An empirical study on the use of context annotations and flow expressions to specify the behavior of context-sensitive systems

Jéssyka Vilela[1], Jaelson Castro[1], João Pimentel[12], Enyo Gonçalves[13]

[1]Universidade Federal de Pernambuco (UFPE)
[2] Universidade de Pernambuco (UPE)
[3] Universidade Federal do Ceará (UFC)

{jffv, jbc, jhcp, ejtg }@cin.ufpe.br

Abstract. Context-Sensitive Systems (CSS) must detect variations in their operating context and adapt their behavior in response to such variations. Accordingly, the specification of the required context and the behavior of the tasks of such systems is a complex and labor-intensive task, which calls for systematic methods. In previous work, we have presented the GO2S (GOals to Statecharts) process to derive the behavior of context-sensitive systems from requirements models. In this paper, we report the findings of an empirical study conducted with twenty-two professionals. We examine how they perceive the complexity of the notations for context and for behavior specification, as well as the complexity of the GO2S process. The study findings showed that approximately 81% of the subjects specified the behavior correctly, whereas 97% identified the contextual variation points correctly. Moreover, according to the subjects' feedback, the notations for behavior and context specification and the GO2S process have a moderate complexity. Finally, we conclude this work by showing promising research opportunities underexplored in current research.

Keywords: Context-sensitive systems, Context Annotations, Flow expressions, GO2S Process, Behavior, Empirical study.

1 Introduction

Context-Sensitive Systems (CSS) monitor the environment and the level of satisfaction of its goals. They detect changes and react to context to provide services and relevant information to their users [1]. Accordingly, in order to satisfy its goals, the system should adapt its behavior, supporting different variants that enable itself to operate in varying contexts.

Many researchers have recognized the importance of context modeling for an effective adaptation such as [1][2][3]. However, the specification of what needs to be monitored as well as the adaptation mechanism in such applications is a challenging task due to their dependence on varying contextual elements, which need to be made explicit.

In order to design the behavior of context-sensitive systems, several steps must be taken. In previous work [4][5] we have presented the GO2S process, a systematic approach for deriving the behavior (expressed in statecharts) of context-sensitive systems, from requirements models (described as goal models). A complete description of GO2S can be found in our website[1].

The GO2S is an iterative process centered on the incremental refinement of a goal model, which is used to obtain different views of CSS, namely: design, contextual, and behavioral views. The GO2S process has a step to perform the context specification through context annotations in a goal model and refinements as well as another step to specify the system's behavior through flow expressions.

This paper contributes with an empirical study, which main purpose is to get initial feedback from industry professionals (subjects). We conducted it in order to understand the complexity of context annotations and flow expressions used to specify the behavior of context-sensitive systems. Moreover, we evaluated the complexity of GO2S process.

We consider the study results as particularly useful for the following audience: software engineers that develop CSS and want to improve their development method, since it reveals first insights on the benefits and limitations of the GO2S process; and researchers in requirements engineering and CSS who want to define research questions and set up hypotheses for future independent empirical investigations.

The remainder of this paper is structured as follows. The requirements model of the system analyzed in this study is presented in Section 2. In Section 3, we discuss related work. Section 4 gives an overview of the research method used. Section 5 presents the findings. Threats to validity are addressed in Section 6. We discuss the study results in Section 7. Last but not least, we draw conclusions and present future work in Section 8.

2 Background

The GO2S process uses context annotations in a contextual goal model [3] to perform context specification. The behavior specification is defined through flow expressions [7]. Flow expressions area particular flavor of regular expressions and can be used to describe the system behavior through its goals and tasks in a goal model.

In this study, we used a small version of the Smart Home system [6]. This system has been designed to make life easier for people with dementia problems and provide continuous care to them, ensuring their safety and comfort. In order to achieve that, the smart home has to act in response to the context.

The contextual goal model of the Smart Home system annotated with flow expressions is presented in Figure 1. The main goal of the system is to control the home for the patient: people with dementia suffer from serious problems with memory. As a result of these problems, a patient may forget to maintain a healthy family

[1] http://www.cin.ufpe.br/~ler/supplement/jffv/wer2016/

environment. The features of the system are the control of gas leaking, management of temperature, protection against theft and entertaining the patient.

