Balancing Functionality, Risk, and Cost in Smart Service Networks

F. Rizal Batubara
Services, Cyber Security and Safety Research Group, University of Twente, Enschede, The Netherlands

1 CONTEXT

Recent developments in the area of software service systems brought a new level of scale, complexity and pervasiveness to cope with. This has, in turn, changed the business model of companies engaged in the software industry and the way they present their services. Healthcare systems, as an example, transform services from a traditional professional-centric care model to model that is characterized as 'pervasive healthcare'. Pervasive healthcare applies extramural networked healthcare systems with sensors and devices at the patients' location, making the patient an active partner in the care process. These transformations of technologies and business models have brought a new breed of systems called Smart Service Networks (SSN). An SSN involves any number of devices, sensors and IT systems, and has diverse stakeholders, which together form a network in which resources are integrated and applied through interaction (Stroulia, 2010).

An SSN is a context-aware system, it can adapt and provide specific services to the user according to the context information received from data collected with sensors. Therefore, SSN must be able to clearly understand the significance of the context information conveyed from its environment. The quality of the context model determines the extent to which the SSN can offer services that fit the actual context, which in turn determines the usefulness of the offered services to the user. The quality of the context model depends on the quality of the collected context data (accuracy, timeliness, etc.). Low(er) data quality means low(er) model quality, which can lead to (more) off-topic service offerings. Off-topic service offerings can be useless to the user, and may even have negative value to the user.

SSN development projects operate in a multi-stakeholders context and potentially experience conflicts among functionality, risk and cost to meet the stakeholders' requirements. This research focuses on trading-off functionality, risk and cost within SSN development. By defining the possible trade-off scenarios in SSN delivery projects, this research will provide a rational analysis to justify SSN design decisions that lead to achieving software systems with adequate functionality, minimum risk and reasonable cost.

2 RESEARCH PROBLEM

Developing a SSN is a valuable business investment decision that is expected to bring returns from its investments (Van Den Heuvel and Papazoglou, 2010). Surely the stakeholders expect the system with as complete as possible functionalities and as high as possible quality, to earn profit for them. While meeting the required functionality, the developers must also meet safety standards and comply with cost estimates based on risk analysis. Strategic decisions during the design of a SSN need a joint control of the estimated functionality, risk and cost.

The relationship between functionality, risk, and cost for SSN development (or any IT system for that matter) is described in Figure 1. Therein, the boxes represent variables of interest – in our case functionality, risk, and cost – and the directed connections represent cause-effect relationships. The '+' sign with a connection means that the connected variables change value in the same direction; the '-' sign means that the connected variables change value in opposite directions.

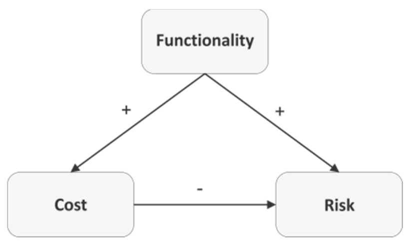

Figure 1: Relationships between functionality, risk, and cost in smart service network development.

We make the note that in Figure 1, the concept of functionality and risk reflect the understanding that in any SSN, there are 'on-topic' SSN services represent value-added functionality and 'off-topic'

SSN services represent risks. Value-added functionality may be enhanced by considering more elaborate context models. However, embedding those models into the SSN would contributes to increased cost. Meanwhile, risks can be mitigated by enhancing the quality of context models and/or only offering services if a threshold quality is satisfied, this contributes to costs as well.

Based on these phenomena in the context of SSN, the problem statement of this research is:

To design a method to balance trade-off among functionality, risk, and cost in order to support decisions on adequate functionality, minimum risk and reasonable cost within smart service network development.

