Impact of Requirements Elicitation Processes on Success of Information System Development Projects

Līga Bormane[1], Jūlija Gržibovska[2], Solvita Bērziša[3], Jānis Grabis[4]
[1–4] *Riga Technical University*

Abstract – **Requirements articulating user needs and corresponding to enterprise business processes are a key to successful implementation of information system development projects. However, the parties involved in projects frequently are not able to agree on a common development vision and have difficulties expressing their needs. Several industry experts have acknowledged that requirements elicitation is one of the most difficult tasks in development projects. This study investigates the impact of requirements elicitation processes on project outcomes depending on the applied project development methodology.**

Keywords – **Agile programming methodology, business requirements, requirement elicitation, stakeholders, user requirements analysis, waterfall model.**

I. INTRODUCTION

In the modern age of information technology (IT), any human activity is associated with one or more information systems (ISs). Ensuring effective support of business process is also vital for the company. Since the efficiency of business process has become one of the cornerstones of enterprise competitiveness, the IS is an integral part of everyday life. Each new activity, each new product and each new project are initiated in response to some of the business needs. Quite often there is a situation that huge consumption of time and other resources is despised, needs are not satisfied and no appropriate solution is found for the company.

According to the Standish Group data, from year to year it is difficult to make IT projects successful [2], [8]. Approximately 20 % of the projects are totally unsuccessful, the implementation of about 50 % of the projects is delayed, budget exceeds or all necessary requirements are not implemented, and only 30 % of projects are successful – meet the original budget, time and are fully functional [2], [8]. Another important factor is that since the 1990s one of the TOP 3 reasons of project failures has been related to the project requirements [5], [8], [10].

The most common reason of project failures is the inadequate attention to user requirements analysis of the developed system. Approximately 50–60 % of the project failures are shoddily made inquiries of requirements [7]. If the customer is not sufficiently involved in defining requirements or the analyst takes into account only the customer requirements, but the customer is unable to correctly formulate requirements meeting the business needs, then the quality of defined requirements is low [2] [3], [4]. If requirements are weak and insufficiently defined at the analysis stage, with each next stage of the project the amount of work needed to be done increases in order to fix

the errors – around 40 % of the work at the design stage and around 70 % at the implementation stage [7].

User requirements elicitation process is an important part of the project requirements analysis that ensures future product compliance or non-compliance with the end user expectations [1], [6]. The elicitation process is one of the most difficult tasks of development projects because it involves understanding of client natural language, correct collection of client problems, needs and expectations, as well as requirement conversion in a natural language understood by the system developers and ensuring a correct transition between the client and the system developer [6], [9].

Although since the 1990s several popular methods have been developed for the requirements elicitation process support [1], the problem of correct method application has not been solved. A number of studies [11] state that 43 % of project failures have been caused by applying inappropriate requirements elicitation methods.

Replacing a well-known waterfall method with the agile model has been promoted as one of the solutions to the project failure problem. Agile approaches distribute all the common project requirements in small parts and accomplish them gradually. It helps reduce the pressure on identification of all system requirements in one period and increases the possibility to correct mistakes at an earlier stage of the project than using a waterfall model but agile does not solve the problems associated with requirements elicitation. Projects managed by the agile methodology tend to come to a standstill if the developer failed to establish contact with the customer and correctly define the software system requirements [12].

The objective of this study is to investigate the impact of requirements elicitation processes on project outcomes depending on the applied project development methodology. In this study, the authors draw attention to the role of qualitative requirements elicitation process in the successful implementation of IS project and reflect the results of case study research, which was conducted in one of the largest and most valuable companies of Latvia. During the study, the authors have carried out a detailed analysis of some company's development projects with an objective to evaluate if mistakes in the requirements elicitation process have an equally significant impact on the project success in different software development (SD) methodologies.

The rest of the paper is structured as follows: Section 2 describes theoretical background and the related studies; Section 3 presents the research methodology used for a case study. Results of the case study are given and discussed in

Section 4. Conclusion and future research are presented at the end of the paper.

II. Background and Related Studies

This section summarises theoretical background and related studies about the successful project, requirements elicitation, SD methodologies and the requirements elicitation process in different methodologies.

A. The Notion of Successful Project

In order to clarify when a project can be called successful, it is necessary to provide the project definition. Different explanations of the concept "project" have been found in different literature sources related to project management. According to these definitions, the project is a set of actions to be implemented for a certain period of time with an objective to create a new compliant product with originally intended money and human resources.