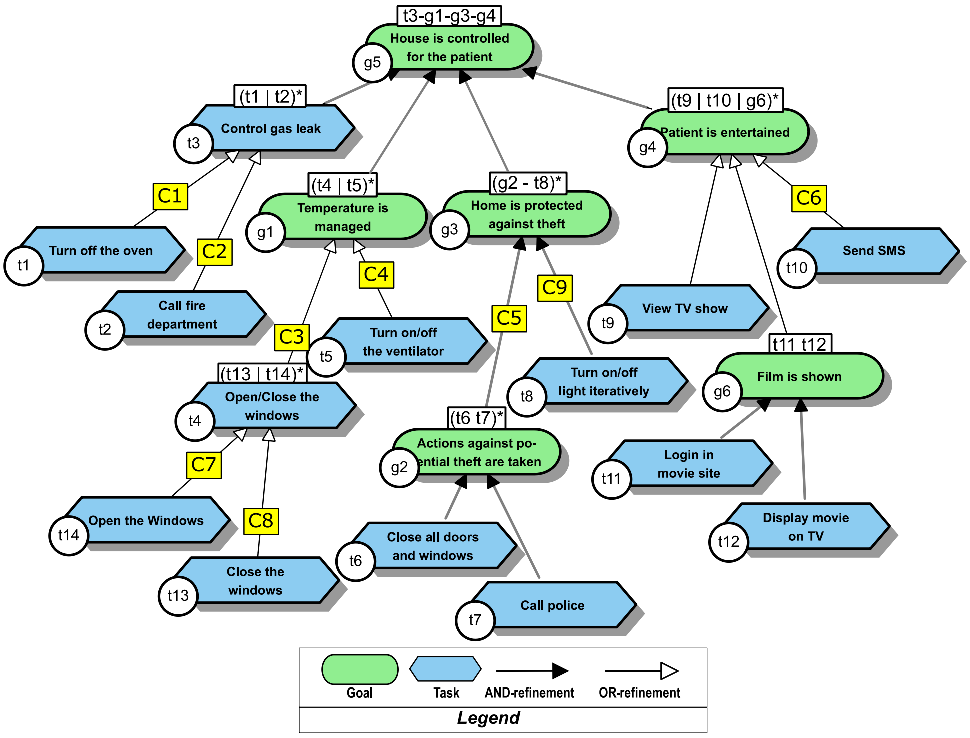

Figure 1. The contextual goal model of the Smart Home system annotated with flow expressions.

3 Related Work

The contextual goals models were analyzed in two systems (smart home and museum-guide) in the work of [6]. The study subjects were analysts that already were familiar with emerging computing paradigms scenarios and had a good expertise in goal modeling. The results of this work showed that the contextual goal model was easy to understand and the framework provided a useful systematic way to analyze contexts. Moreover, the relation between goals and context was strong in certain systems; the context specification could be a subject to viewpoints (in some other cases, it was debatable if a certain context was a statement or a fact) [6].

The use of flow expressions to specify the behavior of systems has been accepted and successfully used in many frameworks [7][8][9]. As evidence of the notation usefulness, [8] exhibited the application of flow expressions in many areas including the modeling of concurrent programs, the description of operating system architectures, the specification of synchronization problems and solutions, the flow and description of command languages, and systems analysis. We relied on this notation in the GO2S process to specify the behavior of context-sensitive systems and to derive contextualized statecharts.

The work of [7] explores the flow expressions in their multi-dimensional approach that exploits inherent variability of the design space. In this work, alternative refinements are considered for the same intermediate problem, resulting in multiple solutions (statecharts) from a single initial problem (requirements).

The flow expressions were also adopted in the work of [9] to propose a conceptual distinction between Design-time Goal Models (DGMs) - used to design a system - and Runtime Goal Models (RGMs) – deployed in the analysis of the system's runtime behavior with respect to its requirements. In their work, RGMs extend DGMs, In essence, they define some additional states and flow expressions as well as historical information about the fulfilment of goals.

4 RESEARCH METHOD

Our study has an investigative focus; hence, the purpose of this study was not to test hypotheses, but to explore what subjects' opinions and attitudes are with regard to a carefully selected set of research questions. We, therefore, designed our study to be explorative in nature, choosing a qualitative approach. As an exploratory study, it can help the identification of further research questions, as well as the definition of hypotheses that can be tested by means of analytical studies [17][10]. In this section, we discuss our research questions, the study design, its subjects, the data gathering methods and the data analysis approach.