Following the methodology of (Wieringa, 2014), we *formulated* the following research questions (RQ) which address the problem stated:

1. *What is empirical evidence about relationship among functionality, risk, and cost in a smart service network, and what mechanisms explain this relationship? What could be a working definition of an 'optimal solution' from client and vendor perspective regarding SSN systems??*
   - RQ 1.1: What definitions of functionality, risk and cost are useful in the context of SSN?
   - RQ 1.2: What empirical evidence exists about the possible relationships among these three concepts?
   - RQ 1.3: Are there any assumptions about the relationships indicated in published literature sources?
   - RQ 1.4: How can we explain these relationships in terms of mechanisms describing a SSN operating in a context?
   - RQ 1.5: What is an 'optimal balance' among these three concepts, according to published empirical evidence?
2. *How to achieve an optimal solution in order to maximize the functionality, minimize the risk, and minimize the cost within the development of SSN?*
   - RQ 2.1: What methods are available to estimate functionality, risk, and cost within SSN development? What are the requirements for such a method? Which method should we use?
   - RQ 2.2: What approaches, theories, and models allow us to balance the trade-off among these three concepts?
   - RQ 2.3: What is the model to calculate and what are the criteria to decide on an optimal balance of these three concepts?
   - RQ 2.4: How to enhance the SSN design process to incorporate the proposed method for balancing between these three concepts?
   - RQ 2.5: How can a SSN architecture support this enhanced design process, e.g. by facilitating traceability between balancing decisions and architecture components?
3. *How to validate the proposed solution approach to balance among functionality, risk, and cost in SSN system?*
   - RQ 3.1: Can we apply (a prototype of) our method in a setting that allows validation of its properties?
   - RQ 3.2: What are the effects of applying our method?
   - RQ 3.3: Do these effects satisfy the requirements that were put forward on the method – optimizing the balance between functionality, risk, and cost?
   - RQ 3.4: What can we say about the dependencies of the observed effects on the chosen setting, and about the generalizability of the observed effects over different settings?

3 OBJECTIVE

The goal of this PhD research is to design a method for balancing functionality, risk and cost in SSN development projects. This method will be applied to support stakeholders in resolving the trade-off problems that are common for SSN development, thereby helping produce SSN software with adequate functionality, minimum risk and reasonable cost.

4 STATE OF THE ART IN SMART SERVICE NETWORK

Integration of legacy and new software systems through service oriented architecture (SOA) and cross-platform standards opens up opportunities for new services that will endorse smart services in various domains. The integration yields many benefits, with new software systems (and associated technologies) enabling a transformation from traditional services to smart services (OECD, 2013).

To increase value for clients, several SSN can integrate existing smart services in order to deliver new services that accommodate the co-production of new knowledge and services through organic peer-to-peer interactions (Wang et al., 2012). Customers and partners who participate in this network can be mutually providing benefits. Hence, SSN utilizes network services with distributed computing, such as sensor network technologies. They interweave smart physical devices, actively monitoring, analyzing and notifying consumers about everyday service needs (Van Den Heuvel and Papazoglou, 2010).

4.1 Characteristics

In order to establish and develop a new discipline of services such as made possible by a SSN, revealing the SSN's essential characteristics is of paramount importance. Understanding these characteristics will be instrumental to formulate the definition of SSN comprehensively. Plus, from a practical point of view, it will help us establish appropriate models of SSN.

In early research, many researchers have revealed some characteristics of SSN and defined various notions of SSN (OECD, 2013); (Allmendinger and Lombreglia, 2005); (Wang et al., 2012); (Van Den Heuvel and Papazoglou, 2010); (Dos Santos Pacheco et al., 2013); (Stroulia, 2010) (Ng et al., 2010). Yet there is still no widely accepted definition of SSN.

Organization for Economic Co-operation and Development (OECD) defined SSN as the result of a combination of three distinct phenomena, namely machine to machine communication to transmit data, cloud computing to process and display the data, and big data analysis to correlate and interpret the data (OECD, 2013). The expansion of machine to machine communication will enable interconnected devices to form an SSN and produce large volumes of data. Large scale processing will be delivered by cloud computing services and analysis of these data will be undertaken around a process frequently called "Big Data".