Taking into account the definition of project, it can be stated that the success of the development project depends on the interaction of three project limiting factors – time, quality and cost. Thus, a "successful" project is a project that has been implemented within the specified time, approved budget and to a level of functionality that meets the determined needs. Even if one factor is not considered, it will lead to project failure, as well as it will definitely affect the other two elements. For example, by improving the project quality the costs of the project also increase and the implementation time is extended [24].

According to the study carried out by the Project Management Institute in 2014 [26], most organisations underestimate the importance of the quality limit on project success. The requirements management process that is primarily responsible for the end-product quality as a critical factor has been referred in half of the reviewed unsuccessful projects. On average, only in one-third of analysed cases requirements management has been carried out properly (Fig. 1).
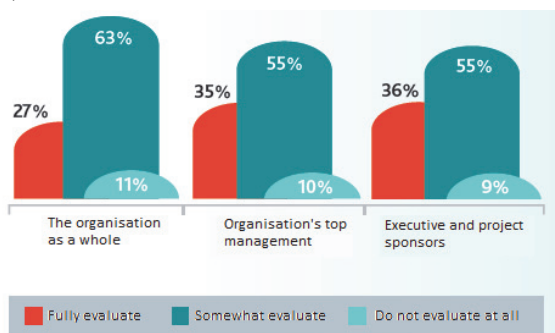

Fig. 1. Evaluation of requirements management as a critical component for project and strategic initiatives by the majority of organisations [26].

B. Requirements Elicitation

Assumed decisions and cooperation with stakeholders in the requirements analysis determine up to 80 % of the final developing system costs [9]. Therefore, the mistakes at this stage of project will have a significant influence on the project outcome. Requirements analysis aims at identifying all the client stakeholders, finding out their needs, experience, view point on the system and recording them in such a way as to be able to implement the correct development [13].

During the first step of requirements analysis (requirements elicitation process), system requirements need to be discussed and coordinated with the stakeholders [9], [15]. In this step, it is important to understand that each of the interested parties is a different person with their needs, vision of a solution, previous experience, as well as prejudice [9]. Requirements elicitation is a complex process because it ensures searching, learning, acquisition and development [15].

The requirements elicitation and related problems [9], [14] have been widely studied in both literature sources – scientific and practical ones. The topicality of the study confirms a growing trend of available articles in scientific databases (Fig. 2).
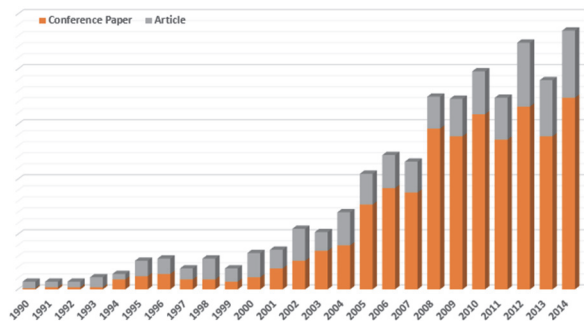

Fig. 2. Number of conference and scientific papers on requirements elicitation from 1990 to 2014.

Requirements elicitation problems have been divided into three main categories: problems of scope, problems of volatility and problems of understanding. In literature, it has been noted that problems of scope are related to amount of information included in requirements description, i.e., too much or too less information. Problems of volatility are concerned with the changing nature of requirements. Problems of understanding are related to poor understanding of requirements, lack of communication, lack of domain knowledge and conflicting views of users on requirements [14].

The requirements elicitation process includes five principal types of activities [16]:

1) *Understanding of the application domain:* "What are the purposes?" and "How does it work?" are the first questions to answer.

2) *Identifying the sources of requirements:* Multiple sources can be used, such as customers, users or experts. Each of them provides a different kind of information.

3) *Analysis of stakeholders:* Not all of the stakeholders have equal importance and impact that is why in this step it is necessary to identify only the most important of them and keep them in the project scope.

4) *Selecting techniques, approaches and tools:* This step has often been considered as a critical factor for success of the requirements elicitation process. This study has been performed to help choose the "right" techniques according to the situation.

5) *Eliciting the requirements of stakeholders and other sources.*

There are several methods that can be used to obtain as much information as possible from the stakeholders. High quality requirements can be collected only if you have selected the right people and the right method for them. The collected requirements can also be compared and prioritised after they have been obtained from users. There is no ideal method that works in all situations. The suitable method for the requirements elicitation process can be selected according to the project type and stakeholders [15].

