Using Non-functional Requirements Templates for Elicitation: A Case Study

Sylwia Kopczyńska
Institute of Computing Science
Poznan University of Technology
ul. Piotrowo 2, 60-965 Poznań, Poland
sylwia.kopczynska@cs.put.poznan.pl

Jerzy Nawrocki
Institute of Computing Science
Poznan University of Technology
ul. Piotrowo 2, 60-965 Poznań, Poland
jerzy.nawrocki@cs.put.poznan.pl

***Abstract*—It is still an open question how to achieve a proper balance between the cost and value of requirements elicitation. When deciding to assign time and other resources to elicitation, one needs to know what its effectives would be. In the paper we investigate the Structured Elicitation of Non-functional Requirements(SENoR) method. The method is composed of a sequence of short brainstorming sessions driven by ISO25010 quality characteristics. It uses Non-functional Requirements Templates (NoRTs) to support the elicitation process. Our exploratory case study on cost and effectiveness of SENoR and NoRTs included 7 projects that developed tailor-made web applications. The findings show that two 1.5-hour elicitation workshops can result in non-functional requirements of stability at the level of 80%.**

I. Introduction

In the era of the waterfall model, the necessity of thorough requirements analysis and of completeness of software requirements specification were one of the most important issues (e.g., IEEE 830 [1]). These needs were relaxed by agile methodologies which simplified RE processes to make quick, short releases delivering working software possible. Agile approaches (e.g., XP [2] and Scrum [3]) introduced user stories and use cases as the main tools to document requirements. However, such an approach focuses on system functionality and may lead to the omission of the requirements regarding *how* the functions provided by the system should be executed. According to the results of the investigation of agile projects carried out by Cao and Ramesh [4], customers often "focus on core functionality and ignore NFRs". Yet, XP recommends to use automated test cases instead of documentation [2]. But it may also be insufficient to communicate non-functional requirements (NFRs), especially in the early software development stages when there is little known about the system architecture. So, people often ignore or poorly define NFRs especially in early development stages [4].

Regardless of project management methodology, it is widely known that neglecting NFRs may have severe consequences, as in the cases of Therac-25 [5], the Airline 5 accident [6] etc., or it may lead to excessive refactoring. As some researchers noticed, what is needed is a balance between discipline and agility [7]. At the level of non-functional requirements this need translates to the challenge of how to achieve a proper balance between the cost and value of formulated requirements [8]. One of the obstacles in answering this question is the fact that there is still little known about the cost and value of using different requirements engineering (RE) methods. In particular there is scarcity of information about elicitation process in real-life projects [9]. So, deciding on using a certain elicitation method, assigning time and other resources is difficult. Although there exists a number of methods and frameworks which provide a solid basis for dealing with non-functional requirements, e.g., NFR Framework [10], KAOS [11], they are regarded as too heavy-weight, especially for the agile context [12]. Moreover, natural language still remains the preferred mode for representing requirements [13], even though it is inherently ambiguous. Thus some researchers and practitioners introduced methods which aim to positively impact at least one of the mentioned factors of requirements elicitation. The methods promotes knowledge reuse, e.g., patterns of Withall [14], the PABRE method [15]. However, in order to answer the open question of how to make elicitation quick, and get requirements of high quality [16], there is a need for more information about the cost and effectiveness of different methods.

Therefore in this paper we provide results of our case study on elicitation of NFR with the use of templates (Section IV). Templates of NFRs on one hand aim to provide expert knowledge and lessons learned, and help formulating NFRs. On the other hand they act as a checklist, and enhance word association. Templates are sometimes used as a solution of an NFR's pattern, e.g., patterns of Withall [14] (in Section V we refer also to other elicitation approaches). We use them in our method, called SENoR (Structured Elicitation of Non-functional Requirements), to elicit as much information regarding NFRs from various project stakeholders during a workshop. The method is composed of a sequence of short brainstorming sessions driven by the ISO25010s quality characteristics [17]. During each brainstorming session a different subset of Non-functional Requirements Templates (NoRTs) is used to support elicitation (see Section II for more information). An initial a catalog of NoRTs was proposed by the authors in 2008 [18]. Since then the method has been used in real projects and the catalog has been maintained and improved. The full version of the catalog of NoRTs was used in our exploratory case study on efficiency (cost) and effectiveness of the elicitation method (for design refer to

978-1-4799-6327-0/14/$31.00 © 2014 IEEE

Section III). Stability and containment measures were used to evaluate effectiveness of elicitation. In Section VI we summarize our findings and show the direction of future work.

II. SENoR — Structured Elicitation of Non-functional Requirements

SENoR is a method for eliciting NFRs that consists of 3 steps: Preparation, Workshop, and Follow-up. Workshop, the cornerstone of the method, is executed as a series of short brainstorming sessions. Each session is dedicated to one ISO25010's quality subcharacteristic [17], and is supported by a set of Non-functional Requirements Templates. The structured and time-boxed design of Workshop supports dealing with the heterogeneous nature of NFRs. While elements of brainstorming, and the use of both textual and visual supporting materials stimulate the creativity of participants and drive the flow of thinking in the right direction. Since the reuse of specific knowledge is proved to be effective [19], SENoR provides a package of reusable input products, e.g., documents templates, of NFRs – NoRTs, to be further improved based on the experience. The design of SENoR conforms with the guidelines for requirements workshops presented by E. Gottesdiener [20].

Roles and Responsibilities.

SENoR defines the following roles:

Moderator — a person who leads over the whole process, takes care about the proper pace, conformance with agenda, enforces the rules, so the objectives are met; is responsible also for completing administrative and logistical tasks.

Presenter — a person who at the beginning of Workshop introduces the participants into the idea of the project, presents its goal, business drivers, the core functionality.

Recorder — a person who records the work of participants as the elicited NFRs, and action items. Frequently, the role is played by a person who is responsible for requirements in the project (e.g., analyst, product owner) so they can easily extract the requirements from casual talk and monitor the discussion.