4.1 RESEARCH QUESTIONS

This study strives to answer the following research questions:

RQ1: How do the subjects perceive the complexity of the contextual goal model and flow expressions?

RQ2: How do the subjects perceive the complexity of the GO2S process steps?

RQ3: What are the benefits stated by the subjects of the contextual goal model and flow expressions as well as the steps of the GO2S process?

RQ4: What challenges are related to the use of the GO2S process?

4.2 STUDY DESIGN

This study was conducted in April of 2015 and it consisted of four phases: profiling, training, study and post-study (see Figure 2). In the first phase, we identified the subjects' profile, current job and previous experiences through an online questionnaire. The findings from this phase are presented in Section 4.3.

The second phase consisted on training. In this phase, we provided a 3 hours training to the subjects about the GO2S process and its modeling notation.

For the study itself (third phase), we prepared a set of resources: a goal model of a simplified version of the Smart Home System, without context annotations nor flow expressions (adapted from [6]); a textual description of the system behavior; a reference guide with a summary of the activities and the notations used by the GO2S process; and the slides used for their training. All the resources were provided in Portuguese and are available at our website[2].

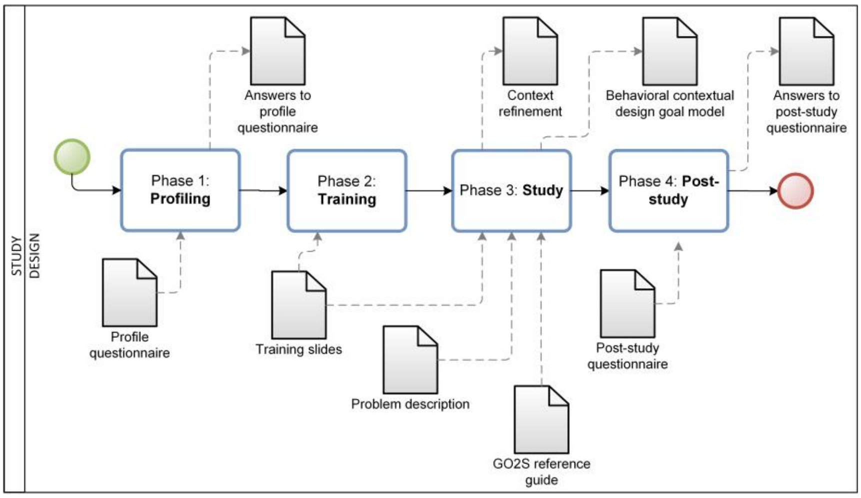

Figure 2. Study Design.

After the training, in the study itself (Phase 3) we asked the subjects to work in a project of a Smart Home system. It consisted in applying the GO2S process to perform the context and flow expressions specification. The subjects had to identify and refine the nine context annotations and eight flow expressions presented in Figure 1.

The last phase of our study consisted in applying a structured and self-administrated questionnaire with the twenty-two subjects in order to obtain their feedback about the GO2S process and its notation. After the study, we collected all the artifacts and answers from the subjects. All questionnaires were reviewed and analyzed – none was rejected due to incompleteness or errors.

4.3 STUDY SUBJECTS

The subjects were twenty-two professionals, i.e. they were working in the industry at the time when we conducted the experiment, with experience ranging from 1 to 15 years. The subjects, students of a professional master course, were invited by email. Their profile, current job and previous experiences were obtained with a pre-study questionnaire. The subjects have different background, i.e. they attended different undergraduate courses. The most studied course was *Information Systems*, followed by *Computer Science* and *Systems Development and Analysis*.

[2] http://www.cin.ufpe.br/~ler/supplement/jffv/wer2016/

The professional experience time of the subjects is presented in Figure 3. They have different degrees of experience. Notice that 50% of the subjects had more than one year of experience when the experiment was conducted.