In practice, all methods (techniques) are divided into the following groups: traditional, collaborative, cognitive, observational techniques, etc. [17], [18], [19]. **Traditional techniques** are interview, questionnaire, data collection from an existing system, survey. **Collaborative techniques** – focus group, brainstorming, prototyping, workshop, story boarding, models, use cases and scenarios. **Cognitive techniques** are document analysis, protocol analysis, laddering and repository grid. **Observational techniques** are observation and ethnography/social analysis.

If the results of studies on effectiveness of the requirements elicitation methods [6], [18], [19] and [25] are compared (Table I), the user survey and interview are recognised as the most effective methods.

TABLE I

THE MOST EFFECTIVE REQUIREMENTS ELICITATION METHODS

METHOD	[6]	[18]	[19]	[25]
Questionnaire	x	x	x	
Use case diagram	x			
Brainstorming	x			
Interview		x	x	x
Requirement reuse		x		
Document analysis			x	
Scenario, passive storyboard			x	
Requirements management	x			
Prototyping		x		

In another study [20], it has been considered how to choose the best method for different agile methodologies (Table II). In product backlog, all requirements are deemed as necessary or useful and the priority of requirements is identified. System functions, possible errors and improvements are included in the product backlog, while in the feature list only system functions are included and the project team arranges its priorities. [20]

TABLE II

REQUIREMENTS ELICITATION METHODS FOR AGILE METHODOLOGIES

XP	SCRUM	AGILE PROTOTYPING	FDD
Use case Interview Brainstorming	Product backlog Interview	Brainstorming	Feature list Interview

Comparing results of Table I and Table II, it has been evident that an interview can be equally effectively applied in both SD methodologies.

C. Software Development Methodology

The traditional or waterfall model (Fig. 3) divides SD into a number of successive activities or stages where the result of each previous activity is input for the next activity proceeding

[21]. The agile methodology (Fig. 4) is based on iterative steps, i.e., the entire SD project is divided into small successive stages [21]. The stage plan is defined or specified at the beginning of each stage (at this point a plan can be adjusted to the changing requirements of the client) and during the stage it cannot be changed [21].
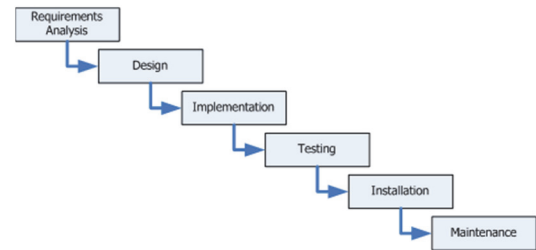

Fig. 3. Traditional or waterfall life cycle model [21].
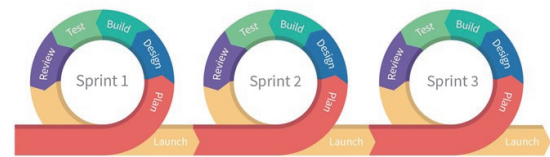

Fig. 4. The life cycle model of agile programming methodology [21].

In the waterfall model, requirements are defined at the beginning of the project and necessary changes can be authorised only after some development and testing activities. Otherwise in the project based on iterative principles (iterative, incremental, agile, etc.) requirements are defined and changes are accepted in each iteration (Fig. 5) that increases flexibility of these models against mistakes in the requirements elicitation process [22].
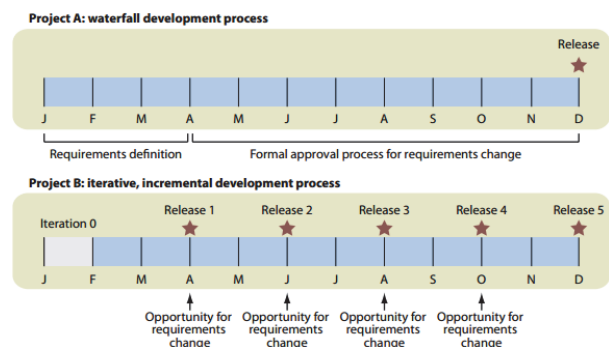

Fig. 5. Two models for managing changes in requirements [22].