Experts — people who have content expertise for defining requirements, they actively take part in brainstorming about the NFRs. They are end-users, subject matter experts, investors, suppliers, future maintainers so they can discuss the ideas, express their expectations, and share domain knowledge.

Phases.

Preparation is the initial step of SENoR and aims at planning and preparing Workshop. Moderator ensures that funding, resources and facilities are available. If needed, appropriate management support must be confirmed. Then, Moderator answers the questions: who? when? where?, so identifies participants, assigns roles, schedules and arranges Workshop. Next, Moderator adjusts the agenda of the elicitation phase of Workshop . All sub-characteristics can be discussed during one Workshop, or they can be split into two sets regarding the interest of: (1) user, investor, (2) supplier.

The goal of the next step, i.e., *Workshop*, is to elicit non-functional requirements executing the consecutive agenda items presented in Figure 1. At first there is an introduction to the rules of Workshop, and a short reminder of the project business case – the overview presentation. Then, participants discuss each ISO25010's subcharacterisitc in succession in a three-step approach. (1) Definition - a quick clarification what a certain subcharacteritic is about, then (2) Individual Work - participants have some time to think over given subcharacterisitc answering internally the question: "What non-functional requirements that can be classified in this category do I expect from the system?", and at the end (3) Discussion - a short brainstorming session, participants propose and discuss NFRs. The second step is supported by NoRTs. Each participant receives a set of NoRTs for each subcharacteristic. Workshop ends with voting. It is not only to make an informed decision about the priorities of each NFR but also some doubts, and inconsistencies can be clarified. Moreover, Voting serves as a presentation of the results of Workshop, which is to grant Participants for their work during Workshop. The total time of Workshop should not exceed 1.5 hours, which is to be controlled by Moderator.

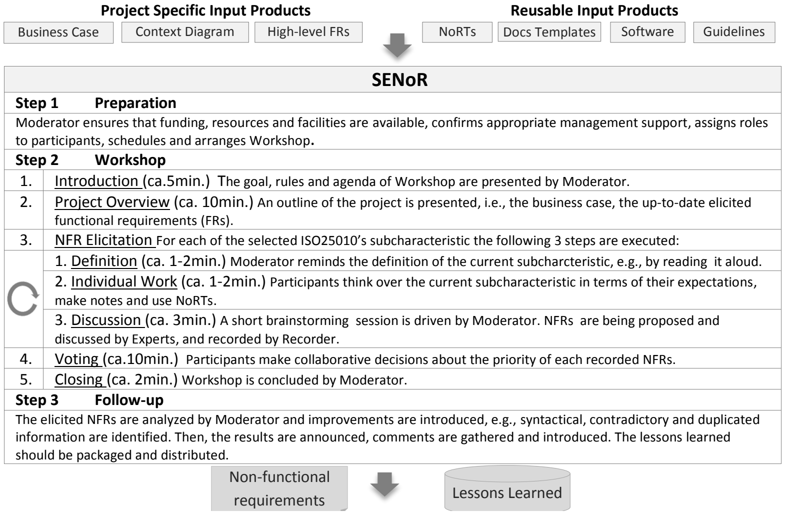
The final step of SENoR is called Follow-up. Then, Recorder first analyzes the elicited non-functional requirements for contradictory, requiring trade-offs and duplicated information, and, if necessary, improve the construction of the sentences (syntactic analysis). Next, the results should be announced, e.g., distributed by mail to the participants, and comments should be gathered. Moderator shall distribute the notes, and action items. The lessons learned shall be packaged and acknowledged. The further steps depend on the software development process, e.g., the requirements might go to a specific modeling or verification process.

Input Items and Output Products.

The structured elicitation workshop requires the use of the following input items available in the form of document templates. *Project overview presentation* is a short presentation given at the beginning of Workshop by Presenter which should cover: description of the project business case (problem, impact, solution of the problem, description of the client), time constraints of the project, scope of the project, scope by describing general functionality. It should be created based on the business case of the project, the context diagram and already defied functionality. *An ISO25010 Subcharacteristic Presentation* is a presentation that is composed of a set of definitions of the ISO25010's subcharacteristics selected for discussion and simultaneously is used to navigate Workshop (e.g., may contain some guidelines to support Moderator). *Templates of non-functional requirements* is a list of templates of NFRs (NoRTs) grouped by ISO25010's quality subcharacteristics and some space for notes. Such list should be distributed, e.g., in print-outs, among the Workshop's participants.

NoRTs (Non-functional Requirement Templates).

SENoR promotes reuse of specific knowledge on non-functional requirements in a form of NoRTs. Generally, a template is an object that is used as a guide to the form of a piece being made [21]. In terms of NFRs that are formulated using natural language, an NFR template is a



Project Specific Input Products

Business Case | Context Diagram | High-level FRs

Reusable Input Products

NoRTs | Docs Templates | Software | Guidelines

SENoR

Step 1 Preparation

Moderator ensures that funding, resources and facilities are available, confirms appropriate management support, assigns roles to participants, schedules and arranges Workshop.

Step 2 Workshop

1. Introduction (ca.5min.) The goal, rules and agenda of Workshop are presented by Moderator.
2. Project Overview (ca. 10min.) An outline of the project is presented, i.e., the business case, the up-to-date elicited functional requirements (FRs).
3. NFR Elicitation For each of the selected ISO25010's subcharacteristic the following 3 steps are executed:
   1. Definition (ca. 1-2min.) Moderator reminds the definition of the current subchracteristic, e.g., by reading it aloud.
   2. Individual Work (ca. 1-2min.) Participants think over the current subcharacteristic in terms of their expectations, make notes and use NoRTs.
   3. Discussion (ca. 3min.) A short brainstorming session is driven by Moderator. NFRs are being proposed and discussed by Experts, and recorded by Recorder.
4. Voting (ca.10min.) Participants make collaborative decisions about the priority of each recorded NFRs.
5. Closing (ca. 2min.) Workshop is concluded by Moderator.