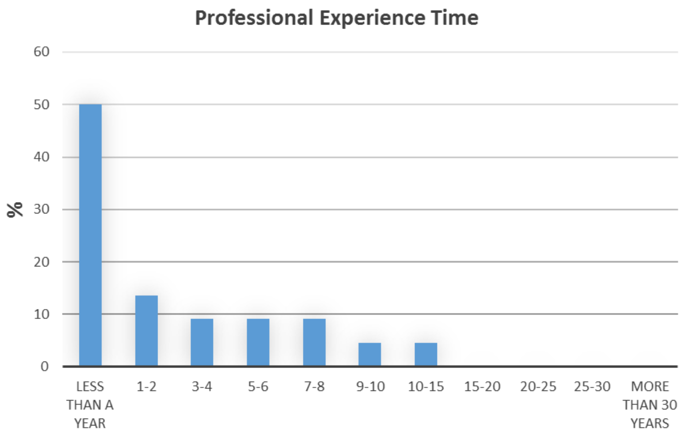

Figure 3. Professional Experience Time of the subjects.

We also wanted to understand the professional experience in software engineering of the subjects. Therefore, we asked them about their job history. This was a multiple selection question and the collected data are presented in Figure 4. Most subjects were Developer and System Analyst.

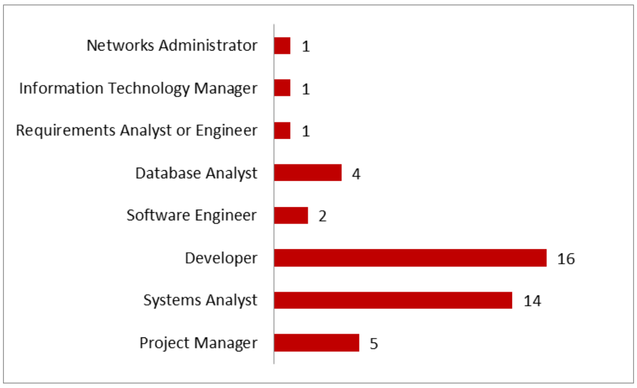

Figure 4. Subjects' previous jobs.

4.4 DATA GATHERING METHODS

In our study, we used a structured and self-administered questionnaire [9] for collecting the data from the subjects. Hence, all subjects were exposed to the same questions. The idea of using structured questionnaire was to control the input that triggered the subjects' answers so that their outputs can be reliably compared [9]. We emailed to the subjects the invitations for answering the pre-study questionnaire applied in order to obtain the subjects' profile and professional experience.

The subjects' answers about the complexity of the notations and the GO2S process were collected in a self-administrated questionnaire, which included a mix of close-ended and open-ended questions. This questionnaire was available online. The open-ended questions gave the subjects the possibility to provide a rationale for their decisions. The close-ended questions followed two well-known and widely used Likert scales [9]. The first one was Very Easy, Easy, Reasonable, Difficult and Very Difficult. The second one was Totally Agree, Agree, Indifferent, Disagree and Totally Disagree. The original questionnaires can also be found in our website.

4.5 DATA ANALYSIS

The analysis of data can be performed either quantitatively or qualitatively. According to [11], each one of these approaches can be classified in four dimensions: the qualitative analysis of qualitative data, the quantitative analysis of qualitative data, the qualitative analysis of quantitative data and the quantitative analysis of quantitative data.

For the analysis of our results, we relied on the quantitative analysis of qualitative data. The analysis of the subjects' profile included the construction of the profile matrix, or simply, the data matrix, which is a table of cases and their associated values [9].

Our analysis of the close-ended questions was performed through a quantitative analysis of quantitative data. This analysis was performed through mathematical analysis of the numeric data. The open-ended questions were analyzed by a quantitative analysis of the qualitative data. The subjects' statements were structured and displayed in an Microsoft Excel spreadsheet based on the selected key aspects using the coding technique. The traceability between the statements and the raw data were also established.

5 FINDINGS

In this section, we analyze the correctness of the models generated by the subjects as well as our research questions.

5.1 Correctness of the models

In relation to the correctness of context annotations specified in the contextual goal model, we observed that the number of context annotations specified correctly, i.e. annotations specified as described in the requirements document, was 97.53% with a standard deviation of 22.32%. We observed that some subjects specified more contexts than described in the specification. The average amount of contexts specified in excess by the subjects had a mean of 1.64% with a standard deviation of 3%. Moreover, the mean of flow expressions specified correctly was 80.95% with a standard deviation of 18.09%.

Figure 5 presents the boxplot of these two variables investigated: (a) context representation and (b) flow expressions. After analyzing this figure, we found 1 potentially outlying observation at the context representation since that subject did not specify correctly any context (bottom-center of Figure 5.a).

