and Risks. IEEE Transactions on Software Engineering 39 (10), 1358-1375.

Budgen, D., Burn, A. J., Brereton, O. P., Kitchenham, B. A. & Pretorius, R. 2011. Empirical Evidence About the UML: A Systematic Literature Review. Softw. Pract. Expert 41 (1), 363-392.

Callahan, E. 2006. Cultural Similarities and Differences in the Design of University Web sites. Journal of Computer-Mediated Communication 11 (1), 239-273.

Carrillo de Gea, J. M., Nicolas, J., Aleman, J. L. F., Toval, A., Ebert, C. & Vizcaino, A. 2011. Requirements Engineering Tools. IEEE Software 28 (4), 86-91.

Carrillo de Gea, J. M., Nicolás, J., Fernández Alemán, J. L., Toval, A., Ebert, C. & Vizcaíno, A. 2012. Requirements Engineering Tools: Capabilities, Survey and Assessment. Information and Software Technology 54 (10), 1142-1157.

Carrizo, D., Dieste, O. & Juristo, N. 2014. Systematizing Requirements Elicitation Technique Selection. Information and Software Technology 56 (6), 644-669.

Cerpa, N. & Verner, J. M. 2009. Why did Your Project Fail? Communication of the ACM 52 (12), 130-134.

Chandler, D. 2007. Semiotics: the basics. (2nd edition) the Taylor & Francis.

Cheng, P. C., Lowe, R. K. & Scaife, M. 2001. Cognitive Science Approaches To Understanding Diagrammatic Representations. Artif. Intell. Rev. 15 (1-2), 79-94.

Cheng, B. H. C. & Atlee, J. M. 2007. Research Directions in Requirements Engineering. FOSE'07 Future of Software Engineering, 285.

Cheng, P. C., Lowe, R. K. & Scaife, M. 2014. Graphical Notations for Syllogisms: How Alternative Representations Impact the Accessibility of Concepts. Journal of Visual Languages & Computing 25 (3), 170-185.

Cockburn, A. 2001. Writing Effective Use Cases. (6th edition) Addison-Wesley.

Costagliola, G., Deufemia, V. & Polese, G. 2004. A Framework for Modeling and Implementing Visual Notations With Applications to Software Engineering. ACM Transactions on Software Engineering and Methodology 13 (4), 431-487.

Coughlan, J. & Macredie, R. D. 2002. Effective Communication in Requirements Elicitation: A Comparison of Methodologies. Requirements Engineering 7 (2), 47-60.

Danado, J. & Paternò, F. 2014. Puzzle: A Mobile Application Development Environment Using a Jigsaw Metaphor. Journal of Visual Languages & Computing 25 (4), 297-315.

Dardenne, A., van Lamsweerde, A. & Fickas, S. 1993. Goal-Directed Requirements Acquisition. Science of Computer Programming 20 (1–2), 3-50.

Deacon, T. W. 1997. The Symbolic Species: The Co-evolution of Language and the Brain. New York London: W. W. Norton & Company, Inc.

Decker, B., Ras, E., Rech, J., Jaubert, P. & Rieth, M. 2007. Wiki-Based Stakeholder Participation in Requirements Engineering. IEEE Software 24 (2), 28-35.

Duboc, L., Letier, E. & Rosenblum, D. S. 2013. Systematic Elaboration of Scalability Requirements through Goal-Obstacle Analysis. IEEE Transactions on Software Engineering 39 (1), 119-140.

Edsall, R. 2007. Cultural Factors in Digital Cartographic Design: Implications for Communication to Diverse Users. Cartography and Geographic Information Science 34 (2), 121-128.

El Emam, K. & Koru, A. G. 2008. A Replicated Survey of IT Software Project Failures. IEEE Software 25 (5), 84-90.

ETSI EG 202-048, 2002. Human Factors (HF); Guideline on the multimodality of icons, symbols, and pictograms.

Ferreira, D. & Silva, A. 2009. An Enhanced Wiki for Requirements Engineering. 35th Euromicro Conference on Software Engineering and Advanced Applications SEAA'09, 87.

Figl, K., Derntl, M., Rodriguez, M. C. & Botturi, L. 2010. Cognitive Effectiveness of Visual Instructional Design Languages. Journal of Visual Languages & Computing 21 (6), 359-373.

Fitrianie, S., Datcu, D. & Rothkrantz, L. J. M. 2007. Human Communication Based on Icons in Crisis Environments*. Usability and Internationalization, Springer-Verlag Berlin Heidelberg, 57-66.

Fitrianie, S. & Rothkrantz, L. J. M. 2007. A Visual Communication Language for Crisis Management. International Journal of Intelligent Control and Systems 12 (2), 208-216.

Galitz, O. W. 2007. The Essential Guide to User Interface Design An Introduction to GUI Design Principles and Techniques. (3rd edition) Wiley Publishing, Inc.

Gotel, O. C. Z. & Finkelstein, A. C. W. 1994. An Analysis of the Requirements Traceability Problem. Proceedings of the First International Conference on Requirements Engineering, 94.

Gotel, O. C. Z., Marchese, F. T. & Morris, S. J. 2008. The Potential for Synergy between Information Visualization and Software Engineering Visualization. 12th International Conference on Information Visualisation, 547.

Gregor, S. & Hevner, A. R. 2013. Positioning and Presenting Design Science Research For Maximum Impact. MIS Quarterly 37 (2), 337-355.

Hadar, I., Reinhartz-Berger, I., Kuflik, T., Perini, A., Ricca, F. & Susi, A. 2013. Comparing the Comprehensibility of Requirements Models Expressed in Use Case and Tropos: Results from a Family of Experiments. Information and Software Technology 55 (10), 1823-1843.

Hall, T., Beecham, S. & Rainer, A. 2002. Requirements Problems in Twelve Software Companies: an Empirical Analysis. IEE Proceedings Software 149 (5), 153-160.

Hansen, S. & Lyytinen, K. 2010. Challenges in Contemporary Requirements Practice. 43rd Hawaii International Conference on System Sciences (HICSS), 1.

Hassenzahl, M., Platz, A., Burmester, M. & Lehner, K. 2000. Hedonic and Ergonomic Quality Aspects Determine a Software's Appeal. Proceedings of the

International Conference on Human Factors in Computing Systems, 201.
Heimbürger, A., Kiyoki, Y. & Kohtala, S. 2012. Intelligent Icons for Cross-Cultural Knowledge Searching. In J. Henno, Y. Kiyoki, T. Tokuda, H. Jaakkola & N. Yoshida (Eds.). Information Modeling and Knowledge Bases XXIII. ISO Press, Amsterdam, 77-89.
Heimbürger, A., Kiyoki, Y. & Ylikotila, T. 2011. Communication Across Cultures in the Context of Multicultural. Software Development. Reports of the Department of Mathematical Information Technology. Series C. Software and Computational Engineering.
Heimbürger, A., Duzi, M., Kiyoki, Y., Sasaki, S. & Khanom, S. 2014. Cross-Cultural Communication with Icons and Images. In T. Tokuda, Y. Kiyoki, H. Jaakkola & N. Yoshida (Eds.) Information Modeling and Knowledge Based XXV. ISO Press, Amsterdam, 306-322.
Heimbürger, A. & Kiyoki, Y. 2010. Pictorial Symbols in Context - A Means for Visual Communication in Cross-Cultural Environments. Proceedings of the IADIS International Conferences, IADIS Press, 463.
Heimgärtner, R. & Holzinger, A. 2005. Towards Cross-Cultural Adaptive Driver Navigation Systems. Usability Symposium, Austrian Computer Society 198, 53-68.
Helming, J., Koegel, M., Schneider, F., Haeger, M., Kaminski, C., Bruegge, B. & Berenbach, B. 2010. Towards a Unified Requirements Modeling Language., 2010 Fifth International Workshop on Requirements Engineering Visualization (REV), 53.
Hevner, A. & Chatterjee, S. 2010. Design Science Research in Information Systems. Integrated Series 9 in Information Systems 22, Springer Science+Business Media, 9.
Hevner, A. R., March, S. T., Park, J. & Ram, S. 2004. Design Science in Information Systems Research. MIS Quarterly 28 (1), 74-105.
Hofmann, H. F. & Lehner, F. 2001. Requirements Engineering as a Success Factor in Software Projects. IEEE Software 18 (4), 58-66.
Hofstede, G. 1997. Cultures and Organizations: Software of the Mind. New York: McGraw-Hill.
Horkoff, J. & Yu, E. 2013. Comparison and Evaluation of Goal-Oriented Satisfaction Analysis Techniques. Requirements Engineering 18 (3), 199-222.
Huang, W., Eades, P. & Hong, S. 2014. Larger Crossing Angles Make Graphs Easier to Read. Journal of Visual Languages & Computing 25 (4), 452-265.
IEEE Std 1220-1998, 1998. IEEE Standard for Application and Management of the System Engineering Process.
IEEE Std 610.12-1990, 1990. IEEE Standard Glossary of Software Engineering Terminology.
IEEE Std 830-1998, 1998. IEEE Recommended Practice for Software Requirements Specifications.
Isherwood, S. 2009. Graphics and Semantics: The Relationship between What Is Seen and What Is Meant in Icon Design. In D. Harris (Ed.) Engineering Psychology and Cognitive Ergonomics.Heidelberg: Springer-Verlag Berlin,

197-205.
Isherwood, S. J., McDougall, S. J. P. & Curry, M. B. 2007. Icon Identification in Context: The Changing Role of Icon Characteristics With User Experience. Human Factors: The Journal of the Human Factors and Ergonomics Society 49 (3), 465-476.
ISO/IEC 11581, 2002. Information technology -- User System Interfaces -- Icon symbols and functions -- Part 1: General Icons, 2: Object Icons, 6: Action Icons.
ISO/IEC 9126, 1991. IT- Software Product Evaluation – Quality characteristics.
ISO/IEC/IEEE 29148, 2011. Systems and Software Engineering —Life Cycle Processes — Requirements Engineering.
Jacobson, I., Booch, G. & Rumbaugh, J. 1999. The Unified Software Development Process. Addision-Wesley.
Kaindl, H. 1999. Difficulties in the Transition from OO Analysis to Design. Software, IEEE 16 (5), 94-102.
Kaiya, H., Shinbara, D., Kawano, J. & Saeki, M. 2005. Improving the Detection of Requirements Discordances Among Stakeholders. Requirements Engineering 10 (4), 289-303.
Katina, P. F., Keating, C. B. & Jaradat, R. M. 2014. System Requirements Engineering in Complex Situations. Requirements Engineering 19 (1), 45-62.
King, A. J. 2000. On the Possibility and Impossibility of a Universal Iconic Communication System. In P. G. Barker & M. Yazdani (Eds.) USA: Intellect Ltd, 17-28.
Kelanti, M., Hyysalo, J., Kuvaja, P. & Oivo, M. 2013. A Case Study of Requirements Management: Toward Transparency in Requirements Management Tools. The Eighth International Conference on Software Engineering Advances ICSEA, 597.
Kotonya, G. 1999. Practical Experience with Viewpoint-Oriented Requirements Specification. Requirements Engineering 4 (3), 115-133.
Lang, M. & Duggan, J. 2001. A Tool to Support Collaborative Software Requirements Management. Requirements Engineering 6 (3), 161-172.
Lapouchnian, A. 2005. Goal-Oriented Requirements Engineering: An Overview of the Current Research. Department of Computer Science University of Toronto.
Leidner, D. E. & Kayworth, T. 2006. A Review of Culture in Information Systems: Toward a Theory of Information Technology Culture Conflict. MIS Quarterly 30 (2), 357-399.
Li, L., Grundy, J. & Hosking, J. 2014. A Visual Language and Environment for Enterprise System Modeling and Automation. Journal of Visual Languages & Computing (in press).
Lindberg, T., Näsänen, R. & Müller, K. 2006. How age affects the speed of perception of computer icons. Displays 27 (4–5), 170-177.
Lodding, K. N. 1983. Iconic Interfacing. IEEE Computer Graphics and Applications 3 (2), 11-20.
Luna, E. R., Rossi, G. & Garrigós, I. 2011. WebSpec: a Visual language for Speci-

fying Interaction and Navigation Requirements in Web Applications. Requirements Engineering 16 (4), 297-321.

Mack, A., Pappas, Z., Silverman, M. & Gay, R. 2002. What We See: Inattention and the Capture of Attention by Meaning. Consciousness and Cognition 11 (4), 488-506.

Majchrzak, A., Wagner, C. & Yates, D. 2013. The Impact of Shaping on Knowledge Reuse for Organizational Improvement with Wikis. MIS Quarterly 37 (2-Appendices), 1-12.

Marcus, A. 2003. Icons, Symbols, and Signs: Visible Languages to Facilitate Communication. Interactions 10 (3), 37-43.

Mathiassen, L., Tuunanen, T., Saarinen, T. & Rossi, M. 2007. A Contigency Model for Requirements Development. Journal of the Association for Information Systems: 8 (11), 569-597.

McDougall, S. J. P. & Curry, M. B. 2007. More than just a Picture: Icon Interpretation in Context. Coping with Complexity Workshop, University of Bath, 1.

McDougall, S. J. P., Curry, M. B. & Bruijn, O. d. 2001. The Effects of Visual Information on Users' Mental Models: An Evaluation of Pathfinder Analysis as a Measure of Icon Usability. International Journal of Cognitive Ergonomics 5 (1), 59-84.

McDougall, S. & Isherwood, S. 2009. What's in a Name? The Role of Graphics, Functions, and their Interrelationships in Icon Identification. Behavior Research Method 41 (2), 325-336.

McGee, S. & Greer, D. 2012. Towards an Understanding of the Causes and Effects of Software Requirements Change: Two Case Studies. Requirements Engineering 17 (2), 133-155.

McGuinness, D.L. & Harmelen van, F. 2004. OWL Web Ontology Language Overview. Available in: http://www.w3.org/TR/owl-features/. Accessed: 15 January 2014.

Meziane, F., Athanasakis, N. & Ananiadou, S. 2008. Generating Natural Language specifications from UML class diagrams. Requirements Engineering 13 (1), 1-8.

Microsoft, 2008. Design Guidance Exploration Icons and Symbology.

Moody, D. L., Heymans, P. & Matulevicius, R. 2010. Visual Syntax Does Matter: Improving the Cognitive Effectiveness of the *i** Visual Notation. Requirements Engineering 15 (2), 141-175.

Moody, D. L. & Hillegersberg, J. v. 2009. Evaluating the Visual Syntax of UML: An Analysis of the Cognitive Effectiveness of the UML Family of Diagrams. In Software Language Engineering, 16-34.

Moody, D. L. 2009. The "Physics" of Notations: Toward a Scientific Basis for Constructing Visual Notations in Software Engineering. IEEE Transactions on Software Engineering 35 (6), 756-779.

Morris, S. & Spanoudakis, G. 2001. UML: an Evaluation of the Visual Syntax of the Language. Proceedings of the 34th Annual Hawaii International Conference on System Sciences, 1.

Nakamura, C. & Zeng-Treitler, Q. 2012. A Taxonomy of Representation Strategies in Iconic Communication. International Journal of Human-Computer Studies 70 (8), 535-551.

Ncube, C., Soo Ling Lim & Dogan, H. 2013. Identifying Top Challenges for International Research on Requirements Engineering for Systems of Systems Engineering. 21st IEEE International Requirements Engineering Conference (RE), 342.

Nuseibeh, B. 2001. Weaving Together Requirements and Architectures. Computer 34 (3), 115-117.

Nuseibeh, B. & Easterbrook, S. 2000. Requirements Engineering: A Roadmap. Proceedings of the Conference on The Future of Software Engineering, 35.

OMG, 2006. Meta Object Facility (MOF) Core Specification.

Orford, S. 2005. Cartography and Visualization. In N. Castree, A. Rogers & D. Sherman (Eds.) Questioning Geography Fundamental Debates India: Blackwell Publishing Ltd, 189-204.

Peffers, K., Tuunanen, T., Rothenberger, M. A. & Chatterjee, S. 2007. A Design Science Research Methodology for Information Systems Research. Journal of Management Information Systems 24 (3), 45-78.

Peirce, C. S. 1931. Collected Papers of Charles Sanders Peirce. Cambridge, Harvard University Press.

Petre, M. 1995. Why Looking Isn't Always Seeing: Readership Skills and Graphical Programming. Communication of ACM 38 (6), 33-44.

Pitula, K. & Radhakrishnan, T. 2011. On Eliciting Requirements from End-Users in the ICT4D Domain. Requirements Engineering 16 (4), 323-351.

Pohl, K. 2010. Requirements Engineering Fundamentals Principles and Techniques. Berlin: Springer.

Pohl, K. & Rupp, C. 2011. Requirements Engineering Fundamentals: A Study Guide for the Certified Professional for Requirements Engineering Exam - Foundation Level - IREB compliant. (1st edition) Rocky Nook.

Pozgaj, Z. 2000. Requirement Analysis Using VORD. Proceedings of the 22nd International Conference on Information Technology Interfaces, 105.