Step 3 Follow-up

The elicited NFRs are analyzed by Moderator and improvements are introduced, e.g., syntactical, contradictory and duplicated information are identified. Then, the results are announced, comments are gathered and introduced. The lessons learned should be packaged and distributed.

Fig. 1. SENoR – the process, the input and output products

statement that requires some completion in order to become a well-formulated NFR, we present an example in Figure 2.

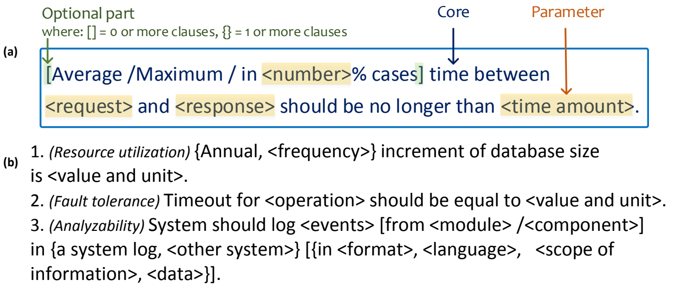
An NFR template consists of: *Core*–basis, essential part of a statement that remains unchanged; *Parameters*–a set of variables, usually the properties of a certain object, whose value must be determined to make an NFR specific and measurable; *Optional parts*–a set of phrases that can be selected. The type of optional part determines the possible number of phrases that may appear in an NFR, i.e., (1)*0..**–zero or more and (2)*1..**–one of more. NoRTs require a participant select a template and provide their expectations regarding the template's parameters and optional parts. Then, it can be augmented with some additional information to form the final NFRs, e.g., IDs, priorities. Such structure aims to provide expert knowledge and help with formulating NFRs (syntactic correctness). Each NoRT is short so it can be quickly read and a decision about its need can be made. A NoRT can be stored within a pattern library of NFRs, so then additional information can be added, e.g., relations to other NoRTs. Our work since 2008 [18] based on our experience in the projects that used NoRTs for elicitation has resulted in a set of NoRTs which we call a catalog. The catalog is divided into parts with regard to ISO25010 subcharacteristics and a set of NoRTs exists for each part.

Tool support.

Some well-known office applications can be used to support SENoR. We developed a software tool supporting Recorder with capturing the work of group during Workshop – MeetingAssistant. Most importantly it allows for recording a set of NFRs just by completing the appropriate NoRTs, supports voting, and exporting data to a document. A demo version can be downloaded from the website of the project [22].

III. Case Study

In our study we wanted to investigate and better understand SENoR (Structured Elicitation of Non-functional Requirements) and NoRTs (Non-functional Requirements Templates) in real-life settings so we decided to carry out a case study. This type of approach is useful to deeply explore a phenomenon and demonstrate its quality-in- use [23]. The goal of the case study was to characterize and understand elicitation of NFRs using SENoR and NoRTs focusing on its efficiency (cost) and effectiveness (value) from the point of view of a project stakeholder in the context of a project developing a tailor-made software application. We refined our objective into a set of the following research questions, which were to be answered through the case study analysis.



[Average /Maximum / in <number>% cases] time between <request> and <response> should be no longer than <time amount>.

1. *(Resource utilization)* {Annual, <frequency>} increment of database size is <value and unit>.
2. *(Fault tolerance)* Timeout for <operation> should be equal to <value and unit>.
3. *(Analyzability)* System should log <events> [from <module> /<component>] in {a system log, <other system>} [{in <format>, <language>, <scope of information>, <data>}].

Fig. 2. (a) an example of NoRT with its parts identified; (b) other examples of NoRTs.

RQ1 What is the effort, complexity of activities included in SENoR and what resources are used to execute them?
RQ2 What is the effectiveness of non-functional requirements elicited with SENoR and NoRTs?
RQ3 What is the participants opinion about SENoR?
RQ4 What are the factors affecting efficiency and effectiveness of SENoR?

A. Case and Subjects Selection

We decided to conduct our study in a software organization that runs several projects developing tailor-made software. The Software Department at Poznan University of Technology (PUT) is responsible for software used for various units from accounting to research and education. The Department cooperates with the Infrastructure Department responsible for the management of hardware and network infrastructure (e.g., servers), and the Department of Service Development responsible for the quality of software services provided at the University (e.g., Service Desk). Moreover, the Software Department closely works with Software Development Studio (SDS) which allows to increase its software development capacity. Each year within this cooperation ca. 5 projects are run. In each project the following people take part: employees of the Software Department (responsible for product quality), one person from the Software Engineering Laboratory (SEL)(responsible for process quality, so-called quality assurance), customer representative(s) (a University employee(s) responsible for business value, and management of project budget), representatives of end-users (employees of PUT, students or the partners of PUT who are to use the software product), a representative of supplier (an employee of PUT, who takes care of the team of developers) and the participants of the SDS: two or three Master students (who play roles of Project Manager, Analyst and Architect), and four Bachelor students (who are software developers and testers). Hence, each team is composed of 9 to 14 senior and junior technical- and process experts as in other software development organizations (e.g., industry), and the students meet with at least 3 different persons who represents the voices of customer, user, and supplier. Each project is run according to the XPrince methodology [24] and lasts for a year. There are two or three releases, each one is finished with the delivery of the software to real environment. According to XPrince, requirements are gathered after defining a business case and analyzed before architecture design, and then just before each release the requirements are being improved and made more precise during GUI workshops and planning games. Functional requirements are documented in use cases, and frequently refined to user stories. Completion of FR elicitation was a prerequisite for teams to take part in the study. More information about the environment can be found in the paper by Kopczyńska, Ochodek, Nawrocki [25]. The teams work in offices with hardware and software equipment. Moreover, each project's budget allocates salaries which are at the level of salaries that students may earn working part time in industry. The projects take ca. 1000-1200 man-hours.

