

# Controlled Experiment in Crosscutting Concerns Identification from Software Requirements Specification

Guilherme Legramante<sup>1</sup>, Maicon Bernardino<sup>1</sup>, João Pablo Silva<sup>2</sup>,  
Elder Rodrigues<sup>1</sup>

<sup>1</sup> Laboratory of Empirical Studies in Software Engineering  
Universidade Federal do Pampa (UNIPAMPA)  
Caixa Postal 15.064 – 97.546-550 – Alegrete – RS – Brazil

<sup>2</sup>Laboratory of Intelligent Software Engineering  
Universidade Federal do Pampa (UNIPAMPA)  
Caixa Postal 15.064 – 97.546-550 – Alegrete – RS – Brazil

guilhermelegramante@gmail.com,

{bernardino, joaosilva, elderrodrigues}@unipampa.edu.br

**Abstract.** *The concern is a group of requirements with the same purpose, which are known as crosscutting concerns when they are scattered and tangled in the system. Identifying and separating these concerns is a matter of huge importance to software maintainability and evolution. For this, some approaches are proposed. There are few experimental studies comparing and analyzing these approaches in detail. In this paper, our aim is to provide empirical evidence about two approaches in this context. We conducted a controlled experiment to compare the effectiveness of two approaches, which identify crosscutting concerns in the requirements specification, followed by an assessment of its perceived utility and ease of use. Our results indicate that, in this given context and planned scenario, there are significant differences between the effectiveness of ObasCIId and Theme/Doc approaches.*

**Resumo.** *Interesse é um grupo de requisitos com o mesmo objetivo, conhecidos como interesses transversais quando dispersos e emaranhados no sistema. Identificar e separar esses interesse é uma questão de enorme importância para a manutenção e evolução do software. Para isso, algumas abordagens são propostas com essa finalidade. Há poucos estudos experimentais analisando essas abordagens em detalhes. Este artigo busca fornecer evidências empíricas sobre duas abordagens nesse contexto. Foi realizado um experimento controlado para comparar a eficácia de duas abordagens que identificam interesses transversais na especificação de requisitos, seguido de uma avaliação da sua utilidade percebida e facilidade de uso. Os resultados do experimento indicam que, nesse contexto e cenário planejado, existem diferenças significativas entre a efetividade das abordagens ObasCIId e Theme/Doc.*

## 1. Introduction

Software requirements define the features and constraints of a given software product. These characteristics may contribute to the solution related to a real-world problem [Bourque et al. 2014]. The concern is a set of software requirements with the

same purpose [Parnas 1972]. Dijkstra [Dijkstra 1976] describes some concepts within this scope, such as concern separation, cohesion, and coupling in the context of Software Engineering by introducing notions of modularization. The Standard Glossary of Software Engineering Terminology [IEEE 1990] defines modularization as the degree of modifications required in software components that are caused by changes in another component. Therefore, a good modularization could bring benefits like reusability, comprehension, adaptability, maintainability [Meyer 1997]. For this reason, software engineers consider modularization as a key principle [Santanna 2008]. A cross-cutting concern is known as a scattered and tangled concern, by transversely intersecting the other software structures [Kiczales et al. 1997].

To decrease the effects caused by scattering and tangling of crosscutting concerns, the Software Engineer must identify them as early as possible in the software development life cycle [RESENDE 2007]. Nonetheless, this is a non-trivial activity, due to many concerns are implicit in the requirements document. According to Baniassad *et al.* [Baniassad and Clarke 2004], Software Engineers find it difficult to identify crosscutting concerns and two causes may be associated with these problems. The first one refers to a lack of understanding regarding the domain of software concerns. The second cause is related to the scarcity of appropriate resources to support software engineers software concerns identification and classification. Although there are several techniques and approaches for crosscutting concerns identification, there are few studies that perform empirical evidence on these approaches. So, in this paper, our objective is to provide empirical evidence about relevant studies in this context. To achieve the given objective, we conducted a controlled experiment to compare the effectiveness of the two approaches to identify crosscutting concerns in the requirements specification, followed by an assessment of its perceived utility and ease of use. We chose the selected approaches after the conduction of a Systematic Literature Review, which was the main aim of another study that we carried out before this experiment, and that we will not explain it in this paper.

We believe that the main contribution of our study to this field is the outcomes of the conducted empirical study, which were supported by well-defined and replicable protocol. In this context, the controlled experiment results provide empirical evidences that in a specific background there are significant differences between the effectiveness of the ObasCIId [Parreira Jr. 2018] and Theme/Doc [Baniassad and Clarke 2004] approaches. Furthermore, we managed to accomplish a qualitative assessment that provided us with interesting and relevant traces of qualitative perceptions in the proposed study.

The paper is organized as it follows. Section 2 presents the controlled experiment, which is one of the core parts in the present study, and finally, in Section 3, show final remarks.

## **2. Experimental Study**

In this experiment we followed the model by Wohlin [Wohlin et al. 2012].

### **2.1. Scope Definition**

The experiment aims to compare two approaches for identifying crosscutting concerns from Requirements Engineering artifacts. Thereunto, main objective is described according to the GQM (Goal, Question, Metric) paradigm [Caldiera and Rombach 1994]:

For the purpose of: **comparison**, with respect to **effectiveness**, from the viewpoint of **researchers**, in the context of **Software Engineering undergraduate identifying cross-cutting concerns in software requirement specification**

## 2.2. Experiment Context

Regarding the experiment context, we use the *in-vitro* approach since we perform the experiment in the laboratory under controlled conditions. The subjects were undergraduate students from Software Engineering, second year. The experiment compared the effectiveness of the `ObasCIId` and `Theme/Doc` approaches for crosscutting concerns identification, addressing a real problem. The experiment was applied in a specific context, with Software Engineering students and two approaches of crosscutting concerns. However, general assumptions regarding the controlled experiment may be applied in other populations and approaches.