As one of the main reasons why the agile methodology is much more efficient is the already mentioned small feedback loop between the idea generation and implementation time. This not only reduces the risk of confusion but also reduces costs of mistake resolution [23]. The cost curve of both SD methodologies, depending on a phase when changes should be carried out (Fig. 6), shows that a waterfall model is much more sensitive against the requirements quality. The cost curve of waterfall model projects is growing exponentially – the later the defect is found, the more expensive it will be to fix it. The main reason is that the definition and analysis requirements take

place at the beginning of the project (Fig. 5). In the agile model, this curve increases almost logarithmically till a specific point and then a little bit faster. In the ideal case, the cost curve should be logarithmical throughout the entire period of project implementation.
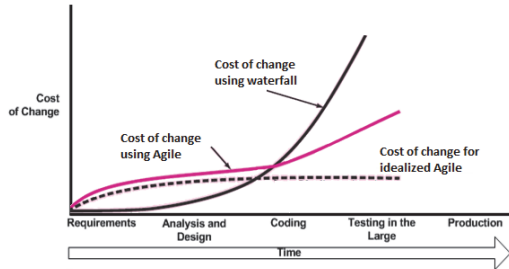

Fig. 6. Cost change curve of traditional and agile methodology [23].

According to the Standish Group research [8], overall success of implementation of small-scale projects is almost equal for both waterfall and agile models (Fig. 7). Comparing project results with regard to the project size, the most successful projects are small-scale (Fig. 8). These projects have simpler project vision, less amount of work, less time to spend for requirements collection, as well as it is possible to achieve the results faster. Several companies have confirmed that applying project optimisation by dividing a large-scale project into some smaller ones the project success has clearly been increased.
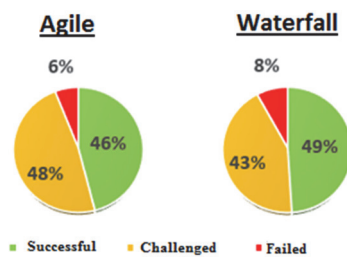

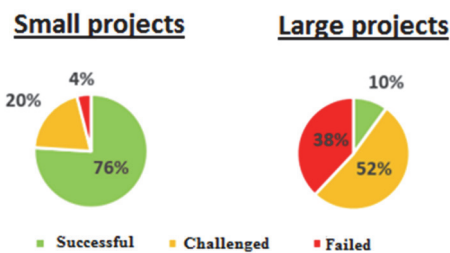Fig. 7. The success of agile and waterfall methodology in small projects [8].

Fig. 8. The outcome of small and large projects [8].

III. Research Methodology

Our research is based on the case study methodology. This methodology is useful for observing, explaining and exploring real-life events. Case study involves an in-depth inspection of a small number of cases.

A. Case Study Questions

One of the most important elements of modern life is money. Money often plays a key role in deciding on the project implementation or cancellation. According to the theoretical cost curves of changes, the agile model is better than the waterfall one. Since the number of project changes directly depends on the quality of the requirements elicitation, in our study the first raised question is related to the costs of changes.

Q1: Does the real character of the cost curve depend on the selected SD methodology?

According to the Standish Group, a more significant project indicator is its size. The theory about small-scale development projects in both waterfall and agile models says that it is almost equally effective because for these projects the total number of requirements is also relatively small; requirements are more easily to elicit and collect before development. The importance of the project size is the second question:

Q2: Is it true that in small-scale SD projects both methodologies equally affect the consumed project person-days and cost?

As already mentioned, high-quality requirements can be obtained only by selecting the requirements elicitation methods that are suitable for project stakeholders. After the analysis of related research, an interview has been referred to as the most appropriate and popular requirements elicitation method of all SD methodologies. In our study, we also included a question about the use of the elicitation methods:

Q3: What requirements elicitation methods are used in real projects? Do they differ in various SD methodologies?

B. Case Selection

The case study research has been conducted in one of the largest and most valuable companies of Latvia. Its activities cover the all territory of Latvia, and it employs approximately 1,000 employees. None of the company's activity fields are directly related to IS development, but as already mentioned the company cannot survive without modern IT support. Therefore, every year this company has several different challenges of SD projects that are related to the introduction of new systems and existing system supplementation.

External service providers mainly ensure the implementation of projects within the company. At the company, there is a separate established structure that provides each project management and business analysis function.

C. Data Collection and Analysis

To answer the questions, the essential characteristics of the SD project have been identified. For data collection, three data collection and analysis steps have been used:

1) usage of multiple information sources to obtain more complete data and increase the reliability of the data;
2) establishment of data registration place – common table, database or other safe place for data storage;
3) testing of resulting data completeness and authenticity – review of all data and search for contradiction.