The case study was carried out in 2011–2012 (1 project) and in 2012–2013 (6 projects) and required from each project to elicit non-functional requirementsusing SENoR and NoRTs at the beginning of the project, record the workshops, report on effort, and provide access to artifacts. In each team an initial version SRS underwent a review of a SEL's member. The checklist ([1]) regarding NFRs was to indicate the issues with regard to the following five aspects: Simple (is an NFR clear, unambiguous, and defied at the appropriate level of detail?), Measurable (does a cost-finite method exist that can be used to verify that an NFR is satisfied?), Achievable (is it possible that the system satisfies an NFR?), Relevant (is an NFR relevant for the particular business case?), Traceable (is there forward- and backward-traceability assured?).

B. Data Collection and Analysis Procedures

In this descriptive and explanatory case study we considered as feasible the following procedures for data collection and analysis:

- *Surveys*– short questionnaires distributed right after each Workshop asking the participants about their impression on the activities and on the quality of requirements.
- *SENoR products*–we collected the input and output products of SENoR (described in Section II).
- *Recordings*–we recorded each SENoR workshop with Polycom CX5000 which is a device allowing to create a video recording showing the participant who is speaking in a certain moment and is specially designed for video- and teleconferences.
- *Project artefacts*–we collected software requirements specifications (SRS) containing sets of non-functional requirements, and task management reports with information about the task related to requirements engineering.

The main data analyses were conducted with descriptive statistics. Video recordings were first broken into chunks so it was possible to determine the duration of each workshop's agenda item. Secondly, we used constant comparison method with a set of preformed open codes, which we intended to extend with postformed ones [26].

In order to determine effectiveness of the proposed elicitation method stability and containment measures were calculated. There can be defined 4 groups of changes in NFRs. A *minor change* is about merging two or more elicited requirements into one requirement in order to eliminate duplicates, is an syntactic improvement of an NFR (e.g., order of words), or a change caused by the change in the business context, e.g., change in law regulations. A *medium change* is about making an NFR more specific by providing more precise information, e.g., adding information about a version of a browser. By a *major change* we understand deleting, restoring, or modifying an elicited NFR. To determine the measures we counted the contribution of the elicited NFRs which are exactly the same in a given moment *t* as right after the elicitation or underwent only minor changes to the number of elicited NFRs (stability), and the number to all NFRs(contribution) (Figure 3).

$$Stability(t) = \frac{\#NFRs(t = elicition) - \#NFRs(t, medium\_change) - \#NFRs(t, major\_change)}{\#NFRs(t = elicitation)}$$

$$Containment(t) = \frac{\#NFRs(t) - \#NFRs(t, new)}{\#NFRs(t)}$$

Fig. 3. Stability and Containment.

C. Validity Procedure

We did not address *Internal validity* as is should be examined when cause-effect relations are to be found. Our case study was not designed to provide statistically valid conclusions for all software development projects or organizations.

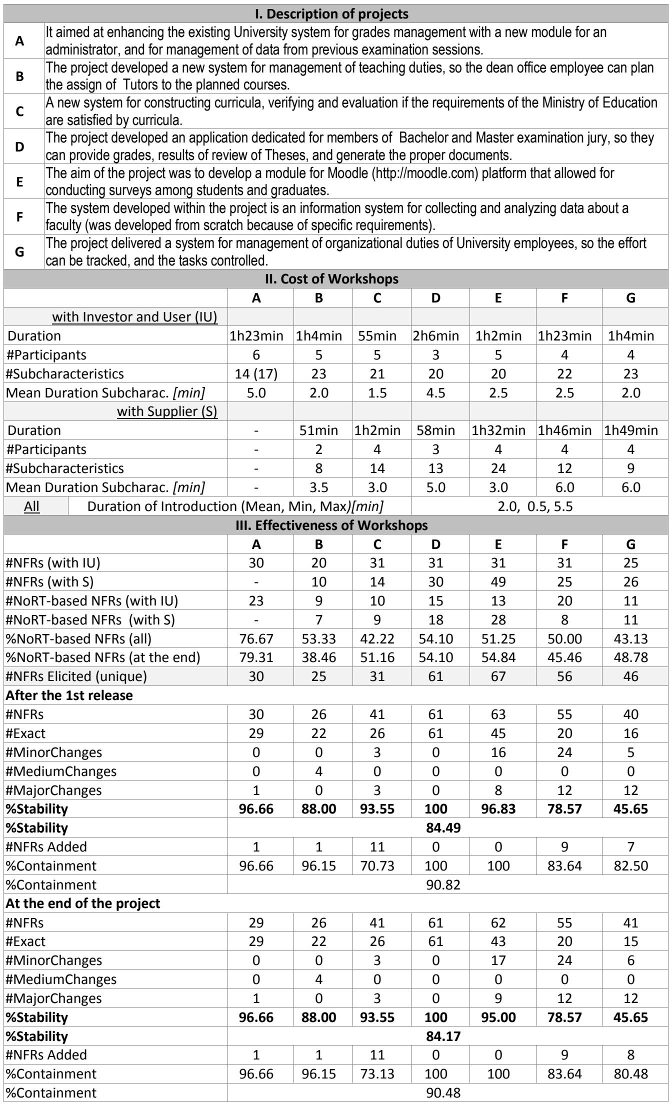
Its aim was to thoroughly describe and deeply understand the proposed method regarding the context. We tried to provide an detailed description of the context in Section III-A. There are different types of Software Studios, e.g.,[27], [28], which meet the goal of conscious application of good software engineering practices in different manner. We claim that our Software Development Studio simulates the real environment well as there are real investors, user representatives, dedicated workspaces, the teams work with other employees, and most importantly they sign contracts, may obtain salary equal to an average salary that a student may get working part-time, and participate in transition. *External validity*–environmental factors (e.g., light, temperature) or human related (e.g., a person who keeps interrupting, strongest voices which tend to win discussion) can affect the workshop. We did not try to eliminate the factors, as our aim was to evaluate the method in real life, and such factors normally appear. Moderator is then responsible for overcoming such obstacles. We applied countermeasures to detect impediments and discussed them in Section IV. The threats regarding *Construct validity* we addressed by describing elicitation process with the previously identified RE process measures and comparing to other studies characterizing elicitation methods. In order to assure *reliability* of study we linked all the data to project repositories. The duration of the agenda items of the workshops was done twice and the precision of half a minute was applied. The threat could be coding that was performed on the video recordings, not on full transcripts.