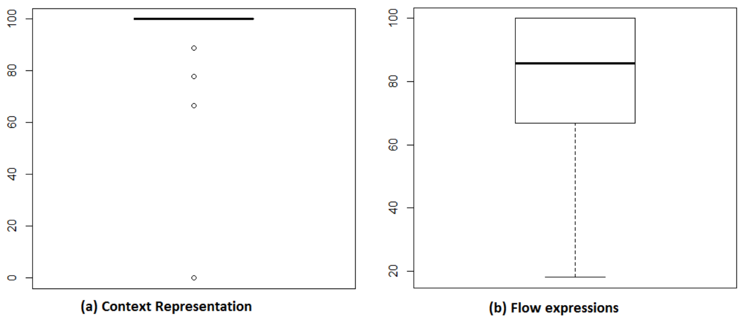

Figure 5. Correctness of context annotations and flow expressions.

However, after deeper investigation of the work, we did not find any justification for rejecting such observation. In relation to the specification of the flow expressions, we had none outliers.

5.2 RQ1: How do the subjects perceive the complexity of the contextual goal model and flow expressions?

The subjects classified the complexity of the notation for context specification using a Likert scale. The findings are shown in Table 1. The majority of subjects (63.6%) stated that it was of Reasonable complexity and 22.7% affirmed that it is Very Easy or Easy. In relation to the complexity of the flow expressions, 54.5% classified it as Reasonable and 31.8% as Very Easy or Easy.

Table 1. Complexity of the notations for context specification and flow expressions.

	Very Easy	**Easy**	**Reasonable**	**Difficult**	**Very Difficult**
Notation for context specification	1 (4.5%)	4 (18.2%)	14 (63.6%)	3 (13.6%)	0
Flow expressions	1 (4.5%)	6 (27.3%)	12 (54.5%)	3 (13.6%)	0

5.3 RQ2: How do the subjects perceive the complexity of the GO2S process steps?

We asked the subjects about different aspects of the GO2S process. The complexity of the process in general was classified by 72.7% of the subjects as Reasonable as depicted in Table 2. There were different opinions about the complexity of the Step 2, which corresponds to the specification of contextual variations points. 36.4% of the subjects stated that is Reasonable, although 31.8% classified it as Easy and 27.3% as Difficult. 45.5% subjects evaluated the complexity of the Step 3 (Specification of adaptation and monitoring) as Reasonable. On the other hand, 40.9% classified it as Difficult as presented in Table 2. The complexity of the Step 4 was evaluated by 59.1% as Reasonable and 27.2% as Very Easy or Easy.

Table 2. Complexity of the GO2S Process.

	Very Easy	**Easy**	**Reasonable**	**Difficult**	**Very Difficult**
GO2S Process	0	1 (4.5%)	16 (72.7%)	5 (22.7%)	0
Step 2 of GO2S	0	7 (31.8%)	8 (36.4%)	6 (27.3%)	1 (4.5%)
Step 3 of GO2S	0	3 (13.6%)	10 (45.5%)	9 (40.9%)	0
Step 4 of GO2S	1 (4.5%)	5 (22.7%)	13 (59.1%)	2 (9.1%)	1 (4.5%)

5.4 RQ3: What are the benefits stated by the subjects of the contextual goal model and flow expressions as well as the steps of the GO2S process?

In relation to the subjects' opinion about the following statement: "*The use of flow expressions makes the creation of statecharts more systematic*", 86.4% of the subjects agree (68.2% or Totally agree - 18.2%) with this statement. These findings are shown in Figure 6.

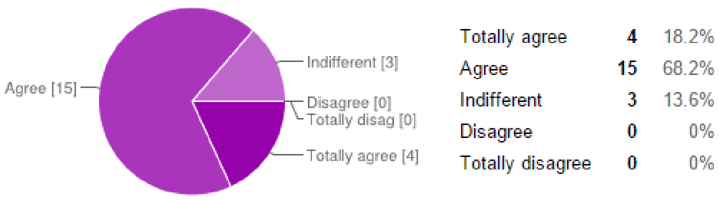

Figure 6. Flow expressions makes the creation of statecharts systematic.

The GO2S process is still an emerging methodology for derivation of the behavior of context-sensitive systems, and many benefits have not yet been fully realized or evaluated.