For collection of case data, the following sources of information available to the company have been used:

- closed tasks of SD project;
- project progress reports;
- project issue management system – records about change requests;
- project final reports.

All available information (paper and electronic data results) about projects has been recorded in the summary table. The authors of the study have also carried out interviews with projects leaders to obtain and capture the undocumented data.

IV. RESULTS

A. Case Study Description

12 different SD projects have been examined during this study performed at the company during the period from 2014 to 2016 (inclusive). Most projects are related to the company's existing IS replenishment and only two are new SD projects. The examined SD project includes the following systems: enterprise resource planning and management system, document management system, budget planning and management system, performance management system, employee self-service system and system of vehicle registration.

The examined projects have been implemented using 2 different SD methodologies – waterfall (7 projects) and agile (5 projects). In each project, a number of stakeholders are greater than 3 and the total number of users exceeds 30 people.

In the examined projects, different teams have worked: project managers, analysts and developers. The overall performance of 4 different teams of developer has been analysed.

B. Data Analysis

During the research, the summary table has been created illustrating the obtained information about projects (Table III), i.e., project methodology, applied elicitation methods, project size, implementation year, complexity, amount of actual time: person-days and budget, as well as changes in budget breakdown by the project stage. As the information on person-days and budget is confidential, it is not included in Table III but displayed in graphs below without certain values.

Using the collected information about project changes, the authors have created cost change curves of each SD methodology (Fig. 9), compared planned and actual project capacity (Fig. 10), as well as planned and actual budget by means of the applied methodology (Fig. 11).

C. Responses to Case Study Questions

Q1: Does the real character of the cost curve depend on the selected SD methodology?
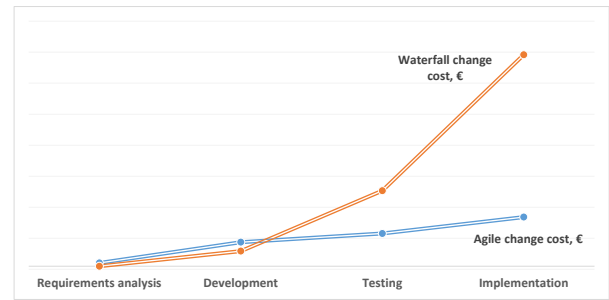

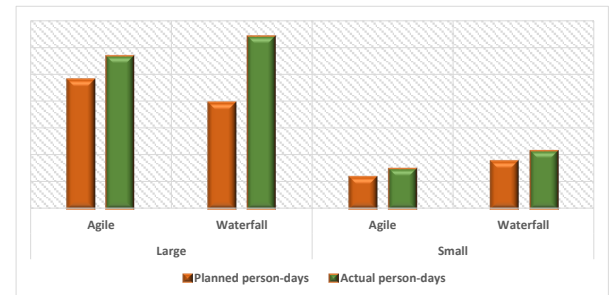Fig. 9. Cost change curves of case study projects.

Fig. 10. Planned and actual person-days of case study projects.

Fig. 11. Planned and actual budget of case study projects.

TABLE III
SUMMARY OF CASE STUDY PROJECTS

PRO-JECT	DEVE-LOPER	METHODOLOGY	ELICITATION METHOD	PROJECT SIZE	YEAR	COMPLE-XITY
P_1	A	Waterfall	Interviews, Old system study	Large	2014	High
P_2	A	Agile	Interviews, Brainstorming	Small	2016	Low
P_3	A	Waterfall	Interviews, Prototyping	Small	2016	Low
P_4	A	Agile	Interviews, Document analysis	Large	2016	Medium
P_5	A	Waterfall	Interviews, Old system study	Small	2015	Medium
P_6	B	Waterfall	Interviews, Old system study	Large	2015	High
P_7	B	Agile	Interviews, Old system study	Large	2015	High
P_8	B	Agile	Interviews, User story	Large	2015	Medium
P_9	C	Agile	Interviews, User story	Small	2015	Medium
P_10	C	Agile	Interviews, Prototyping	Small	2014	Medium
P_11	D	Waterfall	Interviews, Analysis of legal norms	Small	2016	Low
P_12	B	Agile	Interviews, Prototyping	Large	2015	Medium

The cost curve of the case study proves the theory that the change cost in waterfall model rises almost exponentially and in the agile model almost logarithmically. This confirms the above-mentioned assumptions:

- Later change implementation increases costs regardless of the SD methodology;
- Projects implemented by the waterfall model are much more dependent on high quality of the requirements elicitation process;
- The cost curve of agile SD projects increases faster if requirements have not been identified at the analysis phase but later surcharges are relatively small.