Quispel, A. & Maes, A. 2014. Would You Prefer Pie or Cupcakes? Preferences for Data Visualization Designs of Professionals and Laypeople in Graphic Design. Journal of Visual Languages & Computing 25 (2), 107-116.

Regnell, B., Runeson, P. & Thelin, T. 2000. Are the Perspectives Really Different? - Further Experimentation on Scenario-Based Reading of Requirements. Empirical Softw. Engg. 5 (4), 331-356.

Reinecke, K. & Bernstein, A. 2013. Knowing What a User Likes: A Design Science Approach to Interfaces that Automatically Adapt to Culture. MIS Quarterly 37 (2), 427-453.

Reinecke, K. & Bernstein, A. 2011. Improving Performance, Perceived Usability, and Aesthetics with Culturally Adaptive User Interfaces. ACM Transactions on Computer-Human Interaction 18 (2), 8-29.

Rolland, C., Souveyet, C. & Ben, A. C. 1998. Guiding Goal Modeling Using Scenarios. IEEE Transactions on Software Engineering 24 (12), 1055-1071.

Saiedian, H. & Dale, R. 2000. Requirements Engineering: Making the Connection between the Software Developer and Customer. Information and Software Technology 42 (6), 419-428.

Salman, Y. B., Cheng, H. & Patterson, P. E. 2012. Icon and User Interface Design for Emergency Medical Information Systems: A case Study. International Journal of Medical Informatics 81 (1), 29-35.

Simon, H. A. 1996. The Sciences of the Artifacts. (3rd edition) the MIT Press, Cambridge, Massachusetts.

Sateli, B., Angius, E. & Witte, R. 2013. The ReqWiki Approach for Collaborative Software Requirements Engineering with Integrated Text Analysis Support. IEEE 37th Annual Computer Software and Applications Conference (COMPSAC), 405.

Seyff, N., Maiden, N., Karlsen, K., Lockerbie, J., Grünbacher, P., Graf, F. & Ncube, C. 2009. Exploring How to Use Scenarios to Discover Requirements. Requirements Engineering 14 (2), 91-111.

Shen, S., Woolley, M. & Prior, S. 2006. Towards Culture-Centred Design. Interacting with Computers 18 (4), 820-852.

Sommerville, I. & Kotonya, G. 1998. Requirements Engineering: Processes and Techniques. New York, NY, USA: John Wiley & Sons, Inc.

Sommerville, I. & Sawyer, P. 1997. Viewpoints: Principles, Problems and a Practical Approach to Requirements Engineering. Ann. Softw. Eng. 3, 101-130.

Sommerville, l. 2011. Software Engineering. (9th edition) United States of America: Addison-Wesley.

The Standish Group 2013, 2013. CHAOS Manifesto 2013. Boston.

Sutcliffe, A. 2003. Scenario-Based Requirements Engineering. Proceedings of 11th IEEE International Requirements Engineering Conference, 320.

Sutcliffe, A. G., Maiden, N. A. M., Minocha, S. & Manuel, D. 1998. Supporting Scenario-Based Requirements Engineering. IEEE Transactions on Software Engineering 24 (12), 1072-1088.

Sutton, D. C. 2000. Linguistic Problems with Requirements and Knowledge Elicitation. Requirements Engineering 5 (2), 114-124.

Tatomir, B. & Rothkrantz, L. 2005. Crisis Management using Mobile ad-hoc Wireless Networks. Proceedings of the 2nd International Crisis Management using mobile ad-hoc wireless networks, 147.

Thanasankit, T. & Corbitt, B. 2002. Understanding Thai Culture and Its Impact on Requirements Engineering Process Management during Information System Development. Asian Academy of Management Journal 7 (1), 103-126.

Ur Rehman, N., Bibi, S., Asghar, S. & Fong, S. 2010. Comparative Study of Goal-Oriented Requirements Engineering. 4th International Conference on New Trends in Information Science and Service Science (NISS), 248.

Van Lamsweerde, A. 2001. Goal-oriented requirements engineering: a guided tour. Proceedings of Fifth IEEE International Symposium on Requirements Engineering, 249.

Van Lamsweerde, A. 2000. Requirements engineering in the year 00: a research perspective. Proceedings of the 2000 International Conference on Software Engineering, 5.

Vassev, E. & Hinchey, M. 2013. Autonomy Requirements Engineering. IEEE 14th International Conference on Information Reuse and Integration (IRI), 175.

Venkatesh, V., Morris, G. M., Davis, B. G. & Davis, D. F. 2003. User Acceptance of Information Technology: Toward a Unified View. MIS Quarterly 27 (3), 425-478.

Wiegers, K. & Beatty, J. 2013. Software Requirements (Best Practices). (3rd edition) USA: Microsoft Press.

Yang, L., Prasanna, R. & King, M. 2014. GDIA: Eliciting Information Requirements in Emergency First Response. Requirements Engineering, 1-18.

Yu, Y. & Junda, H. 2010. An Analysis of Users' Cognitive Factors Towards Icon in Interactive Interface. 2nd International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), 26.

Yu, E. S. K. 1997. Towards Modeling and Reasoning Support for Early-Phase Requirements Engineering. Proceedings of the Third IEEE International Symposium on Requirements Engineering, 226.

Zave, P. 1997. Classification of Research Efforts in Requirements Engineering. ACM Comput. Surv. 29 (4), 315-321.

Zhang, Z., Arvela, M., Berki1, E., Muhonen, M., Nummenmaa, J. & Poranen, T. 2010. Towards Lightweight Requirements Documentation. Journal of Software Engineering & Applications 3 (9), 882-889.

ORIGINAL PAPERS

I

ICON-BASED LANGUAGE IN REQUIREMENTS DEVELOPMENT

by

Sukanya Khanom, Anneli Heimbürger & Tommi Kärkkäinen, 2012

In Y. Kiyoki & T. Tokuda & N. Yoshida (Eds.) 22nd European Japanese Conference on Information Modeling and Knowledge Bases. Prague, Czech Republic, June 4-9, 20-25.

Icon-based Language in Requirements Development

Sukanya KHANOM[1], Anneli HEIMBÜRGER and Tommi KÄRKKÄINEN
University of Jyväskylä, Department of Mathematical Information Technology, Finland

Abstract. Most errors and misunderstandings in requirements engineering and system development owe to poor communication between users and analysts. Icon-base language is an appropriate means to decrease the difficulty of communication in multi-user environments and among different user backgrounds. It is no wonder that visualized language using icons, symbols and graphics has had a positive reception by software projects in the area of requirements engineering. The primary purpose of this paper is to sketch a tool to support requirements development in multifaceted stakeholder environments in a wiki system. The paper also introduces some new means to apply visual language in the representation of situations or activities which refer to context, in requirements development.

Keywords. Icon-based language, graphic, visualization, stakeholders, requirements development.

Introduction

Requirements engineering has become a crucial component of software development in our sophisticated technological world. It is important because software development project are frequently undertaken without a good understanding of needs and desires of stakeholders. Consequently, performing basic requirements development activities elicitation, analysis, negotiation and validation, they can sufficiently reduce risk of project failure. To support such argument, CHOAS[2] reveals that good requirements engineering practices contribute more than 42% towards the overall success of a project, relatively, greater than other factors. In addition, according to Emam and Koru [1], project failures and cancellations fall into the requirements engineering phase due to the changes in requirements and scopes. Most of these failures are not found until late during the project or when the system has already gone live. One of the inevitable challenges in requirements development is that there are communication problems due to the differences in language, knowledge and culture. There are various approaches to solving this communication problem. Our paper introduces one of them, together with these key concepts: icon-based language, requirements development, context, multifaceted stakeholder environments.

Icons have been developed to aid thinking. Extended usage can evolve iconic set to more stylized and ultimately abstract representations as pictures from antiquity, maps from ancient Egypt and the geometry diagrams of Euclid [2], to mention but few.

---

[1] Corresponding Author.

[2] http://kinzz.com/resources/articles/91-project-failures-rise-study-shows

Requirements engineering in software development typically is classified into two main categories: *requirements development* and *requirements management*. There is a multitude of sub-items associated to requirements development; requirements elicitation, requirements analysis, requirements negotiation, requirements documentation, and requirements validation.

Generally speaking, the concept of context is defined as a situation at hand or as a task a user or team is performing [3]. *Context* includes high and low context, depending on the amount of information given in a communication [4].

When dealing with *multifaceted stakeholder environments*, cross-cultural or cross-linguistic, the unavoidable issue is communication problems. Therefore, it is essential that all communications are generated clearly, ethically, consistently, completely and in a timely fashion [5]. Culture resides in the way we interact with other individuals and with our environment in different situations. Cross-culture is composed of interaction among humans, between humans and machines, and between human and environment [3, 4, 6]. In multifaceted stakeholder environments, universal language is necessary to convey common understanding among different languages and cultures [7].

The following section introduces visual language and its challenges. Section 2 highlights the potential of icons in the context of requirements development. In section 3, we describe the implementation of icon-base language in the area of requirements development. Discussion and future research is presented in Section 4.

1. Visual Language

Requirements visualization is not new. There are several studies and techniques to demonstrate both the improvement in requirements engineering by applying visualized methods as well as the advantages of graphic in software development projects to support requirements engineering. Visualization is defined as a method of forming a mental vision, image, icon or picture of something not visible or present to the sight, or of an abstraction in order to make it visible to the mind or imagination [8, 9, 10]. Many fundamental visualization techniques that are widely accepted in both business and society have long been used in requirements engineering. These techniques typically include bar graphs, charts, and hierarchical structures, which are used regularly to aggregate large amounts of information into a single representation for shared understanding and swift absorption by stakeholders. Most regulars are nevertheless about prototype, storyboards, UML use case diagrams and mock-ups. Numerous researchers emphasize that day after day many companies encounter software project failure [1, 11] even though those visualization techniques are available for supporting requirements engineering practices. One example of visual difficulties is that use case can embody complex requirements for the system therefore detailed scenarios or specific circumstances need to be provided. When using existing visualizations such as Class diagram, UML or State chart the basic technical skill is required which sometimes non-technical users are unable to understand. As consequence, there is the room space for other types of visualization like icon language that can serve to enhance communication and understanding [10].

The deviation between icon-tool derived from this development and other sorts of implemented icons like iOS SDK is the sound of its purpose and usability. This icon-tool aims to distribute widespread receptions in requirements engineering taxonomies and requirements development activities among receivers as those of the other public icons. For instance, icons appear in human life mainly as communication means and they range from icons that are used to operate devices, through icons that appear in public places, airport, hotel, maps and traffic signs. Many of these icons are intended to provide the same information everywhere, visually speaking across languages. Thus, Icon-based language can be recognized as a part of mapmaking communication [3, 4, 9, 10]. On the contrary, iOS SDK (Software Development Kit), formally iPhone SDK, is a native application for iOS allowing developers to make application for the iPhone and iPod Touch.

2. Icons in the Context of Requirements Development

Requirements development process plays an essential role in software development. Therefore, universal and practical methods are needed to make it as simple as possible and to permit all parties understand the purpose. An effective method to communicate and transfer common perception among multiple users in requirements development is icon/graphic.

Intelligent icons, which should be user-friendly, informal and interactive, enable us to receive a better understanding of stakeholders' needs and move us from the technical domain, in which many developers are most comfortable, into the real-world problem domain. We can take advantage of the unique characteristics of icons to construct requirements taxonomies to help drive elicitation design, validation and negotiation. To improve the structure of the requirements categories and requirements development activities, icon-base provides, instead of itemized individual requirements, a sequence of actions between the system and the user. It does a better job in encouraging the users of the system, in a sequential fashion, to accomplish their goals with alternatives and exceptions [12]. Figure 1 depicts two different requirements development activities in multicultural forms. People in one culture are frequently unable to understand another culture therefore icons are capable of reducing impediments connected to misunderstandings arisen due to cross-cultural background environments and providing a common virtual board that can be shared by distributed stakeholders.

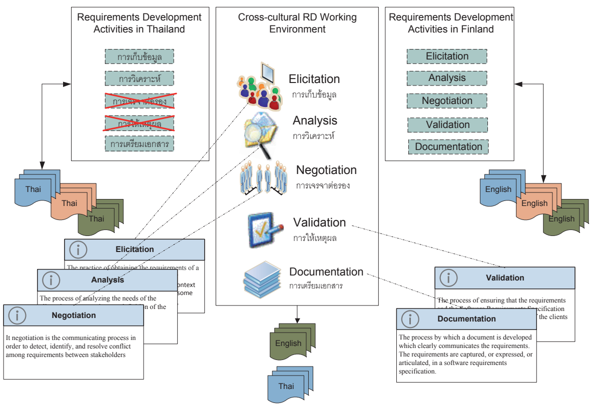

Figure 1. Examples of icons which represent the contexts of requirements development.

3. Towards Icon-Tool Implementation

Wikis as base software for the implementation are a great way to conduct the initial stage of requirements development process. They provide a means by which multiple stakeholders can work together to define requirements in an easy-to-access and easy-to-document manner, where collaboration is not only supported but encouraged by the nature of the technology itself. Figure 2 illustrates example icons used to represent each requirement type. The icons have embedded meaning in its representation. For example, when selecting of User Requirements, multi-subclass will be given by browsing information from wiki database. The demonstration offers link to requirements gathering with default information corresponding to a chosen requirement type such as requirements classification and requirements ID. The key driver of using wiki software in implementation icon-tool is because wiki provides easy page linking to reduce redundancy by making it easier to link content than to copy a page. Moreover, it supports historical page capture which strongly broads users for requirements tractability on a per-document [13, 14]. Specifically, it offers the way to handle and indicate misunderstandings and both expressed and unexpressed conflicts.

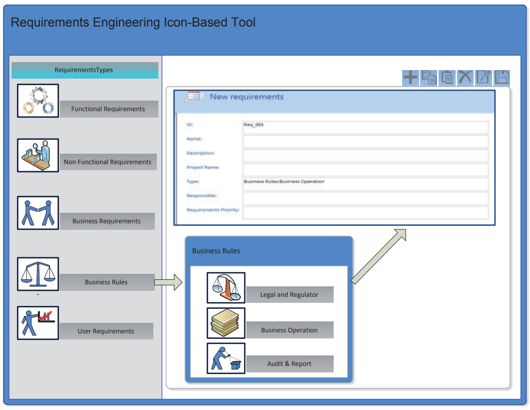

Figure 2. An example of icon-tool set as a framework in the context of requirements development in wiki.

4. Discussion and Future Research

Effective and efficient iconic artifacts are potentially able to reduce misconceptions and gaps in understanding by presenting many aspects of the requirements process. During the requirements phases, visualization can help in understanding the context for requirements development and provide the groundwork necessary for any requirements development process. To this we should add the importance of the development of visualization in promoting elicitation, negotiation, documentation, verification and validation in requirements development tasks.

Future research will focus on the construction of a set of icons providing a fundamental perception among multifaceted stakeholders especially, multicultural practitioners. The starting milestone for establishing an icon-tool project is to represent a set of icons in requirements engineering categories to address an ease of understanding and adoption. Furthermore, future research desires can also be kinds of requirements development activities and point of views identified to be supported by icons. The main contributions are as follow.

- The set of icons are built to support requirements development works and mitigate the communication gaps among stakeholders. The concept of icon sets will be developed based on both theoretical research and industrial survey.
- The identification of icons' characteristics to help clarify requirements development activities and hence enabling to transfer enough understanding for all stakeholders.
- One of the most valuable features is to distribute collaborative environment encouraging multifaceted stakeholders to brainstorming, elicitation, or validation the knowledge involved in requirements engineering projects.

The empirical evaluation of the developed tool intends to perform in two phases: an initial phase to obtain feedback for further improvement, and the second phase to repeat evaluation after proceeding modification. In an evaluation process, there are two practitioner groups students in the Requirements Engineering course (ITKS452) given by the university of Jyväskylä and in some software companies. Test tasks and interviews will be used as empirical evaluation methods.

References

[1] Emam, E. K. and Koru G. A. A. Replicated Survey of IT Software Project Failure, IEEE Computer Society, 2008.

[2] Ware, C. Information Visualization Perception for design, Second edition, 2005.

[3] Heimbürger, A., Kiyoki, Y. and Ylikotila, T. Communication Across Cultures in the Context of Multicultural Software Development. Reports of the Department of Mathematical Information Technology. Series C. Software and Computational Engineering. No. C 1/2011. 64 p. 2011.

[4] Heimbürger, A., Kiyoki, Y. and Kohtala, S. Intelligent Icons for Cross-Cultural Knowledge Searching. In: Henno, J., Kiyoki, Y., Tokuda, T. and Yoshida, N. (Eds.) Proceedings of the 21st European-Japanese Conference on Information Modelling and Knowledge Bases, Tallinn, Estonia, June 6 - 10, 2011. Tallinn University of Technology Press, Tallinn, Estonia. Pp. 84 - 96. 2011.

[5] Laplante P. A. Requirements Engineering for Software and System. Auerbach Publication. 2009.