(Wang et al., 2012) and (Van Den Heuvel and Papazoglou, 2010) considered SSN as "*systems of service systems that are open, complex and fluid, accommodating the co-production of new knowledge and services through organic peer-to-peer interactions*". "Open" in this case means that the system involves a wide range of systems to improve service innovation in an increasingly complex and dynamic environment. "Complex" relates to a joint effort of interdisciplinary collaboration, cooperation and coordination among the network participants. Further, "liquid" means the system should have creative approaches where services and interactions are implemented with awareness and dynamic adaptation to the users' computational environments, changing policies and unknown requirements. Another characteristic of SSN is that it relies on an IT infrastructure that enables network partners to seamlessly integrate their software services with back-end systems and wireless sensor networks.

(Stroulia, 2010) associated SSN with the system attributes: instrumented, interconnected, and intelligent. The term "instrumented" refers to the embedding of "sensing devices" in the environment where the service-delivery process occurs. "Interconnected" is contributed by large-scale instrumentation of the world around us including data, system and people interconnectedness. Finally, the term "intelligent" refers to the numerous analyses that are possible on the multitude of information available and based on which the system can inform and improve our decision making.

In addition, (Villegas and Müller, 2010) explained that "*a system should be able to reason about its current state and provide functionality based on its context and the current user's matters of concern in order to support smart interaction and smart services*", so that the system has to be reflective and context-aware.

Drawing on the literature sources in this section, we conclude that:

- SSNs have the ability to construct and maintain a model of their dynamic context, and to proactively offer services to the user when these services satisfy corresponding criteria regarding the context model state;
- SSNs may have the ability to learn about new criteria (context patterns) for offering services;
- SSNs use sensing devices, machine to machine communication, and reasoning systems to support the above-mentioned context-awareness and learning;

SSNs consist of many different subsystems from different manufacturers and makes, and therefore should be based as much as possible on open standards.

4.2 Definition

In preparing for this research, we did a preliminary literature survey, by searching relevant articles on SSN in electronic libraries (ACM Digital Library, IEEE Xplore, ISI Web of Science, Scopus and SpringerLink). Our survey found only a few definitions of SSN. Also, we found that SSN terminology covers many synonyms as terms, and it is hard to nail down unique words to define the same concept. However, all these concepts are not in contradiction with each other, as they share some commonalities in their meanings and are partially overlapping. We think we could leverage these concepts and map them against the SSN characteristics, if we want to come up with a more specific definition of SSN. Table 1 provides examples of definitions we found in literature.

Table 1: SSN definitions from literature.

Source	Definition
OECD, 2013	An application or service that is able to learn from previous situations and to communicate the results of these situations to other devices and users.
Wang et al., 2012	Open systems accommodating the co-production of new knowledge and services through organic peer-to-peer interactions.
Van Den Heuvel and Papazoglou, 2010	Leveraging service networks with novel distributed computing, such as sensor network technologies.
Allmendinger and Lombreglia, 2005	A brand new configuration of services that are fundamentally pre-emptive rather than reactive or even proactive. Aggressive actions taken by people based on intelligence to prevent undesirable results. For customers, smart services prevent unpleasant surprise in their lives.

As we could see, the definitions in Table 1 do not describe SSN completely because they do not yet include the characteristics of SSN as identified previously. By analyzing the existing definitions and identified SSN characteristics, SSN can be possibly described as suggested in Figure 2.

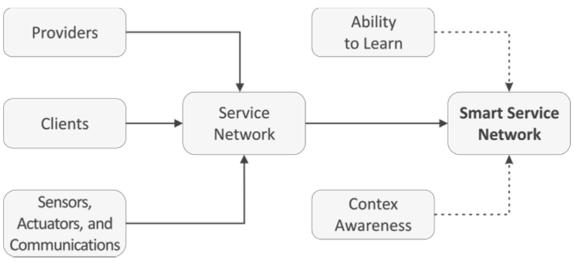

Figure 2: Smart service network (straight-line arrows mean compositions and broken-line arrows mean added capabilities).