Q2: Is it true that in small-scale SD projects both methodologies equally affect the consumed project person-days and cost?

Analysing results of planned and actual values of project capacity and budget (Fig. 10 and Fig. 11), depending on the project size and selected SD methodology it can be concluded that the outcome of a small-scale waterfall and that of an agile project are quite close. Using the agile methodology in large-scale projects, its advantage has been observed – the amount of resources and consumed money is much closer to the planned one than using the waterfall methodology. Therefore, this confirms two previous assumptions:

- Small-scale projects are easy to implement complying with initially defined constraints compared to large-scale projects irrespective of the SD methodology;
- Large-scale project optimisation (dividing it into several small stages) can be used to improve overall project execution.

Q3: What requirements elicitation methods are used in real projects? Do they differ in various SD methodologies?

The collected data have also proven that a user interview is the most common requirements elicitation methods at a company; it is evident regardless of the applied SD methodology (Fig. 12). It can be explained by the fact that this elicitation method is one of the oldest and most popular methods. This method is also customary and easier for users.
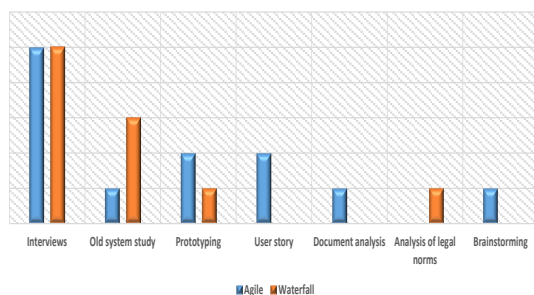

Fig. 12. Requirements elicitation methods used in case study projects.

The interview method is mostly used in both methods and in each SD methodology separately. This confirms one more assumption:

- Interview can be equally effectively applied in the waterfall and agile methodologies.

Within this case study only 12 projects implemented by the company have been analysed, and the information about the requirements elicitation directions can be provided. However, the obtained data set seems to authors too small to discuss the requirements elicitation methods used by the company. The application of different methods is evident in the waterfall and agile methodologies, but it can be related to the knowledge and working practices of developers and analysts.

V. CONCLUSION, DISCUSSION AND FURTHER RESEARCH

A. Compliance of Results with Theory and Related Research

The obtained results on the success of project, applied requirements elicitation methods in different SD methodologies have proven the generally accepted theory. The authors have concluded that the case study confirmed the assumptions made during the analysis of other case studies. Overall, on the basis of the research results, the authors have gained the approval of 6 rules that have already been mentioned in the case study. The approval of these rules allows concluding the following:

- Regardless of the selected SD methodology, changes will always take place in projects. In addition, the introduction of changes will be more expensive further away from the requirements analysis phase. Thus, it makes the requirements analysis phase, especially requirements elicitation, a key process of successful project implementation.
- The budget is much more subject to requirement changes in waterfall projects than in agile ones. Thus, waterfall projects are more sensitive to mistakes made during the requirements elicitation process and their correction.
- Not always the project success depends only on the selected SD methodology and quality of the requirements elicitation. The total amount of work also plays a crucial role. If the project has small-scale capability or a large-scale project is optimised, the successful outcome of the project is much easier to be achieved.
- User interviews can be equally effective in any of the SD methodologies.

B. Implications

Even with so small number of projects covered in the case study, the authors have confirmed the initial assumptions, and this once again confirms the topicality of the research.

Searching for information on the case study theory, the authors have faced the problem that there are a few studies devoted to the following topics:

- How to improve stakeholders' preparedness for the conversation with a developer – how to properly formulate their needs and highlight the basic formalities, what should be mandatory?
- Should the requirements elicitation methods be chosen according to the developer's knowledge and experience or what is the initial stage?
- What is the role of "bridge person" between business people and developers? How a "bridge person" can affect project outcome?

It should be noted that a relatively few success stories have been identified. However, most papers were devoted to failed projects. There was no information found about successful projects and the applied methods.

C. Limitations

As it has already been mentioned, the case study has covered only 12 SD projects implemented within the same company. This gives information about directions of the applied requirements elicitation methods at the company under analysis, but it is too general to identify common trends in Latvia. More projects are needed to be analysed.