[6] Heimbürger, A., Kiyoki Y., Kärkkäinen T., Gilman E., Kyoung-Sook K. and Yoshida N."On Context Modelling in Systems and Applications Development," In: Heimbürger A., Kiyoki Y., Tokuda T., Jaakkola H. and Yoshida N. (Eds.). Frontiers in Artificial Intelligence and Applications, Vol. 225 Information Modelling and Knowledge Bases XXII. Amsterdam: IOS Press. pp. 396-412, 2011.

[7] Aykin, N. Usability and Internationalization of Information Technology, 2005.

[8] Teyseyre, A. R. 3D Requirements Visualization, Journal of Computer Science and Technology, Vol. 3, No. 2, pp.45-51, 2003.

[9] Gotel, O., Marchese, F. T., and Morris, S. J. On Requirements Visualization, Second International Workshop on Requirements Engineering Visualization (rev'07), 2007.

[10] Cooperr, J. R., Lee, S.-W, Gandhi, R. A. and Gotel,O. Requirements Engineering Visualization: A Survey on the State-of-the-Art, Fourth International Workshop on Requirements Engineering Visualization (rev'09), 2009.

[11] Glass L. R. IT Failure Rates-70% or 10-15%, IEEE Computer Society, 2005.

[12] Hull, E., Jackson, K. and Dick, J. Requirement Engineering. Second Edition. Springer 2005.

[13] Fitrianie, S., Datcu,D. and Rothkrantz, J.M. Human Communication Based on Icons in Crisis, pp. 57-66, 2007.

[14] Abeti, L., Ciancarini, P and Moretti, R. Wiki-based Requirements Management for Business Process Reengineering, IEEE ICSC'09 Workshop, pp. 1-11, 2009.

II

ICON-BASED LANGUAGE IN THE CONTEXT OF REQUIREMENTS ENGINEERING

by

Sukanya Khanom, 2013

In R. B. Svensson, D. M. Berry, M. Daneva, et al. (Eds.) 19th International Working Conference on Requirements Engineering: Foundation for Software Quality (REFSQ). Essen, Germany, April 8-11, 215-222.

Icon-based Language in the Context of Requirements Engineering

Sukanya Khanom

University of Jyväskylä, Department of Mathematical Information Technology
Jyväskylä, Finland

sukanya.s.khanom@student.jyu.fi

Abstract. Requirements engineering (RE) has been enormously intensified by the need for simple method, easy to learn, and the desire to communicate with stakeholders holding different backgrounds. This paper introduces the icon based language intended to describe constructs for expressing situations that take part during the requirements process. Icon based language utilizes features of visual notations standard in RE domain incorporating with icon representa tions. Icon based language is designed by combining meta modeling concepts and notations for functional and non functional requirements. The main appli cation area includes RE contexts such as elicitation, analysis, validation and traceability. The primary contribution is aimed at providing the stakeholders with an intuitive and convenient communication environment by using icon based language for describing wide range of applications from business goals and requirements narratives to high level system analysis and design.

Keywords: Requirements engineering, Icon based language, Meta model, Stakeholder.

1 Background and Motivation

Requirements engineering (RE) has exponentially become an essential part of software development process [1]. Several software development problems arise from shortcomings in terms that stakeholders elicit, document, agree and amend the software's requirements [1], [2]. To date, many mechanisms such as goal-oriented, UML and scenario devised to allow the development teams and other stakeholders to discover, specify and review requirements [3]. Unfortunately, one considerable deficiency is the fact that they require knowledge and skill to achieve the tasks. Resolving this fence will accelerate interaction and communication of all stakeholders. For that reason, the advantages of visualization [4], [5], [6] drive the research to make greatly improve and eliminate the host barriers of technical-rich methods such as misinterpretation, misunderstanding and misconception.

The aim of the current research is to introduce an uncomplicated visual modelling method which is based primarily on iconic counterparts. Visual representation is one of the main ways that human beings communicate: it is social practice [7], [8] varying upon situations e.g. the sign language [8], diagrams [9], and comic illustrations [10]. The use of icons, symbols or signs in auxiliary communication makes visual representations different from natural language techniques that constructs on the basis of linear orderings of words [11]. Icons' meaning can be perceived straightly and they also encourage communication across international frontiers. Astonishingly, icons have been accepted successfully in human-computer interface, but seldom in RE visual notations [12], [13].

In this paper, we propose an icon-based language as a communication means for different stakeholders in RE. The study is focused to the construction of feasible visual sentences. The visual sentence is the composition of visual vocabulary, syntax, and semantics. Each construction is understood as a representation of a concept, an object, an action, or a relation.

The rest of the paper is structured as follows. The next section starts with the related work. Then, the research questions and methodology is presented. The following section presents the proposed solution of icon-based language in RE process. After that, up-to-date progress is explained. And final section is reserved for conclusion.

2 State of the Art

Moody [13] indicates that icons and visuals represent important benefit for communication research, especially to communicate about topics in which sound difficult for novices. Research attempts have been made to develop computer-intensive iconic communication systems, which primary aimed at fostering people to communicate with each other. Some systems are dedicated as a communication tool for the people who have speech disorders [8], [14]. Several proposals are rudimentary on linguistic theories such as the conceptual dependency theory introducing the concepts that the units of meaning correspond to the grammatical units of clause and words (e.g. [11], [15]). A profound research has been completed on designing the system that facilitates the reviewers to mutually communicate without sharing common language [15]. In the field of crisis management, Fitrianie et al. [11] announced a comprehensive icon-based interface that exploits graphic symbols to represent concepts or ideas.

In the RE community, wide variety of researches has highlighted on diagram for improving requirements engineering activities [16], [17], [18]. Extended features of use case diagram have been devised by Yang-Turner et al. [16] to support the tasks of stakeholders in elicitation activity. In similar manner, Helming et al. [18] approached an incremental UML as a communication means for delivering collaborative environment whereas, Cardei et al. [17] adapted the UML methodology into specification and validation phases to alleviate the gaps of requirements ambiguities and misinterpretations. Most diagrams make very diminutive use of semantic transparency and are abstract shapes whose meaning are articulately conventional and must be learnt. Visual notation in software engineering has been studies extensively by Moody [13] to define a series of principles for designing cognitively effective visual notation. He also advised to use pictorial icons that their meaning can be conceived directly and easily learnt to enhance cognitive effectiveness in all stages of RE process.

3 Research Questions

The premise research questions (RQ) of this paper are as the following:

- RQ 1: What are the difficulties users currently experienced in performing RE?
  - RQ 1-1: What are the existing problems still left to be solved?
- RQ 2: How can we support the tasks of software developers and other stakeholders in RE process with icon-based language, especially in multi-cultural environments?
  - RQ 2-1: What are the tasks of RE that can be supported by icons, for instance, requirements attributes and RE process etc.?
  - RQ 2-2: How can the concept be designed to take into account of the different cultures and behaviors in effective RE?
  - RQ 2-3: Who are the key stakeholders to be benefited from a proposed solution?
- RQ 3: How to validate if a proposed solution supports RE stakeholders and is easy to learn and understand?

In order to answer those questions, we use the design science research for icon-based language development as depicted in Fig. 1:

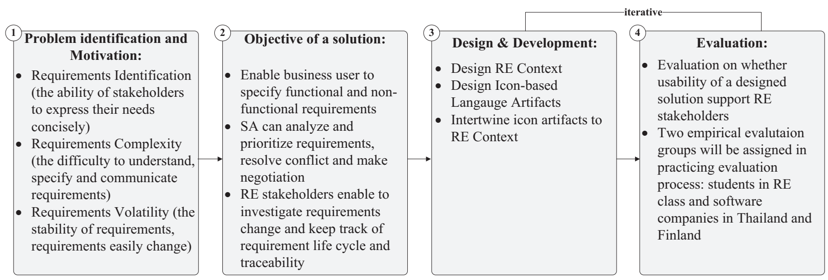

Fig. 1. Design science research methodology for icon based language

We first identify the problem of the whole RE context to understand real interests of stakeholders. According to literature review, we arrived at three problem identification in performing RE [19], [20]. Firstly, there is the ability challenge of system stakeholders to express their needs concisely and concretely. In broad spectrum, requirements are heavily hard to discover in situations where there is a communication gap between technical and non-technical users that appear to speak different languages and apply different approaches for desired outcomes. Secondly, requirements complexity happens when stakeholders encounter the difficulty to understand, specify and communicate requirements. Finally, requirements volatility signifies to the stability of requirements that easily change as a result of environmental dynamic or individual learning. We then define concrete objectives to inform the necessities of a possible solution to the aforementioned problems. Our main objective is to find a solution that provisions the RE stakeholders who have been influenced by multicultural backgrounds to specify requirements, analyze and prioritize requirements, resolve conflicts and make negotiation, and examine requirements change, requirements life cycle and traceability of requirements. At the design stage (see number 3 in Fig. 1), we will develop RE context and icon artifacts including the integration of those two worlds by means of Re process which begins with patterning the scope and vision, use case scenario and ends up with requirements specification. Following the theoretical evaluation, we empirically assess the utility and usability [21] of icon-based language whether it is simple enough to comprehend by RE stakeholders.

4 Proposed Solution: Icon-based Language

Icon-based language is proposed to provide RE stakeholders with technique that does not require advance knowledge, the development of more lightweight and intuitive interaction. In RE context, it is important to take into consideration the life cycle of the requirements as well as the attributes of the requirements. Icon-based language is defined rooted on the visual vocabularies visual grammars (syntax) and semantics [13], [15], [22].

- **Visual vocabularies or graphic symbols:** visual vocabulary is set of icons that annotate the visual sentence schemed on one-dimensional, two-dimensional and three-dimensional space and possibly associated through special relationships (see an example in Fig. 2). The iconic symbols consist of four actor types, three node types, five priority types, five status states, three dependency relationships, two parent-child relationships, two link types and one measurement bar for number of changes. Overall, actor represents stakeholders and systems that have the purpose and actions to achieve goals. Node symbolizes the various types of RE activities such as requirements taxonomy or elicitation tasks that we categorize to be individually exemplified by icon(s). For example, "goal" represents business requirements. Rationale signifies the scenario of activity that will be presented by icon(s). Typically, rationale dynamically describes the requirements' behavior under various conditions such as a group "priority" and a series of "status" states.

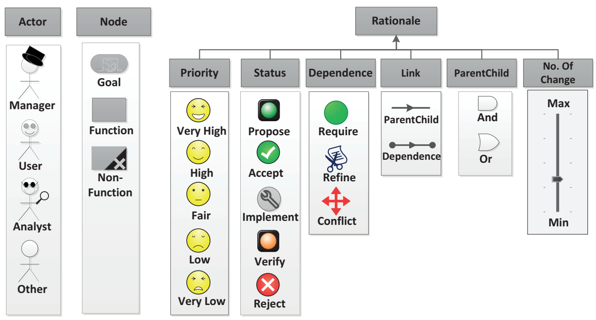

Fig. 2. Icon vocabularies for icon based language corresponding to a semantic structure

- **Visual grammatical rules/Syntaxes:** A set of icon elements and visual grammatical rules are altogether compounded to visual syntax. Currently, visual syntax is designed by applying the "Physics of Notation" theory [13]. We classify the visual vocabularies into three categories: actor, node and rationale as shown in Fig 2. Stick figures would be used to represent actors because they are universally interpreted for the representation of people. Variations of stick figures could help reader to distinguish the different types of actors. For example a stick wearing hat can be representative of manager. Node elements bear a resemblance to concrete icons together with geometrical shapes that are, however, globally accepted. For the remainder, we use abstract objects that are easily recognized such as the mathematical signs emotional faces, different line connectors of arrows and logical signs. In addition, color is used to improve cognitive effectiveness such as green, red and yellow colors which are the standard color for traffic sign.
- **Visual semantics:** The semantics of an icon-based language is represented in the meta-model form. Each syntactic creature is arranged to some semantic construct. As an example of parent-child relationship described in Fig. 3, the announcement of two banking transactions for account inquiry and money transfer involves several relevant requirements. Some of them, for instance, are:
  - FunReq 001: The system shall provide customer two transaction types, account inquiry and money transfer.
  - FunReq 001 1: Customer is able to inquiry only accounts that have been added into the online system.
  - FunReq 001 2: There are three categories of money transfer, (1) 1st party fund transfer: transferring within the same bank, different account but same owner name, (2) 3rd party fund transfer: transferring within the same bank and different owner account, and (3), other bank fund transfer: transferring over different bank account.
  - FunReq 001 3: List of accounts shall be displayed in alphabet order.

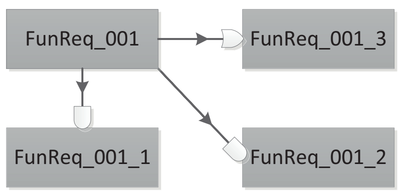

Fig. 3. An example of parent child relationships and semantics

The meaning of an example in Fig. 3 can be inferred as to justify a parent requirement (FunReq 001), it requires two children requirements of FunReq 001 1 and FunReq 001 2 to be justified, too whereas a parent requirement (FunReq 001) can be either fulfilled or not fulfilled by a child requirement (FunReq 001 3).

4.1 Icon-based Language Meta-model

The icon-based language meta-model (ILModelElement) demonstrated in Fig. 4 has been defined on the industry standard conventions (e.g. [23], [24]). In order to model general characteristics, the core meta-model includes entities such as Actor, Relationship, and Requirements. This meta-model can describe and communicate requirements as well as structure the reasoning about them. Actor definitions are frequently used to represent stakeholders or systems. Requirement elements are a linkable element to give an account of and the reasons for the proposed require goals and desires. A Requirement is characteristics of requirements artifacts that have a unique identifier (ID property), a name, a description, a priority (priority type), and a status (status type). A model may contain additional requirements (AdditonalReq) which is customizable. Requirement can have relation with each other by three dependency types: Refine, Require and Conflict. A number of changes is enclosed to a corresponding requirement to provide extra information about the requirements volatility.

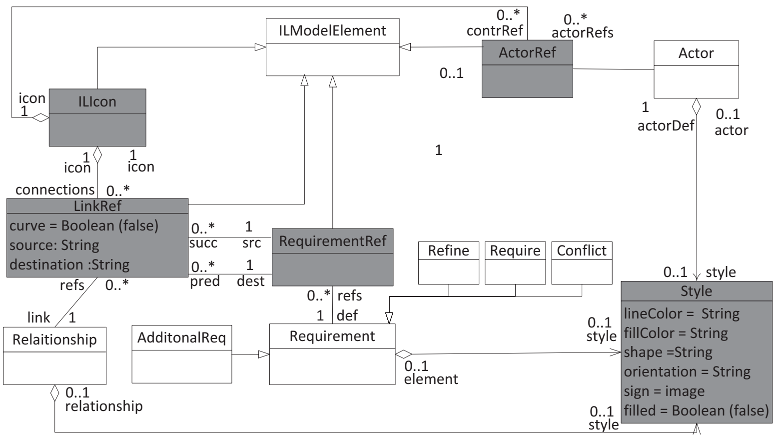

Fig. 4. Meta model for icon based language

Icon-based language-Icon (ILIcon) is a container for all actor reference, requirement reference, and link reference. An actor reference (ActorRef) refers to an actor definition and show its boundary. A link reference (LinkRef) is a direct link that bonds a source element to a different target element. A requirement reference (RequirementRef) shows a requirement element. Its representation associates with the type of the requirement element definition it refers to. The syntax such as color shape, orientation or symbol of an actor, relationship and requirement definition are defined in concrete style (Style) and are therefore shared by all the actors, requirements and relationship types. Ultimately, the concept of core meta-class of icon-based language is flexible for further customization.

5 Research Progress

As part of development phase of the research project, an initial attempt to produce an icon-based language, using RE contexts as a theme is skeletonized on the conclusion of interdisciplinary literature review (phase 1). Key attributes of icon-based language have been implied: - that RE world are identified, - that icon artifacts are designed to support the defined RE world and - that every construct in the language is represented visually by iconic symbols.

In the near future, the empirical evolution needs to be set up for testing the icons. It includes the formulation of questionnaire and test case scenarios. Different kinds of questions will be asked when testing icons: open ended (e.g. having icons available and ask what fitness describes the meaning of those icons?), image meaning (i.e. giving permission to the subject to match icons and meanings from two lists: one of icons and another one of meanings), and icon category (e.g. arranging a specific set of icons and asks the subject which category does those icons belong to?). Data is collected using a combination of methods. Key components are the user test and the user satisfaction questionnaire.

During empirical evaluation, we conduct two iterations: one with multicultural students in the RE course of the Department of Mathematical Information Technology at the University of Jyväskylä, and another with software companies both in Thailand and Finland. The best way to reach the heterogeneous participants is web-based icon test. By having the survey dispatched on the Internet, it can possibly grasp any person in any location that has access to the Internet. We will form an electronic survey and put it on the web. The result of these two iterations will be used to measure if the icon-based language is easy learnt and inform improvement possibility.