Hence, we analyzed which benefits of the GO2S process are perceived by the subjects, with the following open-ended question: "*In your opinion, which is the main benefit of this process?*" We extracted some codes from the subjects' answers (see Figure 7) using the coding technique. Coding is an essential part of the analysis when we are dealing with qualitative information.

We identified seven categories of benefits: accuracy of the statecharts specification, understanding the system's behavior, organization of the generated statecharts, requirements elicitation, the systematization of the derivation process, facilitates understanding of the problem domain and the system, mapping between tasks and states.

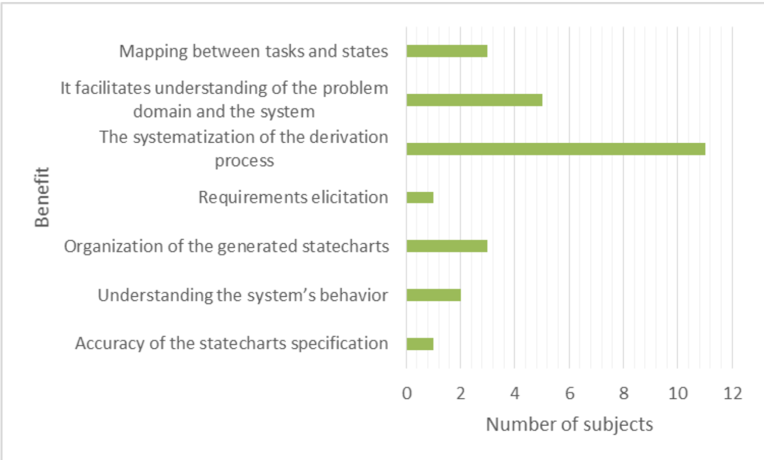

Figure 7. Benefits of the GO2S Process.

We already expected some of these benefits since they constitute the motivation for the process. However, the benefit for requirements elicitation was a surprise for us, since it is not the GO2S purpose. It was assumed that the software engineer had already performed this phase. Hence, a goal model was available as an input to the process. There are some studies such as [14] dedicated to investigate the problem of context elicitation.

5.5 RQ4: What challenges are related to the use of the GO2S process?

The subjects reported some difficulties and challenges about the use and specification of the GO2S process as well as training time. The majority (68.2%) said that they had more difficulty in step 2 (specification of contextual variation points) followed by step 3 (specification of adaptation and monitoring) with 54.5% while 50% had difficulty in step 5 (statechart derivation and refinement). These findings are depicted in Figure 8.

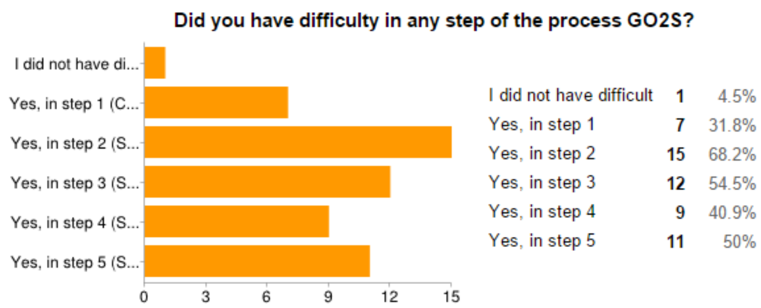

Figure 8. Difficulties in the use of the GO2 process.

The subjects also reported some difficulties regarding the study execution such as little experience in the use of the process, the identification of tasks to be performed at each step of the process, little time of training and the complexity of some steps of the GO2S process. The codes of the difficulties described in the subjects' opinions about the study execution are presented in Figure 9.

One subject complained about not making the material of the training available before the study. We made this call since we want all subject to have the same training time and level of knowledge, so we could draw reliable conclusions and avoid maturation effects. In the next section, we discuss threats to validity.