REFERENCES

[1] H. Liao, "Requirement elicitation of Enterprise Informationization from view of VCA," in *INC2010: 6th International Conference on Networked Computing*, Gyeongju, Korea (South), 2010, pp. 390–395.

[2] R. Stanley and L. Uden, "Why projects fail, from the perspective of service science", *7th International Conference on Knowledge Management in Organizations: Service and Cloud Computing,* (Advances in Intelligent Systems and Computing). Berlin, Germany: Springer, vol. 172, 2013, pp. 421–429. https://doi.org/10.1007/978-3-642-30867-3_38

[3] A. Przybyłek, "A business-oriented approach to Requirements Elicitation," in *ENASE 2014 – Proceedings of the 9th International Conference on Evaluation of Novel Approaches to Software Engineering*, 2014, pp. 152–163.

[4] S. Liu, B. Wu and Q. Meng, "Critical affecting factors of IT project management," in *ICIII 2012 – Proceeding of 2012 International Conference on Information Management, Innovation Management and Industrial Engineering*, Sanya, 2012, pp. 494–497. https://doi.org/10.1109/ICIII.2012.6339710

[5] I. Attarzadeh and H. O. Siew, "Project management practices: Success versus failure," in *ITSim 2008 - Proceedings of International Symposium on Information Technology 2008,* Kuala Lumpur, 2008, pp. 1–8. https://doi.org/10.1109/itsim.2008.4631634

[6] C. P. Noraini and M. Z. Abdullah, "Requirement elicitation: identifying the communication challenges between developer and customer," *International Journal on New Computer Architectures and Their Applications*, pp. 371–383, 2011.

[7] B. R. Shubhamangala, L. M. Rao, A. Dakshinamurthy and C. G. L. Singh, "Ability based domain specific training: A pragmatic solution to poor requirement engineering in CMM level 5 companies," in *CSAE 2012 – Proceedings, 2012 IEEE International Conference on Computer Science and Automation Engineering*, Zhangjiajie, 2012, pp. 459–464. https://doi.org/10.1109/CSAE.2012.6272993

[8] The Standish Group, CHAOS Report, 2014. [Online]. Available: https://www.versionone.com/assets/img/files/CHAOSManifesto2013.pdf

[9] A. Azadegan, K. N. Papamichail and P. Sampaio, "Applying collaborative process design to user requirements elicitation: A case study," *Computers in Industry*, pp. 798–812, Sept. 2013. https://doi.org/10.1016/j.compind.2013.05.001

[10] G. Stepanek, "Software project secrets: Why software projects fail," in *Software Project Secrets: Why Software Projects Fail*, pp. 1–166, 2005. https://doi.org/10.1007/978-1-4302-0055-0

[11] D. Siahaan and F. Irhamni, "Advanced methodology for requirements engineering technique solution (AMRETS)," *International Journal of Advancements in Computing Technology*, vol. 4, issue 5, pp. 75–80, March 2012.

[12] E. Colonese, "Agile: The human factors as the weakest link in the chain," *Proceedings of 4th International Conference in Software Engineering for Defence Applications* (Advances in Intelligent Systems and Computing). pp. 59–73, 2016. https://doi.org/10.1007/978-3-319-27896-4_6

[13] Shams-Ul-Arif, Q. Khan, S. A. K. Gahyyur, "Requirements engineering processes, tools/technologies, & methodologies," *International Journal of Reviews in Computing*, pp. 41–56, 2009–2010.

[14] S. N. Kumari and A. S. Pillai, "Requirements elicitation issues and project performance: A test of a contingency model," in *SAI 2015 – Proceedings of the 2015 Science and Information Conference*, London, 2015, pp. 889–896. https://doi.org/10.1109/sai.2015.7237247

[15] M. Yousuf and M. Asger, "Comparison of various requirements elicitation techniques," *International Journal of Computer Applications*, vol. 116, issue 4, pp. 8–15, Apr. 2015. https://doi.org/10.5120/20322-2408

[16] C. Mauger, T. Schwartz, J.–Y. Dantan and L. Harbouche, "Improving users satisfaction by using requirements engineering approaches in the conceptual phase of construction projects: The elicitation process," in *IEEM2010 – IEEE International Conference on Industrial Engineering and Engineering Management*, Macao, 2010, pp. 310–314. https://doi.org/10.1109/IEEM.2010.5674471

[17] S. Tiwari, S. S. Rathore and A. Gupta, "Selecting requirement elicitation techniques for software projects," in *CONSEG 2012 – 2012 CSI 6th International Conference on Software Engineering*, Indore, 2012, pp. 1–10. https://doi.org/10.1109/conseg.2012.6349486

[18] W. J. Lloyd, M. B. Rosson and J. D. Arthur, "Effectiveness of elicitation techniques in distributed requirements engineering", *Proceedings of the IEEE International Conference on Requirements Engineering*, 2002, pp. 311–318. https://doi.org/10.1109/ICRE.2002.1048544

[19] S. G. Gunda, "Requirements engineering: elicitation techniques," M.S. thesis, Department of Technology, Mathematics and Computer Science, University West, Sweden, 2008.