6 Conclusion

RE has been widely adapted by various communities while it remains a huge challenge in the context of interoperability between stakeholders. This research intends to propose an important framework encompassed with icon protocol towards breaking down communication obstacle between development teams and business stakeholders. We define a number of RE contexts that are needed to express requirements and to be represented by icons. The main expectation constitutes to human-oriented perspective that icon-based language is flexible and simple enough to allow stakeholders to apprehend. Icon-based language is designed rooted on the concept of visual notation and iconic communication that goes beyond the traditional natural languages. Throughout the research, it will contribute significantly to some possible ways for solving the problems caused by language and technical impediments when delivering requirements in software development life cycle. Furthermore, a simple restricted grammar and self-explanatory icons would make the icon-based language more appealing. The icons used for icon-based language should be commonly recognized across cultures. Otherwise, the icons might be designed for localization special icons for each of target cultures.

References

1. Pohl K.: Requirements Engineering Fundamentals Principles and Techniques. Springer, Berlin (2010)
2. Wiegers K E.: Software Requirements, Second Edition, Microsoft Press (2003)
3. Agarwal B. B., Tayal S. P., Gupta M.: Software Engineering & Testing (Computer sci ence series). In:Jones and Bartlett Publishers (2010)
4. Khanom S., Heimbürger A., Kärkkäinen T.: Icon based Language in Requirements Devel opment. In: Information Modelling and Knowledge Bases XXIII. IOS Press (2013)
5. Heimburger A, Kiyoki Y., Ylikotila T.: Communication Across Cultures in the Context of Multicultural, Software Development. In: Reports of MIT. Series C. (2011)
6. Heimburger A., Kiyoki Y., Kohtala S.: Intelligent Icons for Cross Cultural Knowledge Searching. 21st EJC on Information Modelling and Knowledge Bases, June (2011)
7. Tolar T. D., Lederberge A. R., Gokhale S., Tomasello M.: The Development of the Ability to Recognize the Meaning of Iconic Signs. Journal of Deaf Studies and Deaf Edu., (2008)
8. Choo K., Woo Y., Min H., Jo J.: Icon Language Based Auxiliary Communication System Interface for Language Disorders. Advances in Multimedia Information Systems (2005)
9. Moody D. L.: The "Physics" of Notations: Towards a Scientific Basis for Constructing Visual Notations in Software Engineering. IEEE Tran. on Software Engineering, vol. 35, no. 5, pp. 756 778, Dec (2009)
10. Cohn N.: The Visual Language Manifesto Restricting the "Comics" Industry and Its Ideol ogy. In:Second Edition, Emaki Productions (2007)
11. Fitrianie S., Datcu D., Rothkrantz L. J.M.: Human Communication Based on Icons in Cri sis Environments. Usability and International. Global and Local User Interface (2007)
12. Chang S. K., Polese G., Orefice S., Tucci M.: A Methodology and Interactive Environment for Icon based Language Design. In:Journal of Human Computer Studies (1994)
13. Moody D. L., Heymans P., Matulevicius R.: Visual syntax does matter: improving the cognitive effectiveness of the i* visual notation. Requirements Engineering, Springer Verlag, London Limited (2010)
14. Basu A., et al..: Vernacular Education and Communication Tool for the People with Mul tiple Disabilities. In:Development by Design Conference. Bangalore (2002)
15. Leemans N. E. M.: VIL: A Visual Inter Lingua. PhD thesis (2001)
16. Yang Turner F., Lau L.: Extending Use Case Diagrams to Support Requirements Discov ery. Workshop on RE for Systems, Services and Systems of Systems (RESS) (2011)
17. Cardei I., Fonoage M., Shankar R.: Model Based Requirements Specification and Valida tion for Component Architectures. 2nd Annual IEEE Systems Conference (2008)
18. Helming J., et al.: Towards a Unified Requirements Modeling Language. In:IEEE 5th In ternational Workshop on Requirements Engineering Visualization (REV) (2010)
19. Hansen S., and Lyytinen, K. L.: Challenges in Contemporary Requirement Practice, IEEE Proceedings of the 43rd Hawaii International Conference on System Sciences, 2010.
20. Mathiassen L., Tuunanen T., Saarinen T. and Rossi M.: A Contigency Model for Require ments Development, JAIS, Vol. 8, pp. 569 597, 2007.
21. Galitz W. O.: The Essential Guide to User Interface Design: An Introduction to GUI De sign Principles and Techniques. In:Third Edition, Wiley Publishing, Inc. (2007)
22. Harel D., Rumpe B.: Meaningful Modeling: What's the Semantics of "Semantics"?. In:IEEE Computer Society, pp. 64 72 (2004)
23. ISO/IEC, ISO/lEC Standard 24744.: Software Engineering Metamodel for Development Methodologies. In: ISO/IEC (2007)
24. ITU T.: User Requirements Notation (URN). Z.151, Telecommunication Sector (2008)

III

ICON REPRESENTATIONS IN SUPPORTING REQUIREMENTS ELICITATION PROCESS

by

Sukanya Khanom, Anneli Heimbürger & Tommi Kärkkäinen, 2014

In T. Tokuda, Y. Kiyoki, H. Jaakkola & N. Yoshida (Eds.) Information Modelling and Knowledge Bases XXV. IOS Press, Amsterdam, 133-145.

IV

CROSS-CULTURAL COMMUNICATION WITH ICONS AND IMAGES

by

Anneli Heimbürger, Marie Duzi, Yasushi Kiyoki, Shiori Sasaki & Sukanya Khanom, 2014

In T. Tokuda, Y. Kiyoki, H. Jaakkola & N. Yoshida (Eds.) Information Modelling and Knowledge Bases XXV. ISO Press, Amsterdam, 306-322.

V

ICON-BASED LANGUAGE: AUXILIARY COMMUNICATION FOR REQUIREMENTS ENGINEERING

by

Sukanya Khanom, Anneli Heimbürger & Tommi Kärkkäinen, 2013

International Journal of Engineering Science and Technology 5, 1076-1082.

ICON-BASED LANGAUGE: AUXILIARY COMMUNICATION FOR REQUIREMENTS ENGINEERING

SUKANYA KHANOM, ANNELI HEIMBÜRGER AND TOMMI KÄRKKÄINEN
Department of Mathematical Information Technology, University of Jyväskylä, Finland
sukanya.s.khanom@student.jyu.fi , anneli.a.heimburger@jyu.fi and tommi.karkkainen@jyu.fi

Abstract :

In requirements engineering, numerous modelling approaches have been addressed to support communication and shared understanding, for instance, UML (use case diagram) and GORE (goal graph). Most of them use abstract geometrical shapes as fundamental elements. The RE stakeholders, therefore, need to master the prerequisite skill to understand this complex set of abstraction. Recently, there has been an increasing need for a simple and an easy-to-learn modelling technique. We propose an icon-based language as an alternative to RE modelling. Unlike other RE notations, icon-based language is devised principally from a set of icons. Our approach can be derived by identifying RE world, defining icon world and combining these two worlds. This development yields the demand to have a modelling language that unifies requirements activities. The result can be used to overcome the communication barriers between technical and non-technical stakeholders, and to have icon attributes that can be attached to individual requirement.

Keywords*:* Icon-based language; Visual construction; Requirements engineering; Stakeholder.

1. Introduction

Requirements engineering (RE) is one of the most vital factors for project success in software-intensive systems [Pohl (2010)]. The empirical evidences [Emam and Koru (2008); Cerpa and Verner (2009)] reveal that inappropriate and inadequate RE leads to scope and requirement changes in a project which finally influences the project failure syndrome. Many RE visual modelling languages have been developed to communicate, elicit, analyze and deliberate the stakeholders' needs [Agarwal *et al.* (2010)]. For example, UML and goal-oriented models are two outstanding visual techniques. The UML has been widely accepted as the industry standard language for modelling and analyzing the requirements [Morris and Spanoudakis (2001)]. The recent emergence of a goal-oriented model has been employed for formulating different levels of abstraction from high-level goals to the lower level of operational requirements [Moody *et al*. (2010); Lamsweerde (2000)]. Their distinctive function of using graphical elements is claimed as a major advantage. However, both methods use abstract graphical notations and syntax, whose meaning is difficult for novices to understand. Consequently, notational characteristics in existing visual languages have led to misinterpretation and confusion [Morris and Spanoudakis (2001)]. The challenges with existing visual modelling language have motivated the current research, which aims to conduct uncomplicated visual constructions based primarily on iconic representations. Icons have been recognized as the vehicle to transfer information and easy recognition for any context by any person. Presently, icons exist in almost every computer graphic user interface. Their information encryption helps the reviewers to perceive the intended meaning of particular situation in a user friendly manner. The difference between icons and other visual modalities such as graph and diagram lies on intuitive and recognition-based information. Typical examples are icons used in many public information spaces, in trains, traffic signs, aircrafts, and cars [Khanom *et al*. (2012); Heimbürger *et al*. (2011a); Heimbürger *et al*. (2011b)]. Several benefits of icons have been asserted, but surprisingly, there has been little discussion about applicable icons in RE research.

We propose an icon-based language as an alternative communication avenue for multifaceted RE stakeholders. With designing an icon-based language, the focus of the study is dedicated to the construction of feasible visual notations to represent RE context. The visual notation is modelled from the basic elements of visual vocabularies, their syntactic rules, and a set of semantics [Harel and Rumpe (2004)]. The visual vocabulary is the construction of the group of icons. Visual semantics defines the meanings of each icon by mapping it to the construct it represents. Visual syntax generates the relationship between visual vocabulary and visual semantic of the notation. Each construction is originated to represent a concept, an object, an action, or a relation. The apparent potential of the icon-based language would make it an attracting option to bridge the communication gaps among multi-stakeholder. The contributions of this development are expected to allow both experts and non-experts in different nationalities to deliver requirements and attributes that can be ascribe an individual requirement such as status, priority and number of change.

The paper is organized as follows. In the following section, we first give a brief overview of icon communication in relevant fields and visual languages in RE. We then describe our methodology and our approach to icon-based language. In Section 4, we developed two artifacts that demonstrate the approach. Final section concludes the paper and presents future work.

2. State-of-the-Art

Recent studies by [Moody (2009); Moody *et al.*(2010)] indicate that icons and visuals represent important benefits for communication research. Research efforts have been done in developing computer-intensive iconic communication systems, which primary aimed at fostering people to communicate with each other. For instance, Choo *et al*.(2005); Basu *et al.*(2002) have dedicated the icon-based systems as a communication tool for the impaired people. Most of these proposals are complex to comprehend and rudimentary on linguistic theories that the units of meaning correspond to the grammatical rules of clause and words (e.g. [Fitrianie *et al*.(2007); Leemans (2001)]). A detailed research has been emphasized on designing the system that facilitates the reviewers to mutually communicate without sharing common language [Leemans (2001)]. In the crisis environment, Fitrianie *et al*.(2007) has announced a comprehensive icon-based interface containing graphic symbols to represent concepts or ideas.

In the area of RE, A wide range of researches has highlighted on the use of diagrams for improving requirements engineering activities. Extended features of use case diagram have been devised by Yang-Turner and Lau (2011) to support the tasks of stakeholders in elicitation activity. In similar manner, Helming *et al.* approached an incremental UML as a communication means for delivering collaborative environment. The contemporary approach of goal-oriented model [Moody *et al*.(2010)] has enormously driven the importance of the visual notation in the RE field. Other visualization techniques adapted and researched in identifying and modularizing requirements are, for instance, the Aspect-Oriented Requirements Engineering (AORE) [Oliverira *et al*.(2010)] and physualization material such as stickers, markers and sketchpads [Callele (2010)]. Scenario-based approach has also been accepted successfully as a prominent tailor to bridge the communication impediments among multifaceted stakeholders [Sutcliffe *et al*. (2011)].

3. Research Approach

The current research is carried out using a design science research approach. Havner *et al.* (2004) designates design science research as a building and evaluating process with the purpose to conduct a set of artifacts. Our main goal is to define and develop icon artifacts that support RE context. Since icon-based language is a novel approach, its design can involve an iterative evaluation and refinement of artifacts. The research approach we employ follows Peffers *et al.* (2007) (see Figure 1).

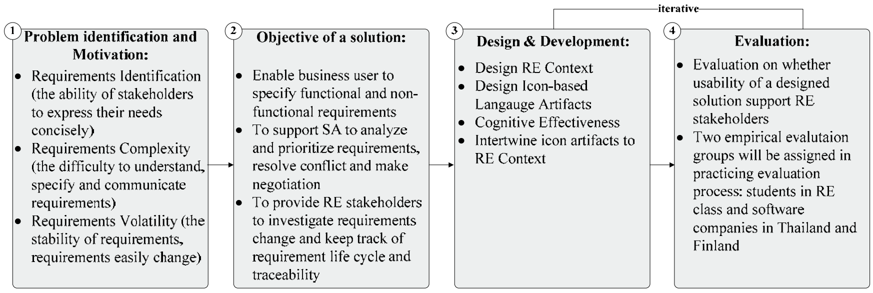

Fig. 1. Design science research methodology.

We first identified the problems of RE according to literature review, e.g. [Hansen and Lyytinen (2010); Mathiassen *et al.* (2007); Kaiya *et al.* (2005)]. We arrived at three difficulties stakeholders encountered in performing RE. Firstly, there is the ability challenge of system stakeholders to express their needs concisely and concretely. In broad spectrum, requirements are heavily hard to discover in situations where there is a communication gap between technical and non-technical users that appear to speak different languages and apply different approaches for desired outcomes. Secondly, requirements complexity happens when stakeholders experience the difficulty to understand, specify and communicate requirements. Finally, requirements volatility signifies to the stability of requirements that easily change as a result of environmental dynamic or individual learning.

We then defined concrete objectives to inform the requisites of a possible solution to the abovementioned problems. Our first objective is to find a solution that enables RE stakeholders to deal with quality requirements. The large variety of stakeholders' backgrounds makes it necessary to find ways for an easy adaptation of a supportive method. A second objective is to reduce the requirements analysis effort. Icon-based language that allows flexible composition of various attributes and relationships increases the number of model elements to support stakeholders to analyze and prioritize requirements, as well as resolve conflict and make negotiation. In addition to that, icon-based language could provide RE stakeholders to investigate requirements change and keep tracking requirement life cycle and traceability.

At the design stage (see also Figure 1), the key insights of our approach in the design solution can be obtained by defining RE world, defining icon world and integrating those two worlds together. We infer the necessities for our artifacts by drawing on theoretical foundations in the interdisciplinary fields of RE, human-computer interaction, and cognitive phycology. We further combine knowledge and techniques from the research fields of modelling languages and iconic communication in order to make design decisions that principally effect the direction of our approach.

In the evaluation phase, we conduct two iterative evaluations: one with inexperienced users and another one with expert users. The first iteration will be tested by multicultural students in the RE course of the Department of Mathematical Information Technology at the University of Jyväskylä. For the latter iteration, intercultural software companies both in Thailand and Finland are the key players. The primary goal is to evaluate whether icon-based language is applicable in practice and whether it can be used in real software project. We take advantage of usability testing to evaluate if the utility of a defined icon-based language model supports the tasks of RE stakeholders. The results of these two iterations are used to inform improvement possibilities.

4. Proposed Solution

In this section, we will describe how our objectives for supporting RE stakeholders inform the development of various artifacts that provide consensus requirements types, attributes and relationships. These artifacts include a RE world definition, icon world definition and integration of those two worlds.

4.1. *Artifact 1: The definition of RE world*

Cooperating with this question what are the tasks of RE that can be supported by icons?, we have established a list of boundaries that stimulated by RE exercise in multicultural backgrounds. We focus on extracting those potential activities that impact entire RE process by conducting a thorough literature review on related work. As a given example shown in Figure 2, the central concept of requirement artifact begins with patterning the scope, associated actors, attributes and scenarios.

Primary Actor: Business Team, System Analyst

Goal and context: Requirements can contain Business requirement, Functional, Non-functional requirements or Constraint. Every requirement request has recorded the requester, status, and priority so that requirement can be traced backward throughout the entire life cycle of the project.

Scope: Requirements, attributes, actors, relationships

Main success scenario:

1. Business Team: initial a requirement request (Propose) with priority type
2. System Analyst: analyze a requirement for its overall impact:
   a. The key stakeholders agree to incorporate the requirement (Accept)
   b. If the requirement is proposed but it is not planned for implementation (Reject)
3. Developer: design, write and test the source code that implements the requirement (Implement)
4. Business Team, System Analyst: verify the correct functionality of the implemented requirement (Verify)
5. Check interrelation between two or more different requirements:
   a. If R1 is justified only when R2 is justified. R2 can be acted as a pre-condition of R1 (Require)
   b. If R1 is derived from R2 by intensifying more detail on it (Refine)
   c. If the negation of R1 can be inferred from R2 or the removing of R1 no longer results in a logical inconsistency in R2 and vice versa (Conflict)
6. Check the relationship of parent and child requirements:
   a. To fulfill R1, R1a has to be fulfilled as well (AND)
   b. To fulfill R1, R1a could be fulfilled: it is optional (OR)

Fig. 2. The use case scenario of requirement characteristic.