Furthermore, for the purpose of this research, we propose the following comprehensive definition of the SSN:

"A *Smart Service Network is a dynamic system which combines smart services and it offers services in accordance its context and user needs, using sensing devices, machine to machine communication, and reasoning systems*"

For example, let's consider a pervasive healthcare system which is able remotely to monitor a patient's vital parameters, such as electrocardiography (ECG), blood pressure and oxygen saturation level. The patient is equipped with ambulatory sensors to acquire patient's health data, which is subsequently transmitted to a local PC station at the patient's home. The local PC runs special application to analyse the acquired data and produces alerts, according to the parameter threshold table (the doctor is able to remotely set the threshold values of the monitored parameters on the patient's PC). In case of any abnormality the findings and the collected vital data are immediately transmitted to the clinic for further analysis. At the clinic, the monitoring module is responsible for monitoring the patient's progress. It encapsulates the rules for manipulating the database and the parametric model of the typical patient, for each disease. Then, using rules related to the monitoring of the patient's progress, the system assigns specific values to the parameters of the model, based on the readings of the patient's sensors.

In this example, two smart services (patient's monitoring and clinical analysis) are combined together to produce a pervasive healthcare service that can provide better recommendations for patient's health. This system may involve some providers such as hospitals, laboratories, pharmacies, and insurance companies. Patients use some sensors and are connected to the system through some communication lines. This healthcare system has the ability to learn from previous patients' health records and context awareness based on patient's vital parameters collected, so it can be classified as SSN.

In this case, if we add new functionality like auto recommendation of drugs to patient, it will cost more on system development and also stimulate a new risk if the recommendation that was produced is wrong. Meanwhile, if we want the system to be safe to use, maybe we have to reduce some functionalities that potentially may cause error in recommendations. These are examples of the trade-off that may occur in the SSN.

4.3 Related Works

In the disciplines of Requirements Engineering and Software Engineering, contradicting or conflicting quality requirements are treated from three perspectives: requirements negotiation, requirements conflict resolution and requirements prioritization (Herrmann and Daneva, 2008). While all perspectives do address trade-offs between quality attributes, the ways through which balance is achieved differs. In turn, there are numerous approaches to reconciling conflicts between aspects of software product quality. Some of the most common include expert judgment, Quality Function Deployment (QFD) and Theory W (Henningsson and Wohlin, 2002). The literature reviewed depicts a number of methods and techniques for use in dealing with software quality trade-offs.

However, there are different opinions as to the source of conflict dependencies. Some authors have stated that conflict is inherent to a pair of quality requirements, while others emphasize that conflict interactions depend on the software architecture and coding (García-Mireles et al., 2013). Both opinions may be true for different requirements of the system. For example, delay and reliability are conflicting, but software architecture and technology choices help to get the required balance of performance. The most common methods reported to support software quality trade-off are the Analytical Hierarchy Process (AHP), model building, the Architecture Trade-off Analysis Method (ATAM), algorithm-based and metric-based methods, expert opinion, Quality Function Deployment (QFD) and prototypes (García-Mireles et al., 2013) (Barney et al., 2012). Additional techniques exist such as goal models and the automated construction of architecture alternatives. However, little empirical support is provided to prove their effectiveness and usefulness when software quality trade-offs are involved.

Furthermore, in the discipline of Software Architecture, a broad range of software architecture trade-off methods have also been proposed and evaluated in order to understand the benefits and shortcomings of each method. (Falessi et al., 2011) compared decision-making techniques at the software design stage, taking into account the difficulties involved in using it. They found that there is no universally best decision-making technique for the resolution of trade-offs in architecture design, it depends on which difficulties, issues, or troubles the architects want to avoid. (Barney et al., 2012) did a systematic map of the software quality trade-off literature to understand the state of the research addressing this area by systematic review. They found that there is an immature approach of software quality trade-off have emerged as candidates to dominate the research space. Only 28% of 168 relevant publications that have been reviewed provide empirical evidence. At 61% of the research provide non-empirically assessed solution proposals. Accordingly, there is no clear approach used for software quality trade-offs.

Nevertheless, there is no approach that I have found to do trade-off analysis among functionality, risk, and cost for SSN. SSN need its own trade-off analysis since it has different characteristics from other systems such as it involves many service providers and clients, can be combined with individual units that are smart, and use a lot of sensors, actuators and communication.