IV. Lessons Learned

In the following paragraphs we discuss the results of our case study investigating 7 projects. The team members organized 13 SENoR workshops in which 50 project stakeholders participated (see Figure 4). One team organized just one workshop in which the user, investor, and supplier representative roles were played by one person. 38 participants responded in the surveys right after the workshops (we call them survey participants further on).

[RQ1] Effort.

The study participants who were to play Moderator and Recorder roles participated in a tutorial on NFRs. The first part of the tutorial (1h30min) was on general knowledge of non-functional requirements. During the second part (1h30min) participants solved some tasks regarding NFRs and familiarize themselves with guidelines on organizing Workshops. The average total time of workshop is ca. 1h16min (min. 55min, max. 2h5min), which is the time to execute the planned agenda items. The time for Project overview, Elicitation of NFRs, and Voting is presented in Figure 5. A short brainstorming session on one subcharacterisitc took ca. 3-4 minutes. During two workshops of project B, ca. 4min and ca. 6min were spent on clarifying functional scope and the aim of the project which has not been properly presented (participants had mixed feelings about the quality of presentation, which was confirmed by the results of the survey). During one workshop of project F, one function was added while presenting the overview of project, which took ca. 6min. The presentations took on average ca. 4min (min. 1.5min, max. 7.5min), and 92.11% (35) survey participants agreed that it is worth to present business drivers. In two workshops the voting was not carried out because of time constrains of participants. For one workshop we cannot provide its duration because of a failure of the recording device.



I. Description of projects	
A	It aimed at enhancing the existing University system for grades management with a new module for an administrator, and for management of data from previous examination sessions.
B	The project developed a new system for management of teaching duties, so the dean office employee can plan the assign of Tutors to the planned courses.
C	A new system for constructing curricula, verifying and evaluation if the requirements of the Ministry of Education are satisfied by curricula.
D	The project developed an application dedicated for members of Bachelor and Master examination jury, so they can provide grades, results of review of Theses, and generate the proper documents.
E	The aim of the project was to develop a module for Moodle (http://moodle.com) platform that allowed for conducting surveys among students and graduates.
F	The system developed within the project is an information system for collecting and analyzing data about a faculty (was developed from scratch because of specific requirements).
G	The project delivered a system for management of organizational duties of University employees, so the effort can be tracked, and the tasks controlled.

II. Cost of Workshops							
	A	**B**	**C**	**D**	**E**	**F**	**G**
with Investor and User (IU)							
Duration	1h23min	1h4min	55min	2h6min	1h2min	1h23min	1h4min
#Participants	6	5	5	3	5	4	4
#Subcharacteristics	14 (17)	23	21	20	20	22	23
Mean Duration Subcharac. *[min]*	5.0	2.0	1.5	4.5	2.5	2.5	2.0
with Supplier (S)							
Duration	-	51min	1h2min	58min	1h32min	1h46min	1h49min
#Participants	-	2	4	3	4	4	4
#Subcharacteristics	-	8	14	13	24	12	9
Mean Duration Subcharac. *[min]*	-	3.5	3.0	5.0	3.0	6.0	6.0
All	Duration of Introduction (Mean, Min, Max)*[min]*			2.0, 0.5, 5.5			
III. Effectiveness of Workshops							
	A	**B**	**C**	**D**	**E**	**F**	**G**
#NFRs (with IU)	30	20	31	31	31	31	25
#NFRs (with S)	-	10	14	30	49	25	26
#NoRT-based NFRs (with IU)	23	9	10	15	13	20	11
#NoRT-based NFRs (with S)	-	7	9	18	28	8	11
%NoRT-based NFRs (all)	76.67	53.33	42.22	54.10	51.25	50.00	43.13
%NoRT-based NFRs (at the end)	79.31	38.46	51.16	54.10	54.84	45.46	48.78
#NFRs Elicited (unique)	30	25	31	61	67	56	46
After the 1st release							
#NFRs	30	26	41	61	63	55	40
#Exact	29	22	26	61	45	20	16
#MinorChanges	0	0	3	0	16	24	5
#MediumChanges	0	4	0	0	0	0	0
#MajorChanges	1	0	3	0	8	12	12
%Stability	**96.66**	**88.00**	**93.55**	**100**	**96.83**	**78.57**	**45.65**
%Stability	**84.49**						
#NFRs Added	1	1	11	0	0	9	7
%Containment	96.66	96.15	70.73	100	100	83.64	82.50
%Containment	90.82						
At the end of the project							
#NFRs	29	26	41	61	62	55	41
#Exact	29	22	26	61	43	20	15
#MinorChanges	0	0	3	0	17	24	6
#MediumChanges	0	4	0	0	0	0	0
#MajorChanges	1	0	3	0	9	12	12
%Stability	**96.66**	**88.00**	**93.55**	**100**	**95.00**	**78.57**	**45.65**
%Stability	**84.17**						
#NFRs Added	1	1	11	0	0	9	8
%Containment	96.66	96.15	73.13	100	100	83.64	80.48
%Containment	90.48						

Fig. 4. Description of Workshops.

[RQ1] Resouces.

During 11 workshops there were two laptops used: one for video recording and documenting the group work with MeetingAssistant, and the other for showing a presentation. 2 workshops used just an audio recorder and took notes on print-outs with the input products. During three workshops there were technical problems with laptops, which prolonged the workshops by up to 4min. The presentations were prepared with Microsoft Power Point or Open Office. Participants were provided printed-out supporting materials: the set of quality

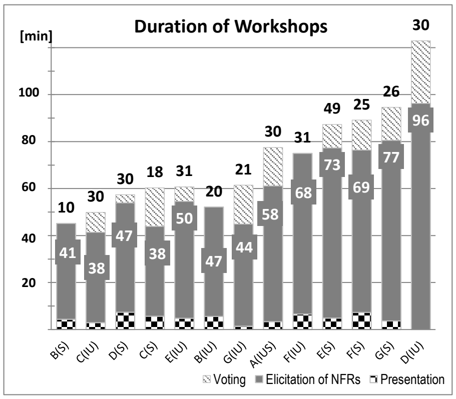


Fig. 5. Duration Workshops. The white font is used for the duration of the Elicitation of NFRs , and the black font for the number of elicited NFRs.

subcharacteristics each with a description and a set of NoRTs for each one.