Firstly, to discover the purpose of the system under development in elicitation phase, different catalogs of requirements are distinguished on the basis of quality characteristics in [ISO/IEC 9216]. Secondly, to monitor the lifespan of requirement, status and priority attributes granted to every requirement are significantly helpful. It embraces the identification of actor who is authorized to change a status, and update status only when the conditions are satisfied. Furthermore, to accelerate the requirements management, the number of requirements changes is systematically retained in the project. Finally, to support requirements traceability, the association among requirements facilitates the ability to describe and follow the life of requirements [Gotel and Finkelstein (1994)].

In the next step, we model the sets of concepts of RE that are able to be expressed by icon-based language in structure of meta-modelling. The meta-model given in Figure 3 is the example of requirements artifact in Deliverable type. In order to systematize the requirements granularity, we identify generic attribute and activities of requirements [Pohl (2010), Wiegers(2003)]. Two types of model can be defined accordingly to the purpose of icon-based language that will be using icons to represent either what are already existed in the RE research and what are newly proposed in this research work. We name those two types as Deliverable and Classification. However, any other exclusive types can be extended in the customization. In the one hand, Deliverable typifies the scenario of scope, actor and goal that needs interrelation between each activity. Typically, this dynamically describes the requirements' behavior under various conditions. In the other hand, Classification is a kind of RE activities such as requirements taxonomy, elicitation tasks, or analysis tasks that we categorize to be individually and statistically exemplified by icon(s). Requirements captured in DeliverableModel contain a set of relevant attributes, actors and relationships.

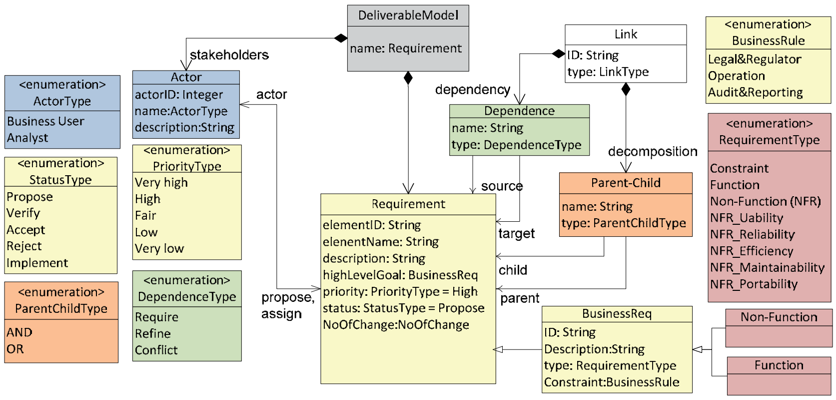

Fig. 3. The meta-model representing concept of requirement artifact, its attributes, relevant actors and relationships.

A Requirement is characteristics of requirements artifacts that have a unique identifier (ID property), a name, a description, a highLevelGoal (BusinessReq), a priority (taking values from PriorityType), a status (determining the requirements life cycle values from StatusType), and a number of changes (NoOfChange). The model may contain three different kinds of requirements: business, functional and non-functional requirements. Business requirements are high-level requirements that reflect a goal or vision of the organization that the system must accomplish. Under business requirements, it may contain functional requirements that present a behavior of a system under specific condition. Otherwise, it enables to contain non-functional requirements that represent a quality attribute in which the system must have. A requirement may be attached with business rules, law, policy, or procedure which constrains the certain degree of freedom in delivering a solution. BusinessReq contained by a Requirement class relies on a unique identifier (ID), a description, type (selecting one or more of the values in a series of RequirementType), and constraint (grasping the value from BusinessRule). Number of change is the level of the requirements that is happened to change over the software life cycle. It is important to keep information about individual actor (actorID, name, and description) for further inquiry. The requirement can be associated to each other through link types: Dependency and Parent-Child. Three relationships of Dependency, require, refine and conflict, have been delineated to qualify the association between two or more different requirements. Moreover, one requirement can be divided into sub-requirements and those sub-requirements are connected to their parent with Parent-Child link. The requirements engineer can use two types of decomposition corresponding to logical combination – one is AND-Parent-Child and the other is OR-Parent-Child. In AND relationship, unless all of sub-requirements are satisfied, their parent requirement cannot be satisfied. On the contrary, with OR relationship, a parent requirement could be satisfied when at least one sub-requirement is satisfied.

4.2. *Artifact 2: The definition of icon world*

In Figure 4, the entire stages in icon-based language design are iterative [Costagliola *et al*. (2004)] which means that if tests expose usage drawbacks, we might decide to review the Icon Library, Grammatical Rule and Semantics Library, as well as, to replicate the usability testing in the next version. Our limitation is the fact that all visual vocabularies in this paper are only used to represent the concepts and ideas. They are not fitness design for what they are representing. The design perspective must be done by the designers who are expert in the area and it relies enormously on cultural experience and cognitive effectiveness.

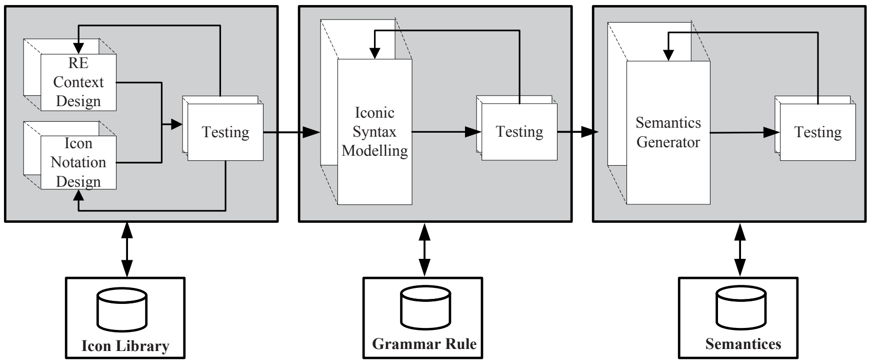

Fig. 4. The process for modelling the icon-based language.

In the context of icon world, first of all we need to derive the icon library of a visual notation being designed. After analyzing and defining the RE context, icons are necessarily produced to simplify that context. For each icon notation that has to be conducted, we must generate a final Icon Library. The Icon Library contains the series of iconic symbols and visual sentences symbolizing the icon notation. When designing iconic symbols, guidelines and standards such as [ETSI EG: 202-048, ISO/IEC 11581] can guide us to gain applicable design for particular purpose. Since the interpretation of icons is a vulnerably subjective matter, the icons should be properly selected, developed and evaluated. Therefore, we will apply ETSI EG 201-379 framework that its direction would ultimately relief such challenge. To solve the problems of misinterpretation and cultural prejudice, the test participants will be chosen to include different nationalities. An example of icon representations depicted in Figure 5 describes requirements attributes in which priority types utilize different vehicles with different speed characteristics to express how urgent of the requirements is, whereas the action icons are representative of status states. Additionally, logical signs, “AND” and “OR” gates represent the Parent-Child relationship.

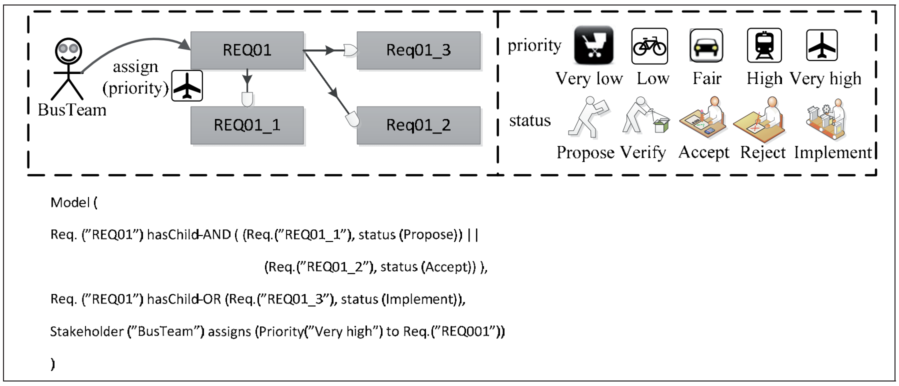

Fig. 5. An example of icon vocabulary and semantics.

During the Iconic Syntax Modelling, we refine the specification of the iconic symbols according to the attribute-based representation approach that must be conformed to the criteria for proper visual syntax. Grounded on the attribute-based tactic, the grammar of icon-based language can be qualified depending on the structure of its iconic objects on the way they can be composed in order to form visual sentence. The criteria of good visual syntax are based on cognitive effective phycology [Moody (2009)]. He provides nine principles of a perspective theory that can be employed throughout the development of notation. The principle of Semiotic Clarity states that there should be one-to-one correspondence between syntactic and semantic constructs. The principle of Perceptual Discriminability affirms that different graphical symbols should be noticeably discriminate from each other. The principle of Visual Expressiveness asserts that the utilization of full range of variables such as color should be accommodating to represent the notation elements. The principle of Dual

Coding presumes that the employment of textual coding have a duty to augment graphic forms. The principle of Semantic Transparency supposes that the exploitation of visual representations whose appearance advocates their meaning reduces the memory load. The principle of Cognitive Integration is taken into consideration when multiple diagrams are used to represent the complete system. The principle of Graphical Economy declares that a limit number of different graphical symbols should be cognitively manageable. The principle of Complexity Management insinuates to the competence of a visual notation that can represent information without over memory load. The principle of Cognitive fit, finally, is mentioned that different visual dialects suitably support different tasks and users.

Accordingly, a language comprises of syntactic notations so called syntax and the meaning (semantics) of those elements. Thus, in Semantics Generator stage, the meaningful language elements (semantics) are mapped from the syntax constructs to the semantic domain [Harel and Rumpe (2004)]. For instance, reverted to Figure 5, we take for granted that different people might be able to view semantics of icon representations for priority types form their velocity as baby carriage for "very low", bicycle for "low", car for "fair", train for "high" and aircraft for "very high". The semantics of the iconic sentence can be translated that a function requirement (REQ01) can be decomposed into three children, REQ01_1, REQ01_2 and REQ01_3 respectively which means that a parent requirement can be accomplished only when REQ01_1 and REQ01_2 are both accomplished. Plus, stakeholder (BusTeam) assigns "very high" priority to a parent requirement. We must analyze the semantics aspects of the icon-based language model notation and designate extra semantics attributes on the whole visual symbols.

5. Conclusion

This paper has given an account of the importance and problems of RE in software development, and the gaps of the available modelling techniques. The purpose of the study was to initial the icon notations used to communicate requirements activities among multifaceted stakeholders in RE. The icon-based language model relies on visual symbols, syntax properties, and semantics. The model provides a competent way to specify, trace and verify requirements life cycle which requires a small amount of prior knowledge. The model can be used for demonstrating the visual notations of requirements activities along the software development life cycle. It would give ways to arrive at the definition of requirements with unambiguity, completeness and consistency. Besides, we aim that icon-based language could elevate stakeholders to deal with essential requirements attributes such as status, priority and number of change. Under this scenario, the expectation of the research is pointed to low technology-oriented audiences. Nevertheless, development teams who are experts and have some methodological experience can also use the model.

In the near future, we need to establish first iterative empirical evaluation for testing the proposed concept of icon-based language by students. It will continue together with the formulation of questionnaire and test scenarios. Different kinds of scenarios will be asked when testing icons (e.g. icon meaning and icon category). Data is collected using a combination of methods: the user test and the user satisfaction questionnaire. The appropriated way to reach the heterogeneous participants is web-based icon test. By having the electronic survey dispatched on the Internet, it can possibly grasp information from any person in any location that has access to the Internet. Currently, the web-based survey is developing to be ready for the first participant group.

References

[1] Agarwal, et al. (2010).*Software Engineering & Testing (Computer science series)*, Jones and Bartlett Publishers.

[2] Basu, et al. (2002): Vernacular Education and Communication Tool for the People with Multiple Disabilities, Development by Design Conference, Bangalore.

[3] Callele, D. (2010): Physualization: Going Beyond Paper Prototype, Fifth International Workshop on Requirements Engineering Visualization, pp. 35-39.

[4] Cerpa, N.; Verner, J. M. (2009): Why did your project fail?, Communication of the ACM, vol, 52-12, pp. 130-134.

[5] Choo, et al (2005): Icon Language-Based Auxiliary Communication System Interface for Language Disorders, Advances in Multimedia Information Systems.

[6] Costagliola, el al. (2004): A Framework for Modeling and Implementing Visual Notations With Applications to Software Engineering, ACM Transactions on Software Engineering and Methodology, vol. 13-4, pp. 431–487.

[7] Emam, K. E.; Koru, A. G. (2008): A Replicated Survey of IT Software Project Failures, IEEE Software, vol. 25, pp. 84-90.

[8] ETSI EG 201-379 (1998-09): Human Factors (HF); Framework for the development, evaluation and selection of graphical symbols.

[9] ETSI EG 202-048 (2002-08): Human Factors (HF); Guideline on the multimodality of icons, symbols, and pictograms.

[10] Fitrianie, et al.(2007): Human Communication Based on Icons in Crisis Environments, Usability and International, 2007.

[11] Gotel, O. C. Z.; Finkelstein A. C. W. (1994): An Analysis of the Requirements Traceability Problem, IEEE RE, pp. 94-101.

[12] Hansen, S.; Lyytinen K. L (2010): Challenges in Contemporary Requirement Practice, IEEE Proceedings of the 43rd Hawaii International Conference on System Sciences, pp. 1-11.

[13] Harel, D.; Rumpe, B. (2004): Meaningful Modeling: What's the Semantics of "Semantics"?, IEEE Computer Society, pp. 64-72.

[14] Helming, et al. (2010): Towards a Unified Requirements Modeling Language, IEEE 5th International Workshop on REV, pp. 53-57.

[15] Heimbürger, et al. (2011a): Communication Across Cultures in the Context of Multicultural, Software Development Reports of the Department of MIT.

[16] Heimbürger, et al. (2011b): Intelligent Icons for Cross-Cultural Knowledge Searching, 21st EJC on Information Modelling and Knowledge Bases, Tallinn.

[17] Hevner, et al. (2004): Design Science in Information Systems Research, MIS Quarterly (28:1), pp. 74-105.

[18] ISO/IEC 9126 (1991) : IT- Software Product Evaluation – Quality characteristics," International Organization for Standardization.

[19] ISO/IEC 11581(2000): Information technology -- User System Interfaces -- Icon symbols and functions -- Part 1: General Icons, Part 2: Object Icons, Part 6: Action Icons.
[20] Kaiya, et al. (2005): Improving the Detecting of Requirements Discordances Among Stakeholders, Requirements Engineering Journal, pp. 289-203.
[21] Khanom, et al. (2012): Icon-based Language in Requirements Development, 22nd EJC on Information Modeling and Knowledge Bases.
[22] Lamsweerde, A. V. (2000): Requirements Engineering in the Year 00: A Research Perspective, 22nd Int. Conference on Software Engineering, pp. 5-19.
[23] Leemans, N. E. M. (2001): VIL: A Visual Inter Lingua, PhD thesis.
[24] Mathiassen, et al. (2007): A Contigency Model for Requirements Development, Journal of the Association for Information System (JAIS), Vol. 8, pp. 569-597.
[25] Morris, S.; Spanoudakis, G. (2001): UML: An Evaluation of the Visual Syntax of the Language, 34th Hawaii Int. Conference on System Science, pp. 1-10.
[26] Moody, D. L. (2009): The "Physics" of Notations: Towards a Scientific Basis for Constructing Visual Notations in Software Engineering, IEEE Tran. on Software Engineering, vol. 35, pp. 756-778.
[27] Moody, el al. (2010): Visual syntax does matter: improving the cognitive effectiveness of the i* visual notation, Springer RE-Journal, pp. 141-175.
[28] Oliverira, et al. (2010): The VisualAORE DSL, 5th International Workshop on Requirements Engineering Visualization, pp. 11-19.
[29] Peffers, et al. (2007): A Design Science Research Methodology for Information Systems Research, Journal of Management Information Systems (24:3), pp. 45-77.
[30] Pohl, K. (2010). *Requirements engineering fundamentals Principles and Techniques*, Springer, Berlin.
[31] Sutcliffe, et al. (2011): Experience with user-centred requirements engineering, Requirements Engineering Journal, pp. 267-280.
[32] Wiegers, K. E (2003). *Software Requirements*, Second Edition, Microsoft Press.
[33] Yang-Turner, F.; Lau, L. (2011): Extending Use Case Diagrams to Support Requirements Discovery, Workshop on RE for Systems (RESS), pp. 32-35.

VI

ICONS: VISUAL REPRESENTATION TO ENRICH REQUIREMENTS ENGINEERING WORK

by

Sukanya Khanom, Anneli Heimbürger & Tommi Kärkkäinen, 2013

Journal of Software Engineering and Applications 6 (11), 610-622.