5 METHODOLOGY

To meet the research objective, we will use the principles and guidelines of design science methodology by (Wieringa, 2014) to answer our research questions. It includes four main parts: implementation evaluation/problem investigation, treatment design, design validation and treatment implementation, as shown in Figure 3.

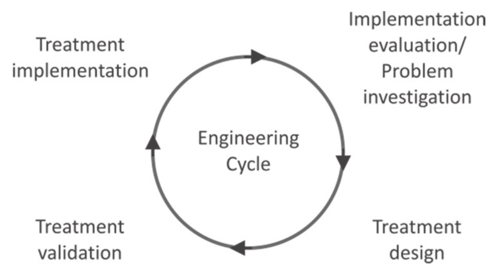

Figure 3: Research methodology used in this research (Wieringa, 2014).

In the context of the work will be conducted in this research, we will perform the following actions for each part:

1. Problem investigation: we have to identify the stakeholders, goals, phenomena and contributions of this research in order to gain a deeper understanding of the problem to be solved. Furthermore, we need to define concepts of smart service network, functionality, risk, cost and relationship among them to get clear objective about the problem definition. These investigations will be carried out by surveying existing literature through systematic literature

review (Kitchenham and Charters, 2007). Observational case studies also needed in this steps to capture mechanisms that produce real-world phenomena. Based on the problem and domain investigation we formulate the criteria and requirements for our solution, which contribute to the stakeholders' goals. Next, we have to determine the best method that suitable with SSN for estimating functionality, risk and cost to be used in calculation to balance the trade-off. This task will be conducted by doing a literature review and accompanied by observational case studies.
2. Treatment design: based on the result of the problem investigation, we will formulate the objective function to calculate the relationship among functionality, risk and cost. This objective function will generate a model to simulate the trade-off optimization. Then based on the simulations and evaluation of the model, we will propose a method to optimize the trade-off that allows us to determine adequate functionality, minimum risk, and reasonable cost provided. In this part, we also need to find out how to use the proposed method to enhance SSN design process.
3. Design validation: to validate the method, we will ask expert opinion and use SSN project in companies to demonstrate the usefulness and the utility of the proposed method. Expert opinion is the simplest way to validate an artefact by asking them about our proposed method whether it can resolve the trade-off problem. If the predicted results do not satisfy our goal, then we have to redesign the method. Therefore, we will use this method in real-life settings to obtain a real picture of the application of this method in the real SSN development project. The lessons learned in this validation will serve to formulate improvements to the method.
4. Treatment implementation: this task is the start of an implementation process to hand over the method to practitioners. This is not part of our research project.

6 EXPECTED OUTCOME

The main expected outcome of this research is knowledge about how to balance the trade-off among functionality, risk and cost required within SSN development. A secondary expected outcome lies in exploring the relationship of functionality, risk and cost in the domain of SSN. Furthermore, by analyzing existing available estimation method, their suitability with SSN and maybe with slight modifications or adjustments, we could propose the best method for estimating such things. Finally, by defining objective function for optimal trade-off and simulating with a model then we can produce a method for balancing the trade-off. This method will allows SSN providers to solve the trade-off problems and provides services with adequate functionality, minimum risk, and reasonable cost.

7 STAGE OF THE RESEARCH

This research will be carried out through four stages as described in Figure. Furthermore, steps to be performed to answer the research questions can be explained in more detail as follows:

1. Problem and domain investigation
   Problem and domain investigation will answers RQ1.1 to RQ2.2 about definitions of functionality, risk and cost, relationships and assumptions of the trade-off, the best method that suitable with SSN for estimating functionality, risk and cost, and optimal balance concept. This stage will be done by literature review and observational case studies.
2. Design a new method
   To answer RQ2.3 and RQ2.5, we need to formulate criteria to decide on an optimal balance. Based on these criteria a model to simulate the trade-off optimization will be generated. Then a method to optimize the trade-off will be proposed and we will try to define an SSN design process enhancement within the method proposed.
3. Validation
   Finally, to answer RQ3.1 and RQ3.2 we will ask expert software engineers to express their opinion about the method in focus groups discussion. The next validation task is to use this method in real projects such as Smart Reasoning Systems For Well-Being at Work and at Home (SWELL) project. Lessons learned from these projects will be used to improve and finalize the method further.