Conclusions: The proposed method required to schedule up to two 1.5h workshops. During one workshop ca. 20 quality subcharacterisitcs can be discussed. The cost of using the method should include an initial investment of ca. 3h for training. The requirements to execute SENoR and NoRTs regarding resources do not differ from those of a typical business meeting or workshop.

Lessons Learned (LL): It is valuable to provide an overview of a project at the beginning of Workshop and invest 4-10min in it. If there are some doubts regarding functionality or project vision, they will probably appear during elicitation of NFRs and negatively influence the efficiency. Discussion about NFRs may lead to new functional requirements.

[RQ1] Stability, Containment and Use of NoRTs.

The results presented in Figure 4 Part III show that the stability of all elicited NFRs is 84.49% after the 1st release and 84.17% at the end of projects. Project D scored stability of 100%, but the thorough analysis of project artifacts suggests that the requirements were probably not sufficiently analyzed by team members. Then, for Project G we obtained low stability – 45.63%, because of lack of knowledge of the supplier representative, and the missing NFRs were added after consultations with other supplier representatives. The stability after the first release (84.49%) contains also the changes that were made after the review of SRS with regard to quality of NFRs. Contaminant of elicited requirements was from 100% up to 70.73%, which means no many new NFRs were added. The effectiveness can also be characterized by the percentage of non-functional requirements that satisfy NoRTs. A non-functional requirement A satisfies a NoRT if after specifying parameters and selecting optional parts of this NoRT we obtain the NFRs A (semantically the same, but there could be some syntactic differences with respect to word order). Figure 4 Part III presents the usage of NoRTs in the investigated projects. 51.56% of all elicited NFRs satisfy NoRT. Moreover, 84.21%(32) (Figure 6) of survey participants regard NoRTs as useful. These two results suggest this type of supporting material is valuable. However, this result shows also that about 48% of requirements were creatively defined, so structured workshops did not stifle creativity of participants. This result is influenced by the completeness of the set of NoRTs provided to meeting participants. But, the more complete the set would become, the higher the cost (effort to retrieve the needed NoRTs) may become.

Conclusions: Over 80% of the elicited NFRs from the investigated projects remained stable (exactly or underwent minor changed), and they constituted ca. 90% of all NFRs. Moreover, ca. 52% of all elicited NFRs satisfy NoRTs.

[RQ3, RQ4] Quaility in use.

The results presented in Figure 6 show that 89.47% (34) survey participants have positive impression (answered ”Strongly yes” or ”Rather yes”) about the workshops and would like to conduct such workshops in their future projects. 92.11% (35) claim that the workshop's goal was achieved, and the quality of the output – NFRs is good-enough (they are sufficiently correct and complete) to start architecture stage. The usefulness of the elements that supported the workshops were rated high, i.e., descriptions of subcharacterisitcs (86.84%, 33 participants), templates of NFRs(84.21%, 32). Furthermore a project overview presentation was claimed as valuable by 78.95% 30) survey participants, and 92.11% regarded it as the element worth to include in such meetings, and the quality was rated positively by 65.79% (25) survey participants. Such situation may indicate that the template of business presentation might require improvement, or further investigation is required. Short surveys were one of the elements that have directed development of SENoR since its first use in 2008 [18]. In the projects that we present in the paper we introduced a new element to the method, namely we asked participants to perform a pre-task. The goal was to investigate if efficiency of SENoR can be improved, as the planned participation is reported to promote productivity [20]. A pre-survey asking about their priorities of the quality subcharacteristics was supposed to be sent to the participants together with the invitations. The goal was to (1) let the participants familiarize with the jargoon of the ISO25010 (especially that in 6 projects people with no experience in software projects were representing end-users), and (2) provide some guidance to the moderator which subcharacteristics should be more thoroughly discussed. In 9 (out of 13) workshops a pre-survey was sent, and reached 27 survey participants. Seven people did not respond, 11 claim that it is hard to say if the survey prepared them to the workshop, 7 participants had a positive opinion about the pre-task. Moreover, we have not found any relation between the stability of NFRs and the fact the participants filled-in the survey.

Conclusions: Participants of the workshops regard the current design of SENoR as well-suited for their context.

Lessons Learned: Using a survey right after a workshop may be a quick and good indicator of improvement areas. A pre-task before attending SENoR in a form of a short survey asking about priority of subcharacteristics to-be-discussed might not be completed.

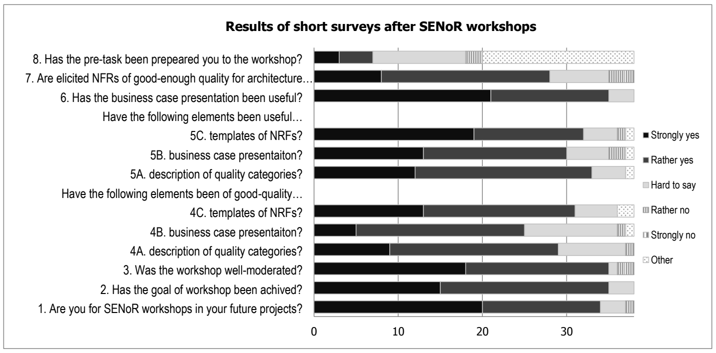


Fig. 6. The results of the survey carried out at the end of workshops.