Journal of Software Engineering and Applications, 2013, 6, 610-622
Published Online November 2013 (http://www.scirp.org/journal/jsea)
http://dx.doi.org/10.4236/jsea.2013.611073

Scientific Research

Icons: Visual Representation to Enrich Requirements Engineering Work

Sukanya Khanom, Anneli Heimbürger, Tommi Kärkkäinen

Department of Mathematical Information Technology, University of Jyväskylä, Jyväskylä, Finland.
Email: sukanya.s.khanom@student.jyu.fi, anneli.a.heimburger@jyu.fi, tommi.karkkainen@jyu.fi

Received October 16th, 2013; revised November 10th, 2013; accepted November 17th, 2013

ABSTRACT

Adapting icons in requirements engineering can support the multifaceted needs of stakeholders. Conventional approaches to RE are mainly highlighted in diagrams. This paper introduces icon-based information as a way to represent ideas and concepts in the requirements engineering domain. We report on icon artifacts that support requirements engineering work such as priority types, status states and stakeholder kinds. We evaluate how users interpret meanings of icons and the efficacy of icon prototypes shaped to represent those requirements attributes. Our hypothesis is whether practitioners can recognize the icons' meaning in terms of their functional representation. According to the empirical data from 45 participants, the findings demonstrate the probability of providing users with icons and their intended functions that correspond to RE artifacts in a novel yet effective manner. Based on these findings, we suggest that icons could enrich stakeholders' perception of the RE process as a whole; however, meaningful interpretation of an icon is subject to the user's prior knowledge and experience.

Keywords: Requirements Engineering; Icon; Culture; Stakeholder; Visual Language

1. Introduction

The growth in sophistication of human-computer interaction (HCI) can be seen in graphical user interfaces (GUIs) [1,2]. These interfaces have significantly reduced the amount of typing needed when using a computer. Nowadays, icons are important visible representations of information. They range from device control and icons in public to iconic communication systems assisting some particular areas [3-7]. The use of icons to reflect objects or actions is not new in modern computer-aided systems or software packages. The icons era in interface and screen design began with the Xerox Star computer in the 1970s and thoroughly bloomed with the onset of Apple Macintosh in the mid-1980s. GUI rapidly turned into the paramount user interface when Microsoft adopted variation of icons with its Windows system [1,2,8].

Because of the increasing presence of icons within computer-intensive communication, it is necessary to consider how icons are interpreted and the factors that influence their effectiveness. In addition, because icons are frequently used to supplement texts and overcome language barriers, this makes the ability to recognize and comprehend icons even more complex. In particular, in requirements engineering (RE) the adaptation of appropriate icons can be difficult. The RE process typically involves collaborations of people with different backgrounds, roles and responsibilities, so different that they appear to speak different languages and apply different approaches for the desired outcomes [9-11]. Recently, dozens of requirements methodologies and techniques have been implemented and made available for practitioners. However, the empirical evidence repeatedly reveals that RE is considered as one of the key contributors to the failure of software [12,13]. This position creates a challenge for researchers to find an uncomplicated and easy-to-learn method that can enhance requirements activities, reduce ambiguity and promote collaboration. Even though icons have been accepted successfully in HCI, information on the utilization of icons in RE is scarce (e.g. [14]).

Our attempt is to refine icon-based information to support the tasks of RE stakeholders and to clarify how readers recognize connotative meanings of icons, that is, what the icon is intended to represent. There may be numerous approaches to solve the problems encountered in

RE. This paper introduces one such approach with the help of the following key concepts: cultural aspects, RE artifacts and icon-based information. The cultural aspects assist in building a cultural user framework with culturally adaptive user interfaces that adapt themselves to the user's background. RE artifacts provide a skeleton for the scenarios of requirements activities that can be characterized by icons, such as attributes, stakeholders and relationships. Attributes are the properties that distinguish one requirement from another, and they establish a context and background for each requirement. Stakeholders are the persons or systems that have the purpose of achieving goals and that take action to achieve them. Relationships signify the correlation between two or more requirements. All RE artifacts are patterned as static elements. This type of patterning means that all users visualize requirements features or functions in the same manner. Icon-based information is designed in visually supporting pieces of RE artifacts that go beyond an ordinary textual description. Unlike RE artifacts that are stationary, icon-based information is dynamic depending on the users' cultural background. The deviation of icon presentations specific to cultural preferences can be seen for example, in how the attached icon for high priority is illuminated. Users from Europe might experience the interface with purple-colored icons, but users in Asia may contemplate the interface with yellow-colored icons [15].

To the best of our knowledge, icon-based information is the first approach that tries to represent icons in RE and that is able to adapt its interface to the preferences according to the user's cultural background. Our research question explores how well practitioners can predict the meaning of icon representations. To answer this question, this paper evaluates icons that represent priority types, status states, stakeholder kinds and relationships. In our study, icon-based information was tested with 45 participants from Finland. Our findings have the potential to inform the development of unambiguous requirements and avoid the aforementioned problems. Consequently, our contributions are as follows: First, we present a theoretically grounded approach for icon-based information in RE and adapting its interface by cultural background. We propose a cultural user framework to approximate a person's cultural preference and intertwine RE artifacts with their prospective iconic representations. Second, we empirically evaluate those icons' meaning and demonstrate whether icons are able to characterize RE artifacts. We expect that the role of iconic information used in RE could facilitate the work of business users and software developers in all phases, from elicitation and negotiation to validation.

In the following section, we introduce the previous work on which we have based our method for designing a concept of icon-based information. In Section 3, we describe our research approach. We develop three artifacts that demonstrate the approach and in Section 4, we describe its stepwise nature. In Section 5, we describe an empirical evaluation of icon-based information. Next, we discuss our results and recommendations for improvement. Last, we propose future work and present our conclusions.

2. Related Work

We review the relevant work on three questions: What prominent methods are regularly employed in RE? What icon applications already exist? How can we understand how cultural aspects affect icon perception?

2.1. The Nature of Iconic Communication in RE and Other Environments

The de facto standard of visualization techniques that have broadly converged in RE are diagram and graphical, such as Unified Modeling Language (UML) and goal-oriented models. UML is one of the conceptual modeling tools in software engineering to represent static and dynamic phenomena of user requirements [16,17]. The goal-oriented model is a paradigm for eliciting, evaluating, elaborating, documenting and analyzing software requirements [18,19]. Nevertheless, the empirical evidence (e.g. [14,17,20]) has revealed some shortcomings of these techniques, such as the problematic use of abstract shapes that have articulately conventional meanings which must be learnt. Iconic visualization is one modality recommended to be employed in enhancing cognitive effectiveness for RE notation [14]. However, the applicable icons in RE can be considered because they are pervasively used in toolbars and menu bars, but not in the requirements engineering context itself.

Over the decades, icons have been at the center of human-computer interaction (HCI). While concrete icons are believed to be effective in graphical user interfaces, abstract icons are judged to be less effective because they do not represent real-world objects [21]. Interface designers repeatedly use concrete icons because of the strength of their relation between icon and function. Notwithstanding, abstract icons can also be utilized in the interface as they strengthen icon-referent relations by producing a less pictorial representation that is meaningful to the user [21,22]. Within the domain of HCI, the methods of cognitive psychology are commonly followed (e.g. [14,23,24]). The cognitive factors of icons embody the cognition of visual information and the association of connection. The effects on user cognition are based on a range of characteristics such as color, shape and size [25]. By improving the usability of icons, the hope is to improve the interactive interface between people and machines.

In crisis situations, icons have been exploited in facilitating auxiliary communication among multifaceted users [26,27]. Icon-based communication interfaces have been developed to represent concepts and ideas in crisis environments by providing icons such as explosion, victim, and ambulance that for a crisis observation interface. Those icons are created to conform to ontology-based knowledge, W3C-OWL [28], by defining a case for each icon; the icon victim contains number, location, and status, for instance. This crisis observation interface provides iconic symbols, geometrical features or icon strings. Geometrical shapes, such as arrows, lines, ellipses, and rectangles, can be used to indicate a distinctive area, an object, an event, or a location.

2.2. Cultural Influences on Iconic Perception

Culture has a crucial role in the use of information and communication technology. Information system research has long admitted that cultural difference can inhibit the successful use of information technology [15,29-31]. The major finding from the existing literature on cultural differences and icon recognition is that there are differences of modality that groups of users have regarding what icons are. Cultures have different degrees of contexts: some cultures are determined to be high context whereas others are considered to be low context. Context refers to the amount of information given in communication. In high-context communication, most of the meaning is in the context. By contrast, in low-context communication, most of the meaning is in the transmitted message. Problems and conflicts emerge when people from high- and low-context cultures communicate with each other [32]. The differences have mostly been considered on national or organization levels. We distinguished the characteristics between countries based on cultural theories (e.g. [15,29,33]). Hofstede has distinguished five dimensions of power distance (PDI), individualism (IDV), masculinity (MAS), uncertainty avoidance (UAI), and long-term orientation (LTO). Power distance, for example, describes the extent to which the hierarchies exist and are accepted by the members in a society. Within countries (e.g. Thailand) that have been assigned a high power distance score, inequalities are believed to be much more acceptable in society than in low power distance countries (e.g. Finland and Australia). The people in highly individualist countries (e.g. Finland and Australia) are usually seen as more independent from a group. In contrast, people in collectivist countries (e.g. Thailand) often see themselves as part of a group. The third dimension, masculinity, refers to a high preference for competitive achievement (high masculinity) versus low preference (femininity). The degree to which the members of society tolerate uncertainty and ambiguity is inversely reflected by UAI, that is, people from high uncertainty avoidance countries prefer less ambiguity than those in low uncertainty avoidance countries. The fifth dimension, long-term orientation, measures how people perceive time. In LTO countries, people are comfortable with sacrificing for long-term benefit, but in countries with short-term orientation people are more focused on immediate results. In **Table 1**, we have summarized the rules for a cultural interface based on Hofstede's theory and human-computer interaction components such as color, appearance and contents [15,29,33-35]. The table lists the influences as high or low scores.

3. Research Approach

We employed the research methodology of design science [36,37] to construct icon-based information in RE context (see **Figure 1**). We operationalized our three research problems (see number 1 in **Figure 1**). First, requirements identification means the challenges of the capability of a system's stakeholders to express their needs concisely and concretely. In other words, we can say that requirements are difficult to capture in the situation where there is a communication gap in RE between business teams and development stakeholders. Secondly, requirements complexity refers to the difficulty of understanding, communicating and reviewing the requirements. Thirdly, requirements volatility refers to the stability of requirements, which are easily changed as a result of environmental dynamic or individual learning.

We then defined solid objectives to inform a potential solution to the aforementioned problems (see number 2 in **Figure 1**). Our aim is to find a solution that benefits

Table 1. Relations between five dimensions and user interaction design variables.

	Low score	High score
PDI	Less structure data Supportive message Informal representative	Complex structure data Strict error message Formal representative
IDV	High context Colorful interface	Low context Tediously colored interface
MAS	Multiple choices/tasks Social structure (relationship orientation)	Limited choice/task Business structure (goal orientation)
UAI	Complex information Abstraction representation	Simple/precise information Daily representation
LTO	Tolerance complex communication	Preference for friendly communication

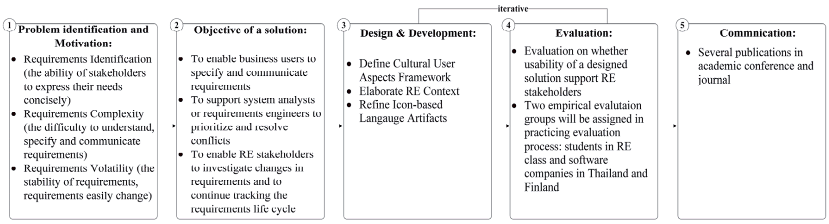

Figure 1. Design science research methodology.

RE stakeholders, particularly in multicultural environments. Large diversity in the cultural backgrounds of stakeholders makes it indispensable to find ways easily adapting interaction. For this adaption process, icon-based information has three benefits.

The first benefit is to enable business users to specify and communicate requirements. The second benefit is to support system analysts or requirements engineers to prioritize and resolve conflicts. The third benefit is to enable RE stakeholders to investigate changes in requirements and to continue tracking the requirements life cycle.

At the design stage (see number 3 in **Figure 1**), the key insights of our approach can be obtained by defining cultural user aspects, elaborating RE artifacts and refining icon artifacts. The details of these three artifacts are described in the next section. We inferred the necessities for our artifacts by drawing on theoretical foundations in the interdisciplinary fields of RE, human-computer interaction, and cognitive psychology. We further combined knowledge and techniques from the research fields of modeling languages and iconic communication in order to make design decisions that principally affect the direction of our approach.

In the evaluation phase (see number 4 in **Figure 1**), we conducted two iterative evaluations: one with student users and another with expert users. The first iteration was tested by students in the RE course of the Department of Mathematical Information Technology at the University of Jyväskylä. The results of students from the first iteration are detailed in Section 5. For the latter iteration, software companies in Thailand and Finland (including Australia, if possible) will be the notable key players. We took advantage of usability testing to evaluate if a defined icon-based language supports the tasks of RE stakeholders. The results of these two iterations will be used to inform improvement possibilities.

4. Designing for Icon-Based Information in the RE Domain

4.1. Modeling Requirements Engineering

We established a list of RE artifacts to support multicultural environments. It is focused on delineating potential activities that affect the entire RE process and that help diminish ambiguity and misinterpretation. **Figure 2** shows an example of how the central concept of requirements artifacts is in relation to stakeholders, attributes, relationships and taxonomies. Each requirement is proposed by a stakeholder and thus it is essential to record information about associated stakeholders. We categorized the requirements into groupings (a taxonomy) to facilitate better organization and management. The eight types were created in order to tackle software quality and development process quality [38]. Business requirement is an abstraction level that reflects a goal or vision of the organization. Business requirements may contain functional requirements that present the behavior of a system under specific conditions. Otherwise, the level also includes non-functional requirements that represent a quality attribute the system must have, such as reliability, usability, efficiency, maintainability, or portability. A requirement may be appended with business rules, laws, policies, or procedures which constrain the degree of freedom in delivering a solution.

The importance and urgency (priority) assigned to every individual requirement helps moderate imprecise conflicting requirements and assist the development team in identifying the core requirements. A requested requirement must be assigned one out of five priority levels. A "very high" priority is both important and urgent, a "high" priority is important but not urgent, a "fair" priority is neither important nor urgent, a "low" priority is neither important nor urgent and can wait for the next release, and a "very low" priority is for when the requirement can be delayed for the next release or not implemented.

A preference score is attached to the initialized requirement. It stands for the degree of preference or satisfaction of the requirement for each stakeholder. The score, arranged by each stakeholder, can be used to inspect the similarity of detected overlap among stakeholders.

The classification of several statuses is more meaningful to help stakeholders monitor the progress of each single requirement throughout development process:

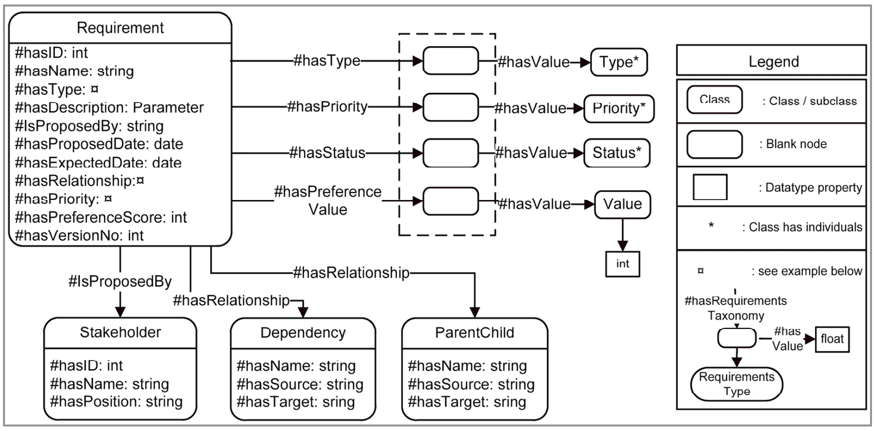

Figure 2. Model of requirements management with relevant attributes.

"Propose" when the requirement has been initialed by an authorized source; "Accept" when the requirement is analyzed and key stakeholders agree to incorporate such requirement; "Reject" if the requirement is proposed but it is not planned for implementation; "Implement" when designing, writing and testing the source code that implements the requirement, and "Verify" when verifying for the correct functionality of implemented requirement.

The existence of a relationship between two or more requirements presents and reasons taxonomy of traceability. The requirements can be associated to each other through the link types: dependency or parent-child. Three relationships of dependency (require, refine, and conflict) have been delineated to qualify the association between two or more requirements. Additionally, one requirement can be divided into sub-requirements and those sub-requirements are connected to their parent with a parent-child link. The requirements engineer can use two types corresponding to a logical combination—one is AND-parent-child and the other is OR-parent-child. With an AND relationship, unless all sub-requirements are satisfied, their parent requirement cannot be satisfied. On the contrary, with an OR relationship, a parent requirement could be satisfied when at least one sub-requirement is satisfied.