We started this research with a literature review to get familiar with the state-of-the-art and to find characteristics and clear definition of SSN. Based on this review we have found some characteristics of SSN and tried to define SSN comprehensively. Moreover, from reviewed literature, we found that there is no approach exist to do trade-off analysis

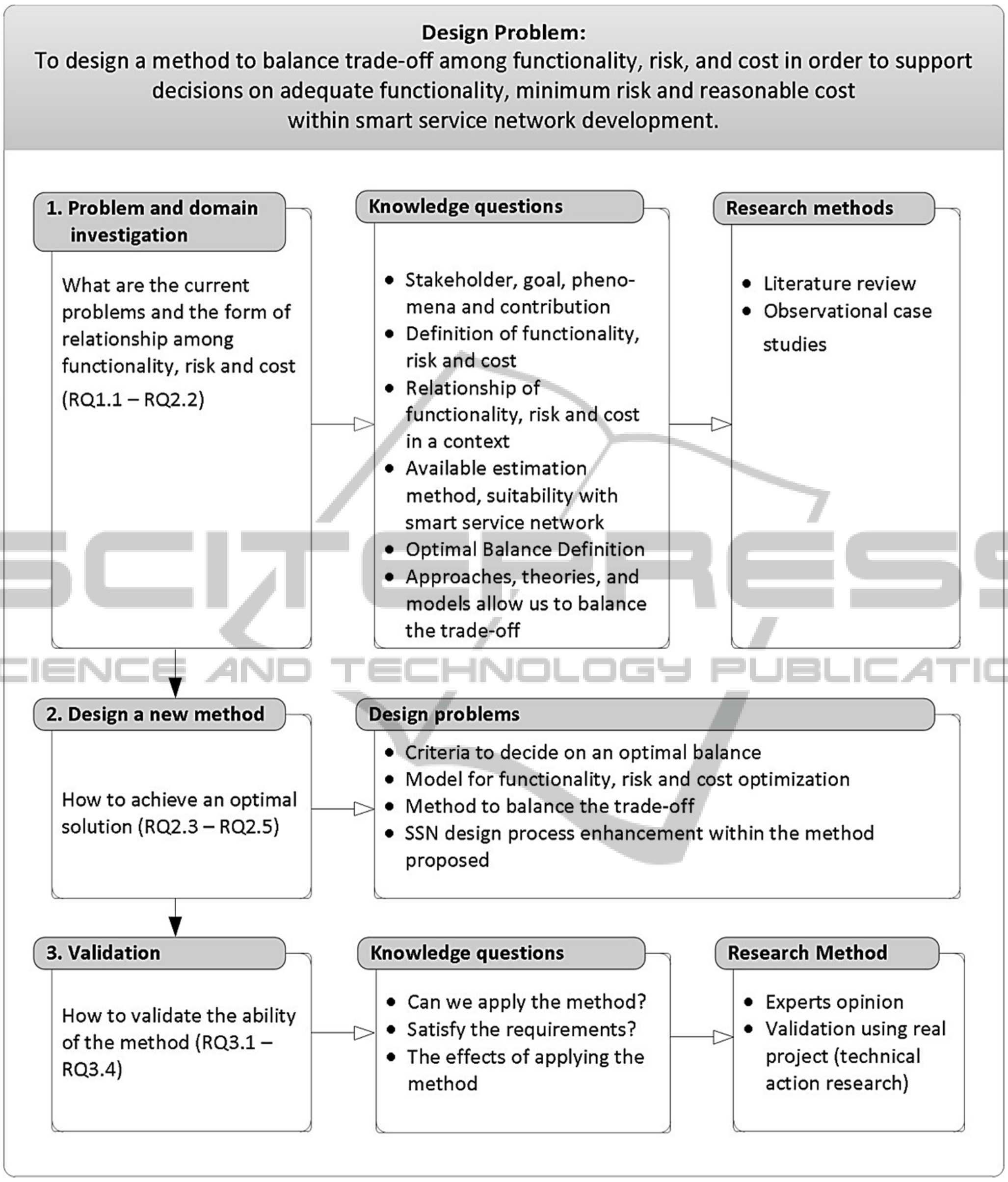

Figure 4: Methodological structure (The boxes mean tasks, white arrows mean task decompositions, and numbers indicate the sequence of tasks will be performed. The black arrows connecting boxes at the left side of the figure indicate sequence of tasks in the design cycle and will be executed from top to bottom).