V. Related Work

As the first in the series of the RE processes (elicitation, modeling, negotiation, specification, and validation [29]), elicitation influences the performance and quality of further steps. It is regarded as one of the most critical activities in software development, but also as very difficult. Hence probably it frequently happens in ad hoc manner [30]. However, there is some work on more systematic approaches. From those which recommend some general best practices for requirements elicitation, e.g., [31], through those which discuss in detail some general techniques that could be used also for RE like Joint Application Development workshops [32], to publications dedicated for RE, e.g., EasyWinWin [33]. Some researchers reported their experience on eliciting requirements with specific techniques or on combing different techniques [34]. Much has especially been done in the area of supporting functional requirements elicitation, e.g., Gottensdiener provides a comprehensive set of valuable guidelines for functional requirements workshops [20]. On these guidelines we based our method (Section II). Other studies investigated certain aspects of requirements elicitation, and report for example that it is crucial to involve the customer and the user into the RE process [35].

With regard to the effectiveness and efficiency of the elicitation techniques, little still is known. Especially about the experience in the real-life projects, which has recently been raised as a vital issue [9]. With regard to non-functional requirements, they are either elicited together with functional requirements or dealt with separately with the use of the existing methods, e.g., MOQARE [36], NFR Framework [10], KAOS [11]. However, existing methods are reported to be costly in terms of effort and complexity [37], and regarded as too heavy-weight especially for the agile context [12]. In consequence, new light-weight methods are proposed, e.g., ASP-Lite [12]. A few researchers attempted to characterize the existing methods for elicitation of NFRs [38].

The findings published in 2001 that successful project teams allocate ca. 28% of the project effort to RE, and ca. 11% is spent on elicitation [35] driven our work on SENoR and NoRTs. Such amount of project effort is significant and so optimization in this matter shall be sought. Since the reuse of project artifacts and knowledge [19] is reported to successfully decrease the project effort we applied reuse of templates of documents as, e.g., for specifications ([39], [1]). Reuse of knowledge is also realized e.g., by applying lessons learned, coping similar requirements from other projects, or using patterns.

The latter approach has won recognition in the field of NFRs, many different patterns were proposed, e.g., for security requirements [40]. In order to document patterns various pattern templates are used, so different attributes may describe it, e.g., ID, name, problem, solution, relationships, examples. Sometimes, as, e.g., [15], [14], to demonstrate the way in which a certain pattern should be used a template of NFR is used. Since the choice a pattern template has great impact on the effectiveness and efficiency of the pattern management, i.e., creation, maintenance, and also usage [41], we decided that we will create a catalog of templates of NFRs. Although then, such information as relationships between other NFRs is missing (as it is not in [42]), the catalog contains a set of short (hence quick) and easy to scan templates by the user and the customer during an elicitation meeting. The aim of creating short statements containing expert knowledge and expressed in natural language was also to reduce the need of learning a new notation for stakeholders, e.g., clients, users, as might be in, e.g., NFR Framework [10]. Although it is known that a pattern library (or a catalog) should exhibit high retrievability and relevance, there is scarcity of information about the evaluations in real projects looking at reuse from project perspective, including, e.g., effort, cost, influence on other practices. The criteria shall be considered when deciding about usage of requirements reuse technique [43].

VI. Conclusions

In the paper we present a method for eliciting non-functional requirements called SENoR which is composed of a series of short brainstorming sessions driven by the ISO25010's quality subcharacteristics. Elicitation is supported by templates of input products, and by Non-functional Requirements Templates (NoRTs). The main goal of our paper is to report on our exploratory case study on 7 software development projects which used the presented method. We aim to extend the up-to-date knowledge of elicitation methods used in real projects which is reported as scarce [9]. Thus making decision on assigning time and other resources to elicitation becomes informed.

The cost of using the method was to execute two ca. 1.5-hour elicitation workshops. The teams did it just after general functional requirements definition, but before architecture design and coding. About 90% of participants were satisfied with the quality of the elicited NFRs, and would like to use this approach in their future projects. The effectiveness of the method was evaluated with the stability and containment measures. The results show that ca. 80% of elicited NFRs remained the same or almost the same till the end of the projects, and they constituted ca. 90% of all NFRs.

The findings help draw future research directions. We continue our study in other projects and organizations, and work to improve its cost and effectiveness. We would like also to carry out case studies in real projects on other elicitation methods.

Acknowledgment

We thank the team members of the software development projects who participated in our study and Konrad Siek for a review of the paper.

References

[1] IEEE, "IEEE Recommended Practice for Software Requirements Specifications," *IEEE Std 830-1998*, pp. 1–40, 1998.

[2] K. Beck and C. Andres, *Extreme Programming Explained: Embrace Change (2nd Edition)*. Addison-Wesley Professional, 2004.

[3] K. Schwaber, "Scrum development process," in *Proc. of the 10th Annual ACM Conf. OOPSLA*, 1995, pp. 117–134.

[4] L. Cao and B. Ramesh, "Agile requirements engineering practices: An empirical study," *Software, IEEE*, vol. 25, no. 1, pp. 60–67, 2008.

[5] N. G. Leveson and C. S. Turner, "An investigation of the therac-25 accidents," *Computer*, vol. 26, no. 7, pp. 18–41, 1993.

[6] B. Nuseibeh, "Ariane 5: Who dunnit?" *Software, IEEE*, vol. 14, no. 3, pp. 15–16, 1997.

[7] B. Boehm and R. Turner, *Balancing Agility and Discipline: A Guide for the Perplexed*. Addison-Wesley, 2004.

[8] J. Cleland-Huang, "Quality Requirements and their Role in Successful Products," in *IEEE Intl RE Conf.* IEEE, Oct. 2007, pp. 361–361.

[9] T. Dyba and D. S. Cruzes, "Process research in requirements elicitation," in *Empirical Requirements Engineering (EmpiRE), 2013 IEEE Third International Workshop on*, 2013, pp. 36–39.

[10] J. Mylopoulos, L. Chung, and B. Nixon, "Representing and using nonfunctional requirements: a process-oriented approach," *IEEE Trans. on Software Engineering*, vol. 18, no. 6, pp. 483–497, 1992.

[11] A. Van Lamsweerde, "Goal-oriented requirements engineering: A guided tour," in *Proc. of 5th IEEE Intl Symp. on RE*. IEEE, 2001, pp. 249–262.