4.2. Modeling Icon-Based Information

Although many visual features (e.g. size, shape and color) have been allocated to aid the recognition, it is now acknowledged that the correlation between recognition and interpretation also plays a crucial role [39,40]. Interpreting and comprehending a single icon is the simplest process of reading iconographic communication, but when icon interpretation occurs in isolation, it makes icon interpretation too complex [39]. Currently, the outstanding icon characteristic that has received the most attention is icon concreteness. In this characteristic, icons are used to represent real-world objects because they happen to convey meaning accurately [23,39]. Abstract icons, in contrast, represent information using visual features such as shapes, arrows, and colors. We propose that the RE context cannot be wholly represented with concrete icons. As a consequence, we decided to combine concrete and abstract icons to represent RE artifacts. This combination is a plausible reason for making icon-based language scalable, so that a single icon may share the same semantic system.

In **Figure 3**, we depict an icon-based ontology. First we need to derive the icon library of a visual notation being designed to attach to the requirement itself, requirements process and user interface. The main element is the class Icon-based Information which defines icon characteristics. To benefit scalability and variability, we build libraries for Icon-based Information into two categories: separated and shared icon libraries. A MainLib acts as a centric base and will store all icons that can be shared among other three Libraries. UILib mainly collects icons that will be utilized for user interface whereas ProcessLib serves icons that associate to the process output. Likewise, AttributeLib provides icons related to attributes that can be adhere to every requirement. Icons in these four libraries must be design in accordance with cultural aspects of Hoftede's dimensions. Other four subclasses are: Position which characterizes the icon orientation in X and Y axis, Size which exemplifies icon size including 1D iconic elements (lines), 2D iconic ele-

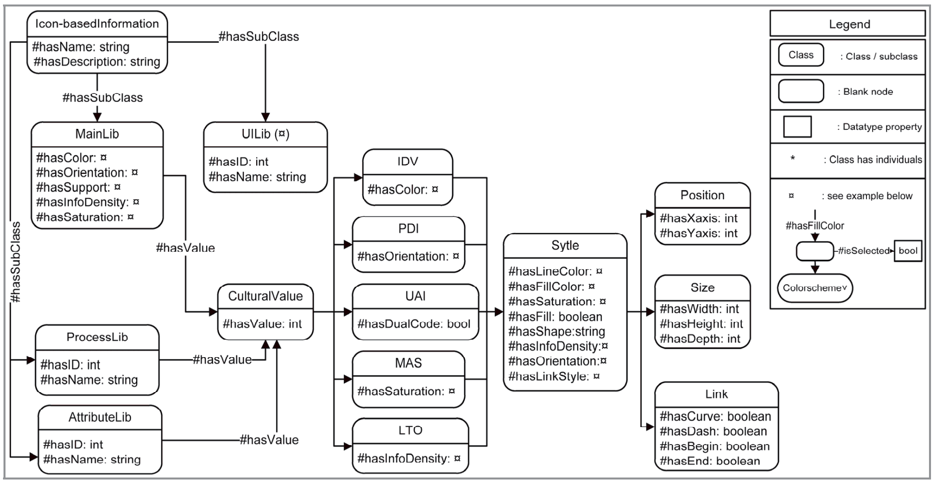

Figure 3. A set of icon classes and variables.

ments (areas), and 3D graphic elements (volumes), Style which typifies the color and shape, and Link which symbolizes link property such as curve and dashed lines.

For each icon notation that has to be conducted, we must generate a final library. The library contains the series of iconic symbols for both abstraction and concreteness, and visual sentences that symbolize the icon notation. When designing iconic symbols, guidelines and standards such as ETSI EG: 202-048 [41] and ISO/IEC 11581 [42] can guide us to the applicable design for a particular purpose. Since the interpretation of icons is subjective, the icons should be properly selected, developed and evaluated. Therefore, we applied the ETSI EG 201-379 [43] framework so that its direction would ultimately solve such a challenge. To solve the problems of misinterpretation and cultural prejudice, the test participants were chosen to include different nationalities.

Next, we refined the syntax specification of the iconic symbols in accordance with the attribute-based representation approach, which must conform to the criteria for proper visual syntax. When it is grounded in the attribute-based tactic, the grammar of icon-based language can be classified, depending on the structure of its iconic objects, in the way they can be composed in order to form a visual sentence. The criteria for good visual syntax are based on cognitive effective psychology [14,23, 24]. They mainly point to the essential characteristics of icons that must be taken into consideration, for instance, familiarity, concreteness, complexity, meaningfulness and semantic distance. All stages in icon-based language design are iterative [44,45], which means that if tests expose usage drawbacks, we might decide to review the libraries as well as to replicate the usability testing in the next version. Our limitation is the fact that all visual vocabularies in this paper were gathered from existing ones and only used to represent concepts and ideas. At this state we are not concerned with their appearance regarding what they are representing. The design perspective must be taken by the designers who are experts in the area. Design relies enormously on cultural experience and cognitive effectiveness.

4.3. Modeling Cultural Aspects

In combination with cultural geographic aspects, we have based our cultural theoretical aspects on those refined in a number of sources [30,31,34]. We focused on extracting the aspects of culture that impact icon usage by conducting a systematic literature review of related work from the interdisciplinary fields of human-computer interaction, cognitive psychology, and RE.

As shown in **Figure 4**, all of these cultural aspects are rudimentarily defined in the web ontology language (OWL) [28]. OWL gives an advantage in that it is an extensible way to represent uniquely identified objects that can be asserted across various users and agents [46]. The focal concept in cultural aspects is the Person class together with its subclass of Female and Male. The Person class further connects to the classes Education Level, Religion and Computer Literacy. Data type properties of the range integer record the five national dimensions. To model the cultural influence of different nationals, the ontology composes the object property, has Nationatlity.

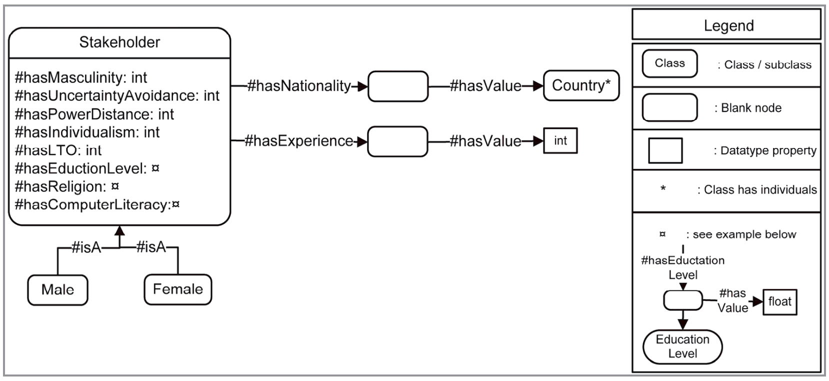

Figure 4. A set of cultural components that govern icon-based interface adaptation.

The ontology also pertains to Country class, which contains individuals of all continents and countries, but in the beginning we have focused on only three countries (Thailand, Finland and Australia). Likewise, the ontology has been complemented with the class has Experience, which provides us with information about a user's RE knowledge and skill.

No # Month class is inherited to has Expereince class in order to provide a validated series of months. In this paper we have not yet implemented this ontology. Instead we have drawn upon the theoretical framework to derive a cultural conceptual process for executing the first empirical evaluation by students in RE course at the University of Jyväskylä, Finland. This framework can be further improved and developed for icon-based information adaptivity to specify different icon preferences that correspond to cultures.

Figure 5 illustrates the convergence of three main artifacts—cultural user framework, RE artifacts and icon-based information—to achieve an icon-based information adaptivity process. First, it is necessary to build a user framework on cultural particularities before the adaptation can be achieved. The idea is that the register process elicits the users' background by taking into account various influences that affect a user's iconic preference, such as their nationality and work experience. This information is passed on and stored in a cultural user framework (CUF). A CUF acts as a knowledge base about each user and inherited rules that trigger the adaptation of the icon-based interface.

When the user framework is used for the first time, the user needs to provide information in a short questionnaire arranged by an application. This acquired information helps to manage the icon-based information for the user according to that user's national preference. The application receives the cultural dimensions for each

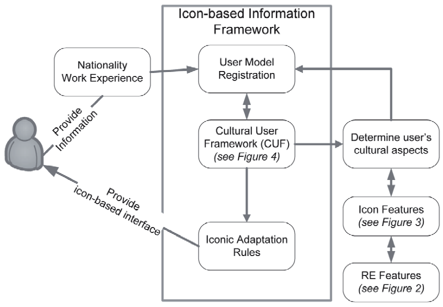

Figure 5. The process for icon-based adaptivity.

user's nationality from the CUF. The application retrieves the RE features and the embedded iconic features that corresponded to the user's background information. The icon-based interface is tailored to the user on the basis of adaptation rules. For instance, if a user has a high score in UAI, then an interface with very simple, clear imagery and limited choice is provided. Since icon-based information for RE that can adapt its appearance differently among cultures is a novel approach, its design can be partly applied from cultural adaptivity [47] that involves the refinement of interface preferences. Users can interact with this application (MediaWiki), which is enabled to access the cultural user framework. Users can also explicitly add or modify information in their personal user registration. This, in turn, triggers adequate adaptations that change the icon information of the user interface.

5. Experiments

In this section we report on our summative evaluation of the ability of icon-based information for RE to adequately adapt to varying requirements scenarios. The evaluation was carried out to ensure that icons are effective and usable. Icon usability testing was conducted to assess the degree to which the graphic chosen for the icon represented the intended concept, so-called icon intuitiveness [39]. The study focuses on participants with a Finnish cultural background.

5.1. Participants

To evaluate the cognition of icon-based information for requirements engineering by diverse participants, we invited students attending a Requirements Engineering course at the University of Jyväskylä to participate in the study. Some of them were studying the subject for first time and some already had experience in software engineering. A total of 45 students took part in this study: 14 novices, (0 years of experience) and 31 experts (an average of 1.2 years of experience). Each student completed three testing dimensions: individual icon interpretation, multiple icon interpretation and compound icon construction.

5.2. Test Apparatus

A web-based survey, a common instrument for gathering information from participants, was set up for the empirical evaluation. The icons were presented to participants on webpages. The website consisted of three primary sections: background information, a form for personal information and the icon test. Each test contained an explanation that helped users to complete the task.

5.3. Procedure

Prior to the real execution of tasks, we briefly explained to participants the purpose of the testing, the amount of tasks they needed to complete and the step-by-step interaction. In addition, iconic symbols were chosen from a variety of sources in order to ensure that they were representative of the icons that are currently well-known in the broad spectrum of RE artifacts. These included the use of norms from standards such as ISO/IEC-11581 [42]. All the selected icons were abstract ones with predictable meanings. However, we selected the icons that could be simply interpreted and recognized without requiring prior knowledge. For this study, our selection consisted of 14 icons, including 5 icons for priority, 5 icons for status, and 4 icons for stakeholder type. Throughout the experiment, participants were encouraged to interpret the icons from the details they were given. The empirical survey ended with a small questionnaire that gathered feedback and comments on icon-based information in RE. Participants completed three series of tasks. For examples, see **Figure 6**.

5.4. Preliminary Result

5.4.1. Individual Icon Interpretation

The requirements life cycle diagram, which consists of five blanks (see **Figure 6(a)**), was delivered to all replies, in conjunction with icons that resemble all five stages of the requirements life cycle: Propose, Accept, Reject, Implement and Verify. The participant selected the most correlated icon and dropped it into every single blank stage. The number of accurate selections per stage varied from 52% to 94%.

5.4.2. Multiple Icon Interpretation

We categorized the iconic symbols into three requirements attributes: five status states, five priority types and four stakeholder kinds. Each respondent mapped icons to their corresponding meaning from two lists (see **Figure 6(b)**): one list of icons and another list of meanings. **Table 2** presents the outcomes for interpreting the icons and their meaning for three distinct categories.

5.4.3. Compound Icon Construction

The icons were also used to represent requirements type symbols and relationships imitating the concept of a goal-oriented model (see **Figure 6(c)**). We offered four types (business requirement, functional requirement, non-functional requirements, and constraint), four relationship symbols (Refine, Require, AND and OR), and five priority symbols (Very high, High, Fair, Low and Very low). Each respondent constructed the iconic sentence according to the statement. The correctness of iconic constructs achieved a prediction accuracy of approximately 84%.

5.5. Role of Experience

The expected outcome of this study was that practitioners would be able to recognize the iconic symbols representing RE artifacts, such as stakeholders, attributes and relationships. From the results, we observed that icons were interpreted very well. The average correct prediction accuracy was more than 50 percent. This finding informs our direction to explore improvement possibilities of icon-based information. The individual icon testing revealed that the participants' capability to answer the status states required prior knowledge and experience of the requirements life cycle and the interaction that takes place during the process, from the beginning of proposing to the end of verification. Training before taking part in this test would probably assist the respondents in understanding the basic elements of RE. The multiple icons testing showed that a participant's interpretation depended to some extent on their conventional knowledge. For example, participants who have children might perceive a baby carriage as high priority. There was also

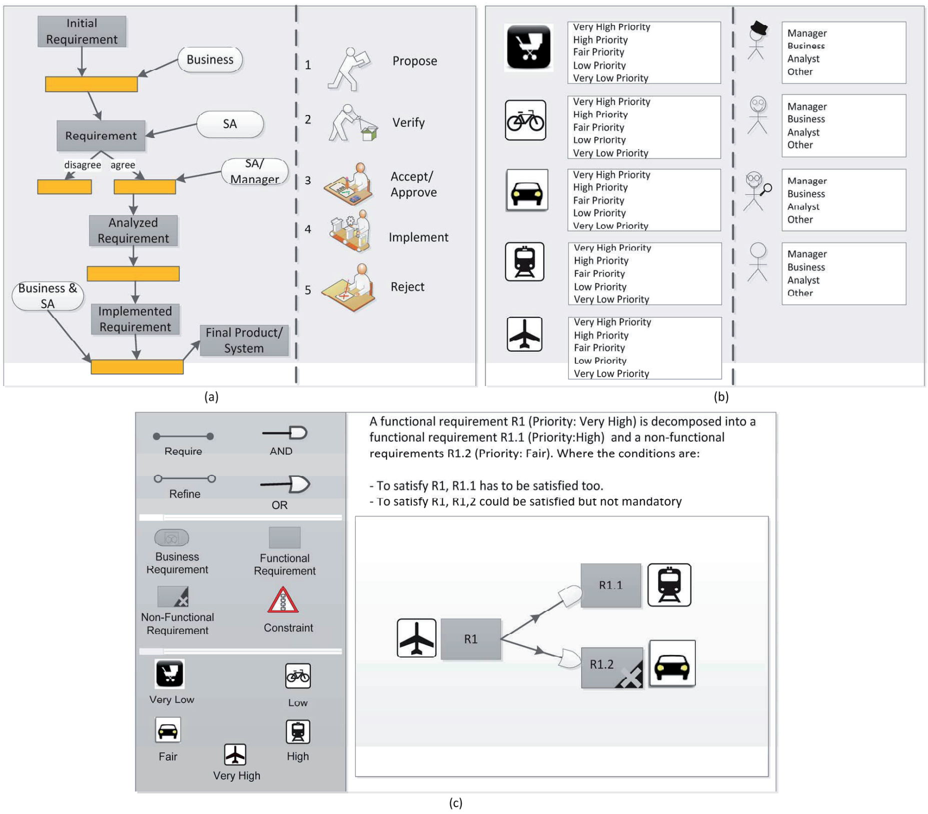

Figure 6. The test task series, (a) type 1: individual icon mapping that exemplifies the requirements life cycle; (b) type 2: multiple icon meaning that typifies requirements attributes; and (c) type 3: iconic sentence construction that characterizes requirements types and relationships.

confusion over whether a train or a car is faster. The results for multiple icon testing also revealed that out of interpretations for 14 icons, roughly 7.5% generated misunderstanding. In the difficult recognition of goal-graph relationship, compound icon construction generated positive feedback. Apparently, the use of dual-code symbols with label captions enabled easy apprehension, with the result being roughly 80% correct for the constructions.

We proposed the null hypothesis (H0) that the interpretations for icons of practitioners in the same country are similar and we also put forward the hypothesis (H1) that the interpretations of icons by practitioners in the same country can be different. These hypotheses were informed by statistical results from binomial confidence intervals. The binomial confidential interval is persuasive because only two outcomes are possible in each identical trial: success (correct) or failure (incorrect). As the outputs in **Table 3** show, we reject the null hypothesis and accept H1 because even though people in the same country tested icons, some misinterpretations remained. We conclude, therefore, that iconic representations are dependent on context and personal perspective. The difficulties of interpretation might have arisen because the set of icons chosen were not obviously representative examples of the intended functions.

We tested another hypothesis that no differences exist in the interpretation of icons by novices and experienced participants among the three test series. In **Table 4**, we used the Fisher Exact Probability Test to investigate the

Table 2. Summary of the icon-tested results for N = 45 (in percent).

Priority types		Life Cycle (status) states		Stakeholder kinds	
Icon & Meaning	**Correct prediction**	**Icon & Meaning**	**Correct prediction**	**Icon & Meaning**	**Correct prediction**
Very Low 	66.66	Propose 	82.22	Manager 	84.44
Low 	68.88	Verify 	88.88	Business 	77.77
Fair 	73.33	Accept 	95.55	Analyst 	97.77
High 	75.55	Implement 	82.22	Other 	91.11
Very High 	77.77	Reject 	66.66		

Table 3. Three categories of icons test for finnish country with confidential level 95%.