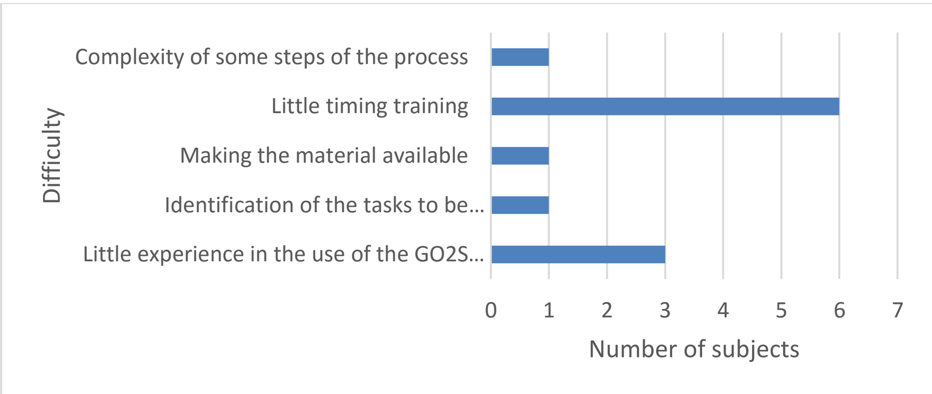

Figure 9. Difficulties reported in the study execution.

6 THREATS TO VALIDITY

We will base our discussion of threats to validity on the threat categories used by Wohlin et al. [12]. Due to space limitation we will only go into details for the threat categories that we consider most important for our study.

The conclusion validity is concerned with threats against being able to draw the correct conclusions about the relationship between treatment and observations. In relation to the statistical power, it should be noted that the purpose of the study was not to perform statistical testing of hypotheses, as this study is qualitative in nature.

The internal validity concerns the threats due to something else than the treatment influencing the observations. We minimized threats related to instrumentation due to bad forms, diagrams, score sheets etc. since the questionnaires had a straightforward layout, common to the questionnaires used in most cases where the Likert scale is used.

The construct validity addresses if the study be able to reflect the real world phenomena that we want to observe. We minimized threats related to instructions to the subjects since all of them got the same training with the same material and same instructor. Hence, differences in the study results can thus not stem from the instructions.

The external validity is related to the ability of generalization of the study results. We acknowledge that the limited number of subjects does not allow generalization outside the scope of the study. On the other hand, we expect that the results, including the subjects' feedback, can be used as guidelines to better improve the GO2S process. Besides, we believe that they are considerable contributions. This study can guide other studies in order to evaluate alternative approaches for context and behavior specification.

7 DISCUSSION

In this paper, we present the design and the findings of an empirical study conducted to understand the complexity of the notations for context and behavior specifications as well as the steps of the GO2S process. The study used a self-administrated questionnaire for collecting data from the subjects [17].

The questionnaire had a mix of close-ended and open-ended questions [17]. The close-ended questions have as advantages the efficiency and the unambiguity. The possible answers for the close-ended questions were exhaustive and mutually exclusive since we want the subjects to check just one answer. We used open-ended items for those aspects that we want to get more feedback from the subjects. Besides, we made short questions and tried to avoid loaded and double-barreled questions [17]. A double-barreled question is a single question asking about more than one issue, but that only allows for one answer. Moreover, we gave some training to the subjects so they to understand and answer the questions we want to investigate.

The research findings showed that 97% of the subjects identified appropriately the context annotations and 81% the flow expressions correctly. Moreover, according to the subjects' answers, the notations as well as the GO2S process have a reasonable perceived complexity.

Moreover, the subjects reported some benefits of the GO2S process such as the mapping between tasks and states. They also indicated that it facilitates the understanding of the problem domain and the system, the systematization of the derivation process. Moreover, it contributes to the requirements elicitation, organization of generated statecharts, as well as to understanding the system's behavior. Last but not least, it improves the accuracy of the statecharts specification.

The GO2S has a certain degree of complexity since it requires knowledge of several different notations such as goal models, context annotations and refinements, flow expressions and statecharts. Hence, the subjects classified the complexity of GO2S as reasonable.

Although the subjects were concerned with its reasonable complexity, such classification may be due to the little time available for training the subjects and execution of the study. Moreover, it is important to highlight that the subjects had no contact with these notations before this study.

A common issue with the reporting of empirical studies is the publication bias. This concept refers to the problem that positive results are more likely to be published than negative results [13]. However, the concept of positive or negative results sometimes depends on the viewpoint of the researcher. Accordingly, it is important to report difficulties such as little time for training and reasonable complexity in proposals, so that improvements can be investigated and improved in future works. Therefore, we believe that some results presented in this paper could not be interpreted as a drawback for adopting the process.

Finally, further replication by independent groups is necessary, and also new studies must be carried out with larger samples of subjects.

8 CONCLUSIONS AND FUTURE WORK

In this paper, we presented the findings of an empirical study conducted with twenty-two professionals to analyze the complexity of context annotations presented in goal models, along with flow expressions that specify the behavior of context-sensitive systems. Moreover, the steps of the GO2S process were also analyzed.