[12] J. Cleland-Huang, A. Czauderna, and E. Keenan, "A persona-based approach for exploring architecturally significant requirements in agile projects," in *REFSQ*. Springer, 2013, pp. 18–33.

[13] J. Bhatia, R. Sharma, K. Biswas, and S. Ghaisas, "Using grammatical knowledge patterns for structuring requirements specifications," in *IEEE RePa 2013*, pp. 31–34.

[14] S. Withall, *Software Requirement Patterns (Developer Best Practices)*. Microsoft Press, 2007.

[15] S. Renault, Ó. Méndez Bonilla, J. Franch Gutiérrez, M. C. Quer Bosor *et al.*, "A pattern-based method for building requirements documents in call-for-tender processes," *IJCSA*, vol. 6, no. 5, 2009.

[16] L. Zong-yong, W. Zhi-xue, Y. Ying-ying, W. Yue, and L. Ying, "Towards a Multiple Ontology Framework for Requirements Elicitation and Reuse," in *31st COMPSAC 2007*, vol. 1. IEEE, Jul. 2007, pp. 189–195.

[17] ISO, *ISO/IEC 25010:2011 - Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models*. Geneva, Switzerland: International Organization for Standardization, 2011.

[18] S. Kopczyska, M. Maćkowiak, and J. Nawrocki, "Structured meetings for non-functional requirements elicitation," *Foundat. of Computing and Decision Sciences*, vol. 36, no. 1, pp. 35–56, 2011.

[19] K. Pohl, "The three dimensions of requirements engineering," in *Advanced Information Systems Engineering*, vol. 6353. Springer, 1993, pp. 275–292.

[20] E. Gottesdiener, *Requirements by Collaboration: Workshops for Defining Needs*. Addison-Wesley Professional, 2002.

[21] Template – a definition, "from Merriam Webster, http://www.merriam-webster.com/dictionary/template, last access 30/07/2013." [Online]. Available: http://www.merriam-webster.com/dictionary/template

[22] Website of SENoR. last access 08/06/2014. [Online]. Available: http://senor.cs.put.poznan.pl

[23] P. Runeson, M. Host, A. Rainer, and B. Regnell, *Case Study Research in Software Engineering: Guidelines and Examples*. Wiley, 2012.

[24] J. Nawrocki, L. Olek, M. Jasinski, B. Paliświat, B. Walter, B. Pietrzak, and P. Godek, "Balancing agility and discipline with XPrince," in *Proc. of RISE 2005 Conf., vol. 3943 of LNCS*. Springer, 2006, pp. 266–277.

[25] S. Kopczynska, J. Nawrocki, and M. Ochodek, "Software development studio - Bringing industrial environment to a classroom," in *Porc. of 1st EduRex*. IEEE, 2012, pp. 13–16.

[26] C. Seaman, "Qualitative methods," in *Advanced Empirical Software Engineering*, F. e. a. Shull, Ed. Springer, 2008, pp. 35–62.

[27] C. Wang, "Software development studio," The Univ. of Texas at Austin, Tech. Rep., 2004. [Online]. Available: http://www.cs.utexas.edu/ wcook/projects/misc/WantStudio1.pdf

[28] D. Root, M. Rosso-Llopart, and G. Taran, "Proposal based studio projects: How to avoid producing "cookie cutter" software engineers," in *Proc. of the 21st CSEET*. IEEE, 2008, pp. 145–151.

[29] A. M. Hickey and A. M. Davis, "Requirements Elicitation and Elicitation Technique Selection : A Model for Two Knowledge-Intensive Software Development Processes Unsolved Problem Software Development Software Solutions," in *Proceedings of HICSS'03*. IEEE, 2002.

[30] G. J. Browne and V. Ramesh, "Improving information requirements determination: a cognitive perspective," *Information & Management*, vol. 39, no. 8, pp. 625–645, Sep. 2002.

[31] I. Sommerville and P. Sawyer, *Requirements engineering: a good practice guide*. John Wiley & Sons, 1997.

[32] J. Wood and D. Silver, *Joint application development*. Wiley, 1995.

[33] P. Gruenbacher, "Collaborative requirements negotiation with easywinwin," in *Database and Expert Systems Applications, 2000. Proceedings. 11th International Workshop on*, 2000, pp. 954–958.

[34] A. Sutcliffe, "A technique combination approach to requirements engineering," in *RE 1997., Proceed. of*, 1997.

[35] H. Hofmann and F. Lehner, "Requirements engineering as a success factor in software projects," *IEEE Software*, vol. 18, no. 4, pp. 58–66, 2001.

[36] A. Herrmann and B. Paech, "Moqare: misuse-oriented quality requirements engineering," *Requirements Engineering*, vol. 13, no. 1, pp. 73–86, 2008.

[37] M. Strohmaier, J. Horkoff, and E. Yu, "Can Patterns improve i* Modeling? Two Exploratory Studies," in *REFSQ*. Springer, 2008, pp. 153–167.

[38] A. Herrmann, D. Kerkow, and J. Doerr, "Exploring the characteristics of nfr methods–a dialogue about two approaches," in *REFSQ*. Springer, 2007, pp. 320–334.

[39] S. Robertson and J. Robertson, *Mastering the Requirements Process: Getting Requirements Right (3rd Edition)*. Addison-Wesley, 2012.

[40] M. Riaz and L. Williams, "Security requirements patterns: understanding the science behind the art of pattern writing," in *RePa 2012*. IEEE, pp. 29–34.

[41] J. Naish and L. Zhao, "Towards a generalised framework for classifying and retrieving requirements patterns." in *RePa*, 2011, pp. 42–51.

[42] S. Supakkul, T. Hill, L. Chung, T. T. Tun, and J. Sampaio do Prado Leite, "An NFR Pattern Approach to Dealing with NFRs," in *RE 2010*, pp. 179–188.

[43] W. Lam, J. McDermid, and A. Vickers, "Ten steps towards systematic requirements reuse," in *RE 1997*, pp. 6–15.