Type of Question	Test Results
Individual Icon Interpretation	The 95% confidence interval for the proportion of potential practitioners who can place icons in the requirements life cycle stage well is 0.3650 to 0.6572.
Multiple Icon Interpretation	The 95% confidence interval for the proportion of potential practitioners who can interpret multi-icons well is 0.7242 to 0.9297.
Compound Icon Construction	The 95% confidence interval for the proportion of potential practitioners who can construct iconic sentence well is 0.7386 to 0.9503.

Table 4. The differences in the distribution of interpretation between novices and specialists within Finnish culture.

Finnish culture	P value between two groups
Individual Icon Interpretation	0.659
Multiple Icon Interpretation	0.720
Compound Icon Construction	0.749

distribution of two participant groups: novices and specialists. This test resulted in a contingency table with α = 0.05. We can reject the above hypothesis only if the P value is less than the alpha value. With derived probabilities (P value) for all tests greater than $\alpha = 0.05$, it would confirm the hypothesis. We conclude that work experience is not associated (at the $\alpha = 0.05$ level) with how novices and specialists were able to comprehend the meaning of icons.

6. Discussions and Recommendations for Improvement

We have mentioned the survey outcomes of three sample sizes that belong to other nationalities. Not only can we not use these small numbers as nationally representative, but also all of them are affiliated with the experienced group, so it is insufficient to analyze a two-sample test. **Table 5** shows the proportional correctness by percentage. Even though it was our intention to focus on participants with a multicultural background, the low number (3) of other participants of other nationalities means those are not significant enough to be systematically exploited for cultural summarization. We can only make clear conclusions regarding users with Finnish nationality. This limitation motivates us to further evaluate these issues with a significant number of participants from other cultures.

However, the satisfactory result for priority types has several implications for our approach. We cannot readily presume that our proposed icons generalize to any person in any country. This means that it is important to provide users with clear-cut groups of requirements priorities. Breaking down priority into three scales—high, medium, and low—could advance straightforward recognition and utilization [48]. First, high priority requirements are important and urgent. We advise the use of an aircraft icon for high priority. Second, medium priority requirements are important but not urgent. We recommend a car icon for medium priority. Finally, low priority requirements are neither important nor urgent. We encourage the use

Table 5. The summary of correct interpretation of 2 groups for Finnish and 1 group for other cultures (3 replies).

Finnish culture	Novices	Experts	Other cultures
Individual Icon Interpretation	57.15	48.39	66.7
Multiple Icon Interpretation	82.14	82.95	90.5
Compound Icon Construction	78.58	87.10	100

of a bicycle icon for this priority.

While the results of stakeholder kinds are encouraging, they also expose the need for amendment. In practice, a sticky figure is used in as a mnemonic device for the ACTOR in the Use Case diagram. To increase the semantic transparency and cognitive effectiveness of stakeholder kinds in icon-based information, the detailed image of people and circumstances can be employed as portrayed in **Figure 7**.

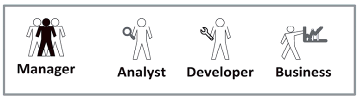

Figure 7. The stakeholder icons.

The results of life cycle mapping indicate the confusion if icons are separately presented. When participants have all status icons to compare, it seems to be easy. But if respondents need to match one distinct icon to one of five life cycle stages without seeing the other icons, it becomes more difficult for some of those icons. To avoid this difficulty, icons with text description or textual encoding could expand and reinforce the meaning of icons more effectively than using either on their own [14]. Text can reinforce an icon's meaning by offering an additional hint to what they mean.

The comparison between novice and experienced users is especially interesting, because the use of icon-based information is not relied on the degree of experience. With the statistics in **Table 5** (Finnish Culture) as our basis, we can justify that it is not always true that specialists are capable of understanding icons better than amateurs. This finding enlightened our decision to investigate the obstacles encountered by users and we realized that icon-based information is a new approach for both novice and experienced users. Thus, they probably need assistance in the form of training or education before dealing with the approach in actual situations.

7. Conclusions

One of the strongest justifications for the use of symbols, notably for icons, is that they are easy to use and understand. One of the strongest claims made for RE is that it is the most vital factor influencing the success of software. Consequently, the importance of RE and advantages of icons motivate the current research on adapting icons for RE to bridge the communication barriers between business stakeholders and development teams. The main objective of this work is to introduce icon-based information as an alternative communication medium for multifaceted stakeholders and thereby to enrich RE. The competency of icon-based information could help describe and communicate requirements. Generally speaking, icon-based information could be appropriate for expressing ideas in a wide range of uses, from business desires and requirements descriptions to high-level architecture design. Icon-based information could support elicitation, analysis and traceability of requirements.

The results of our study suggest that icons can be used to support RE. Icons are capable of heightening cognitive effectiveness if they are properly designed and used. We argue that abstract icons are possible to signify the concept in RE. Our results do not show, however, whether the designs of our icons are sufficient for all aspects of RE. Further work is needed to establish the cultural user framework (CUF). Moreover, we have to implement improvements according to the feedback obtained in the first iteration and the second experimental iteration, which will be evaluated by software companies.

REFERENCES

[1] O. W. Galitz, "The Essential Guide to User Interface Design an Introduction to GUI Design Principles and Techniques," 3rd Edition, Wiley Publishing, Inc., 2007.

[2] A. Marcus, "Icons, Symbols, and Signs: Visible Languages to Facilitate Communication," *Interactions*, Vol. 10, No. 3, 2003, pp. 37-43. http://dx.doi.org/10.1145/769759.769774

[3] Heimbürger, *et al*., "Intelligent Icons for Cross-Cultural Knowledge Searching," *Information Modelling and Knowledge Bases XXIII*, Amsterdam, 2012, pp. 77-89.

[4] A. Heimbürger and Y. Kiyoki, "Pictorial Symbols in Context: A Means for Visual Communication in Cross-Cultural Environments," *Proceedings of the IADIS International Conferences*, Germany, 27-29 July 2010, pp. 463-467.

[5] S. Khanom, A. Heimbürger and T. Kärkkäinen, "Icon-Based Language in Requirements Development," 22*nd EJC*, 2012, pp. 20-25.

[6] S. Khanom, A. Heimbürger and T. Kärkkäinen, "Icon-Based Language in the Context of Requirements Elicitation Process," *Information Modelling and Knowledge Base XXIV*, IOS Press, Amsterdam, in press.

[7] S. Khanom, "Icon-Based Language in the Context of Requirements Engineering," *REFSQ Requirements Engineering*: *Foundation for Software Quality*, Germany, 8-11 April 2013, pp. 215-222.

[8] P. G. Barker and M. Yazdani, "Iconic Communication," Intellect Ltd., 2000.

[9] S. Hansen and K. Lyytinen, "Challenges in Contemporary Requirements Practice," 43*rd Hawaii International Conference on System Sciences* (*HICSS*), Honolulu, 5-8 January 2010, pp. 1-11.

[10] L. Mathiassen, T. Tuunanen, T. Saarinen and M. Rossi, "A Contigency Model for Requirements Development," *JAIS*, Vol. 8, No. 11, 2007, pp. 569-597.

[11] H. Kaiya, D. Shinbara, J. Kawano and M. Saeki, "Improving the Detection of Requirements Discordances among Stakeholders," *Requirements Engineering*, Vol. 10, No. 4, 2005, pp. 289-303.

[12] N. Cerpa and J. M. Verner, "Why Did Your Project Fail?" *Communication of the ACM*, Vol. 52, No. 12, 2009, pp. 130-134. http://dx.doi.org/10.1145/1610252.1610286

[13] K. El Emam and A. G. Koru, "A Replicated Survey of IT Software Project Failures," *IEEE Software*, Vol. 25, No. 5, 2008, pp. 84-90. http://dx.doi.org/10.1109/MS.2008.107

[14] D. L. Moody, "The 'Physics' of Notations: Toward a Scientific Basis for Constructing Visual Notations in Software Engineering," *IEEE Transactions on Software Engineering*, Vol. 35, No. 6, 2009, pp. 756-779. http://dx.doi.org/10.1109/TSE.2009.67

[15] N. Aykin, "Usability and Internationalization of Information Technology," Lawrence Erlbaum Association, Inc., London, 2005.

[16] R. Bendraou, J. Jezequel, M. Gervais and X. Blanc, "A Comparison of Six UML-Based Languages for Software Process Modeling," *IEEE Transactions on Software Engineering*, Vol. 36, No. 5, 2010, pp. 662-675. http://dx.doi.org/10.1109/TSE.2009.85

[17] S. Morris and G. Spanoudakis, "UML: An Evaluation of the Visual Syntax of the Language," *Proceedings of the* 34*th Annual Hawaii International Conference on System Sciences*, Hawaii, 3-6 January 2001, p. 10.

[18] L. Duboc, E. Letier and D. S. Rosenblum, "Systematic Elaboration of Scalability Requirements through Goal-Obstacle Analysis," *IEEE Transactions on Software Engineering*, Vol. 39, No. 1, 2013, pp. 119-140. http://dx.doi.org/10.1109/TSE.2012.12

[19] D. L. Moody, P. Heymans and R. Matulevičius, "Visual Syntax Does Matter: Improving the Cognitive Effectiveness of the *i** Visual Notation," *Requirements Engineering*, Vol. 15, No. 2, 2010, pp. 141-175. http://dx.doi.org/10.1007/s00766-010-0100-1

[20] D. L. Moody and J. V. Hillegersberg, "Evaluating the Visual Syntax of UML: An Analysis of the Cognitive Effectiveness of the UML Family of Diagrams," *Software Language Engineering*, Vol. 5452, 2009, pp. 16-34. http://dx.doi.org/10.1007/978-3-642-00434-6_3

[21] S. J. P. McDougall, M. B. Curry and O. D. Bruijn, "The Effects of Visual Information on Users' Mental Models: An Evaluation of Pathfinder Analysis as a Measure of Icon Usability," *Journal of Cognitive Ergonomics*, Vol. 5, No. 1, 2001, pp. 59-84. http://dx.doi.org/10.1207/S15327566IJCE0501_4

[22] H. F. Wang, S. H. Hung and C. C. Liao, "A Survey of Icon Taxonomy Used in the Interface Design," *Proceedings of the* 14*th European Conference on Cognitive Ergonomics*: *Invent*! *Explore*! London, 28-31 August 2007, pp. 203-206.

[23] A. W. Y. Ng and A. H. S. Chan, "Visual and Cognitive Features on Icon Effectiveness," *Proceedings of the International Multiconference of Engineers and Computer Scientists*, Vol. 2, 2008, pp. 19-21.

[24] S. J. P. McDougall, M. B. Curry and O. D. Bruijn, "Measuring Symbol and Icon Characteristics: Norms for Concreteness, Complexity, Meaningfulness, Familiarity, and Semantic Distance for 239 Symbols," *Behavior Research Methods*, *Instruments & Computers*, Vol. 31, No. 3, 1999, pp. 487-519. http://dx.doi.org/10.3758/BF03200730

[25] Y. Yu and J. D. He, "An Analysis of Users' Cognitive Factors towards Icon in Interactive Interface," *Intelligent* 2*nd International Conference on Intelligent Human-Machine Systems and Cybernetics* (*IHMSC*), Nanjing, 26-28 August 2010, pp. 26-28.

[26] S. Fitrianie and L. J. M. Rothkrantz, "A Visual Communication Language for Crisis Management," *International Journal of Intelligent Control and Systems*, Vol. 12, No. 2, 2007, pp. 208-216.

[27] S. Fitrianie, D. Datcu and L. J. M. Rothkrantz, "Human Communication Based on Icons in Crisis Environments," *Usability and Internationalization*, Vol. 4560, 2007, pp. 57-66. http://dx.doi.org/10.1007/978-3-540-73289-1_7

[28] D. L. McGuinness and F. V. Harmelen, "OWL Web Ontology Language Overview," 2004. http://www.w3.org/TR/owl-features/

[29] G. Hofstede, "Cultures and organizations: Software of the Mind," McGraw-Hill, New York, 1997.

[30] K. Reinecke and A. Bernstein, "Improving Performance, Perceived Usability, and Aesthetics with Culturally Adaptive User Interfaces," *ACM Transactions on Computer-Human Interaction*, Vol. 18, No. 2, 2011, pp. 8-29. http://dx.doi.org/10.1145/1970378.1970382

[31] D. E. Leidner and T. Kayworth, "A Review of Culture in Information Systems: Toward a Theory of Information Technology Culture Conflict," *MIS Quarterly*, Vol. 30, No. 2, 2006, pp. 357-399.

[32] A. Heimbürger, *et al.*, "Communication across Cultures in the Context of Multicultural," *Software Development. Reports of the Department of MIT. Series C. Software and Computational Engineering*, 2011.

[33] S. K. Ackerman, "Mapping User Interface Design to Culture Dimensions," *Proceedings of IWIPS*, Texas, 11-13 July 2002, pp. 89-100.

[34] S. Shen, M. Woolley and S. Prior, "Towards Culture-Centred Design," *Interacting with Computers*, Vol. 18, No. 4, 2006, pp. 820-852. http://dx.doi.org/10.1016/j.intcom.2005.11.014

[35] A. Marcus and E. W. Gould, "Crosscurrents: Cultural Dimensions and Global Web User-Interface Design," *Interactions*, Vol.7, No. 2, 2002, pp. 32-46.

[36] A. R. Hevner, S. T. March, J. Park and S. Ram, "Design Science in Information Systems Research," *MIS Quarterly*, Vol. 28, No. 1, 2004, pp. 74-105.

[37] K. Peffers, T. Tuunanen, M. A. Rothenberger and S. Chatterjee, "A Design Science Research Methodology for

Information Systems Research," *Journal of Management Information Systems*, Vol. 24, No. 3, 2007, pp. 45-78. http://dx.doi.org/10.2753/MIS0742-1222240302

[38] ISO/IEC 9126, "IT-Software Product Evaluation-Quality Characteristics," International Organization for Standardization, Geneve, 1991.

[39] S. J. P. McDougall and M. Burry, "More than Just a Picture: Icon Interpretation in Context," Coping with Complexity Workshop, University of Bath, 2007, pp. 1-8.

[40] S. Isherwood, "Graphics and Semantics: The Relationship between What Is Seen and What Is Meant in Icon Design," *Engineering Psychology and Cognitive Ergonomics*, Vol. 5639, 2009, pp. 197-205. http://dx.doi.org/10.1007/978-3-642-02728-4_21

[41] ETSI EG 202-048, "Human Factors (HF); Guideline on the Multimodality of Icons, Symbols, and Pictograms," European Telecommunication Standard Institute, France, 2002.

[42] ISO/IEC 11581, "Information Technology-User System Interfaces-Icon Symbols and Functions. Part 1: General Icons, Part 2: Object Icons, Part 6: Action Icons," ISO Copyright, Switzerland, 2002.

[43] ETSI EG 201-379, "Human Factors (HF); Framework for the Development, Evaluation and Selection of Graphical Symbols," European Telecommunications Standards Institute, France, 1998.

[44] G. Costagliola, V. Deufemia and G. Polese, "A Framework for Modeling and Implementing Visual Notations with Applications to Software Engineering," *ACM Transactions on Software Engineering and Methodology*, Vol. 13, No. 4, 2004, pp. 431-487. http://dx.doi.org/10.1145/1040291.1040293

[45] S. Khanom, A. Heimbürger and T. Kärkkäinen, "Icon-Based Language: Auxiliary Communication for Requirements Engineering," *Ijest*, Vol. 5, 2013, pp. 1076-1082.

[46] Aroyo, *et al.*, "Interoperability in Personalized Adaptive Learning," *Educational Technology & Society*, Vol. 9, No. 2, 2006, pp. 4-18.

[47] K. Reinecke and A. Bernstein, "Knowing What a User Likes: A Design Science Approach to Interfaces That Automatically Adapt to Culture," *MIS Quarterly*, Vol.37, No. 2, 2013, pp. 427-453.

[48] K. E. Wiegers, "Software Requirements," 2nd Edition, Microsoft Press, 2003.

VII

ICONS RECOGNITION AND USABILITY FOR REQUIREMENTS ENGINEERING

by

Sukanya Khanom, Anneli Heimbürger & Tommi Kärkkäinen, 2015

In 2014, B. Thalheim, H. Jaakkola & Y. Kiyoki (Eds.) 24th International Conference on Information Modeling and Knowledge Bases. Kiel, Germany, June 3-6, 248.

Re-edit paper will be published in Information Modelling and Knowledge Base XXVI. ISO Press, Amsterdam, (in press).

VIII

CAN ICONS ENHANCE REQUIREMENTS ENGINEERING WORK?

by

Sukanya Khanom, Anneli Heimbürger & Tommi Kärkkäinen, 2014

In Journal of Visual Languages & Computing (submitted).