The findings presented in this study can be useful to the software engineering community, since it gathers evidences of the feasibility and of the benefits of using the context annotations of goal models to specify the context of context-sensitive systems. Besides, the subjects reported significant benefits in using flow expressions and the GO2S process to specify the behavior of context-sensitive systems, such as systematization of the process, better understanding of the problem domain and of the system itself; and good organization of the generated statecharts.

As future work, we expect to develop a case tool to support the process. It could be used to produce the goal model as well as to guide the application of the GO2S process, generating its different views (design, contextual and behavioral) of our process and implementing the statechart derivation. A prototype tool with some of these functionalities has already been developed [15].

Finally, the analysis of properties of the statecharts generated with the GO2S process is necessary. These properties can include completeness, correctness, deadlocks, and state reachability, as well as the satisfaction of quality attributes. We expect to be able to perform some of these analyses using ontological mechanisms, considering their suitability for requirements engineering as identified in previous work [16].

Acknowledgments

The following Brazilian institutions have supported this work: FACEPE, CAPES and CNPQ. The authors would like to express their gratitude to all those involved in this study.

References

1. V. Vieira; P. Tedesco; and A. C. Salgado. Designing context-sensitive systems: An integrated approach. In Expert Systems with Applications vol. 38, n. 2, pp. 1119-1138, 2011.
2. R. de Lemos et al. Software engineering for self-adaptive systems: A second research roadmap. In: Software Engineering for Self-Adaptive Systems II. Springer Berlin Heidelberg, pp. 1-32, 2013.
3. R. Ali; F. Dalpiaz; and P. Giorgini. A goal-based framework for contextual requirements modeling and analysis. Requirements Engineering, v. 15, n. 4, pp. 439-458, 2010.
4. J. Vilela; J. Castro; J. Pimentel. Deriving the behavior of context-sensitive systems from contextual goal models. In: Proceedings of the 30th SAC, 2015, pp. 1397-1400
5. J. Vilela; J. Castro; J. Pimentel, P. Lima. On the behavior of context-sensitive systems. In: Proceedings of the 18th WER, 2015, pp. 632-645.
6. R. Ali. Modeling and Reasoning about Contextual Requirements: goal-based framework. 2010. PhD Thesis, Universita degli Studi di Trento.
7. J. Pimentel et al. From requirements to statecharts via design refinement. In: Proceedings of the 29th Symposium of Applied Computing (SAC), pp. 995–1000, 2014.
8. A. C. Shaw. Software descriptions with flow expressions. IEEE Transactions on Software Engineering, n. 3, pp. 242-254, 1978.
9. F. Dalpiaz; A. Borgida; J. Horkoff; J. Mylopoulos. Runtime goal models: Keynote. In Seventh International Conference on Research Challenges in Information Science (RCIS), pp.1-11, 2013.
10. E. Sikora; B. Tenbergen; K. Pohl. Industry needs and research directions in requirements engineering for embedded systems. Requirements Engineering, v. 17, n. 1, p. 57-78, 2012.
11. H. R. Bernard. Social research methods: Qualitative and quantitative approaches. Sage, 2012.
12. C. Wohlin et al. Experimentation in software engineering: An introduction. Norwell, MA: Kluwer Academic, 2000.
13. B. Kitchenham; S. Charters. Guidelines for performing systematic literature reviews in software engineering. In: 2.3 EBSE Technical Report. EBSE. 2007.
14. C. A. T. Batista and C. T. L. L. S. Schuenemann. Um Processo Criativo de Elicitação de Contextos para Sistemas Sensíveis ao Contexto. In: Anais do XI Simpósio Brasileiro de Sistemas de Informação (SBSI). 2015.
15. J. Pimentel; J. Vilela; J. Castro. Web tool for Goal modelling and statechart derivation. In: 23rd International Requirements Engineering Conference (RE), 2015, pp.292-293.
16. D. Dermeval, J. Vilela, Ig Bittencourt, J. Castro, S. Isotani, P. Brito, A. Silva. Applications of ontologies in requirements engineering: a systematic review of the literature. Requirements Engineering, pp. 1-33, 2015.
17. H. R. Bernard. Social research methods: Qualitative and quantitative approaches. Sage, 2012.