[20] W. Helmy, A. Kamel and O. Hegazy, "An evaluation framework for requirements elicitation in Agile methods," *ICSEA 2012 - The Seventh International Conference on Software Engineering Advances*, Lispon, Portugal, 2012, pp. 588–593.

[21] S. Balaji and M. S. Murugaiyan, "Waterfall vs v-model vs agile: a comparative study on sdlc," *International Journal of Information Technology and Business Management,* vol. 2, no. 1, pp. 26–30, June 2012.

[22] Forrester Research, The Root of The Problem: Poor Requirements, 2006. [Online]. Available: http://www.techworld.com/resources/white-paper/it-management/root-problem-poor-requirements-5104/

[23] D. Duka, "Agile experiences in software development," *MIPRO 2012 – 35th International Convention on Information and Communication Technology, Electronics and Microelectronics – Proceedings*, Opatija, 2012, pp. 692–697.

[24] M. Hasanzadeh, Z. Tavakolirad and P. Abbasi, "Review of affective factors on cost, time and quality of construction projects in developing countries," *ICEMMS 2011 – Proceedings: 2011 2nd IEEE International Conference on Emergency Management and Management Sciences*, Beijing, 2011, pp. 858–861. https://doi.org/10.1109/icemms.2011.6015818

[25] B. Davey and C. Cope, "Requirements elicitation – what's missing?" *Issues in Informing Science and Information Technology*– vol. 5, pp. 543–551, 2008.

[26] Project Management Institute 2014, PMI's Pulse of the Profession: Requirements Management — A Core Competency for Project and Program Success. [Online]. Available: http://www.pmi.org/learning/pulse.aspx

[27] S. Sheppard, "The Principles of Product Development FLOW: Second Generation Lean Product Development," 2016. [Online]. Available: http://labs.blogs.com/its_alive_in_the_lab/2016/04/book-the-principles-of-product-development-flow-second-generation-lean-product-development-by-donald.html

Līga Bormane holds a Bachelor's degree in Computer Science and IPMA Certified Project Manager (Level C) certificate. This is her first publication, but her main interests are project requirements analysis phase, problems related to requirements analysis phase and role of "bridge person" between business people and developers. She works as a Project Manager at one of the largest companies of Latvia.
E-mail: Liga.Bormane@edu.rtu.lv

Jūlija Gržibovska holds a Bachelor's degree in Electrical Engineering. At the moment, she is studying at the professional Master study programme "Information Technology" at Riga Technical University. This is her first publication. Her main research interest is related to requirements elicitation and its methods. She works as a System Analyst at one of the largest companies of Latvia.
E-mail: Julija.Grzibovska_1@edu.rtu.lv

Solvita Bērziša holds a Doctoral degree and is an Assistant Professor and Researcher at the Institute of Information Technology of Riga Technical University (Latvia). She obtained her Dr. sc. ing. (2012), B.Sc. (2005) and M.Sc. (2007) degrees in Computer Science and Information Technology at Riga Technical University. Her main research fields are IT project management, implementation and application of project management information systems, as well as project data analytics. She also works as an IT Project Manager at Exigen Services Latvia. She holds a PMP and CBAP certificate. She is a member of PMI and Latvian National Project Management Association.
E-mail: Solvita.Berzisa@rtu.lv

Jānis Grabis holds a Doctoral degree and is a Professor at Riga Technical University (Latvia) and the Head of the Institute of Information Technology. His main research interests lie within the application of mathematical programming methods in information technology, enterprise applications and system integration. He has published more than 60 scientific papers, including a monograph on supply chain configuration. He has led a number of national projects and has participated in five projects in collaboration with the University of Michigan-Dearborn (USA) and funded mainly by industrial partners, such as SAP America and Ford Motor Company.
E-mail: Grabis@rtu.lv