among functionality, risk, and cost for SSN. SSN need its own trade-off analysis since it has different characteristics from other systems such as it has ability to construct and maintain a model from dynamic context, has ability to learn, and uses sensing devices, machine to machine communication and reasoning system. These characteristics cause the SSN becomes very risky, could even endanger the user if the recommendations produced was wrong.

For next step, we will assess the form of relationship among functionality, risk, and cost and to define an 'optimal solution' for the trade-off in a SSN development. To assess this relationship, we will perform a literature review and observational case studies to validate the results.

8 CONCLUSION

Due to the complexity, heterogeneity, and dynamism, the development of SSN may potentially be fraught with conflicts among functionality, risk and cost. Reasoning about those conflicts and resolving them effectively is an important part of achieving a well-managed SSN development process. This research focused on designing a method to balance functionality, risk and cost in SSN development projects. The method is expected to help the stakeholders to choose and decide on balanced functionality, risks and costs according to their needs. We provided our first working definition of SSN based on a literature survey. We motivated the need for more research in the area and defined our research questions and research process by using design science principles. Our research methodology and planned research stages will translate into more specific research activities as our research is progressing. We started only recently and in the next months we expect .many concepts to become clearer and new challenges to crystallize while the research is advancing.

REFERENCES

Allmendinger, G. & Lombreglia, R. 2005. Four strategies for the age of smart services. *Harvard business review,* 83, 131-+.

Barney, S., Petersen, K., Svahnberg, M., Aurum, A. & Barney, H. 2012. Software quality trade- offs: A Systematic Map. *Information and software technology,* 54, 651-662.

Dos Santos Pacheco, R. C., Salm Junior, J. F., Koller, k. & Schneider, V. 2013. Knowledge and ontology engineering for smart services.

Falessi, D., Cantone, G., Kazman, r. & Kruchten, P. 2011. Decision- making techniques for software architecture design: a comparative survey. *Acm computing surveys (csur),* 43, 1-28.

García-mireles, G., Moraga, M., García, F. & Piattini, M. 2013. A Framework to support software quality trade-offs from a process-based perspective. *In:* Mccaffery, f., O'Connor, r. & Messnarz, r. (eds.) *Systems, software and services process improvement.* Springer berlin heidelberg.

Henningsson, K. & Wohlin, C. Understanding the relations between software quality attributes – a survey approach. 12th international conference on software quality, 2002 ottawa, canada. Asq software division.

Herrmann, A. & Daneva, M. 2008. Requirements prioritization based on benefit and cost prediction: an agenda for future research.

Kitchenham, B. A. & Charters, S. 2007. Guidelines for performing systematic literature reviews in software engineering. *Technical report ebse-2007-01.* Version 2.3 ed. Keele, uk: ebse.

NG, J. W., Chignell, M., Cordy, J. R. & Yesha, Y. 2010. Smart services. *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics),* 6400, 173-177.

OECD 2013. Building blocks for smart networks. *Oecd digital economy papers.* OECD.

Stroulia, E. 2010. Smart services across the real and virtual worlds. *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics),* 6400, 178-196.

Van den heuvel, w. J. & papazoglou, m. P. 2010. Toward business transaction management in smart service networks. *IEEE internet computing,* 14, 71-75.

Villegas, N. M. & Müller, H. A. 2010. Managing dynamic context to optimize smart interactions and services. *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics),* 6400, 289-318.

Wang, Y., Taher, Y. & Van den Heuvel, W. J. 2012. Towards smart service networks: An interdisciplinary service assessment metrics.

Wieringa, R. J. 2014. Design science methodology for information systems and software engineering. *Springer*.