In order to implement those approaches, the software requirements specification of the Course Management System proposed by Baniassad *et al.* [Baniassad and Clarke 2004] and `ObasCIId Tool` proposed by Junior and Penteadó [Parreira Jr. 2018] were used as inputs. The software has a specification that allows identification and also already has the concerns identified and cataloged. Also, they are equivalent in complexity level as a means of having the same number of concerns.

## 2.3. Research Hypotheses

The mainstay for the statistical analysis of an experiment is the hypothesis test. The hypothesis is statistically formulated and the data was collected during the experiment conduction to, if possible, reject the hypothesis. If we could reject the hypothesis, We can infer some conclusions.

For hypotheses formulation, we analyze the following research question:

**RQ. Which of the approaches for identifying crosscutting concerns in the software requirements specification is most effective?**

The experiment has the following hypotheses <sup>1</sup>:

**Null hypothesis**,  $H_0 = E_{OC} = E_{TD}$ : The effectiveness is the same when using `ObasCIId` e `Theme/Doc` approaches.

**Alternative hypothesis**,  $H_1 : E_{OC} > E_{TD}$ : The effectiveness is lower when using `Theme/Doc` than when using `ObasCIId`.

**Alternative hypothesis**,  $H_2 : E_{TD} > E_{OC}$ : The effectiveness is lower when using `ObasCIId` than when using `Theme/Doc`.

Furthermore, it is possible to evaluate the results through precision and recall metrics. Hence, Precision is the fraction of the documents already examined that are relevant, and Recall is the fraction of the relevant documents observed among the examined documents [Monteiro 2017]. Thus, to calculate Precision and Recall metrics, some variables are used:

**True Positives (TP)**: Concerns correctly identified using the approach.

---

<sup>1</sup> $E_{OC}$ : `ObasCIId` Effectiveness and  $E_{TD}$ : `Theme/Doc` Effectiveness

**False Positives (FP):** Concerns incorrectly identified using the approach.

**False Negatives (FN):** Concerns correctly not identified using the approach.

In our context, the effectiveness was inferred from the calculation of *F-measure* or *F-Score*, which is the harmonic mean of precision and recall metrics.

- *Precision (P):*  $P = \frac{TP}{TP+FP}$
- *Recall (R):*  $R = \frac{TP}{TP+FN}$
- *F-measure (F):*  $F = 2 * ((P * R)/(P + R))$

## 2.4. Subjects Selection

We selected the experiment subjects by non-probabilistic sampling, where there is a deliberate choice, for convenience. Thirty-six (36) undergraduate students from the second year in Software Engineering course at University participated in the controlled experiment. They were students enrolled in Software Verification and Validation disciplines, Experimental Software Engineering, and Problem Solving VI. The subjects were chosen because they composed a representative sample of the software engineers population since they have knowledge related to the area and already perform activities related to software engineering even though in academia.

## 2.5. Experiment Design

The four general principles for the experiment are as follow:

**Standard Design Types:** The experiment applied the pattern of *One Factor with Two Treatments*. The Factor is the approach for identifying crosscutting concerns and Treatments are the ObasCId and Theme/Doc.

**Blocking:** Thus, we applied a leveling questionnaire so that it was possible to classify the subjects.

**Balancing:** We divided the selected subjects into two homogeneous groups.

**Randomization:** We allocated the subjects randomly placed into each group and each approach.

## 2.6. Instrumentation

The experience level of the subjects was obtained through a questionnaire<sup>2</sup>, in order to assess the knowledge level in the area that is the object of the study. With this information it was possible to identify a subject profile.

*Objects:* among the objects used in the experiment are the requirements documentation of the software ObasCId Tool, of the Course Management System and the training guidelines of the subjects. The training sessions were available in videos divided in 3 parts. The first training session addressed main concepts on Requirements Engineering and Aspect-Oriented Requirements Engineering in order to level the subjects' knowledge. The second and third training sessions explain the Theme/Doc and ObasCId approaches. We provided a printed document containing all the material to each student during the video lessons and was also a possibility to be used by them. The

---

<sup>2</sup>Questionnaire: <https://bit.ly/2wRInwF>

requirements documentation was properly prepared for the identification of software concerns according to the approaches used in the task. Since participation in the experiment was voluntary, a consent form was presented to the subjects, formalize their agreement to participate in the experiment. Requirements templates were also elaborated for listing the concerns that should be identified by the subjects.

*Guidelines:* We conducted three sessions of the experiment divided into two stages, training and execution. During the training stage, the subjects received instructions on Software Concerns and Requirements Engineering, then instructions on the approaches to be performed. We defined randomly the order of the instructions on the approaches and remained the same throughout all sessions of the experiment.

*Measures:* The quantitative metrics encompass relevant elements, selected elements, true positives, false positives, and false negatives. All subjects performed the same tasks under the same conditions, except when sessions were on different days, but with no significant differences.

## 2.7. Operation

The first activity was to contact the course professors for the sessions. After their agreement, we applied a questionnaire to assess the knowledge level of research subjects. From this, we classified the students into two levels: basic and intermediate. We sent all experiment artifacts for validation with a specialist, Aspect-Oriented Requirements Engineering Ph.D. We made all the corrections suggested by the specialist. The experiment execution has three sessions that occurred on consecutive days, September 11, 12, and 13, 2018, respectively. For this stage, we invited the students of the Software Verification and Validation - V&V (Session 1), Experimental Software Engineering - ESE (Session 2), and Problem Solving VI - PSVI (Session 3) courses to attend. It stands out that the disciplines mentioned are part of the Software Engineering curriculum of the university where the research took place.

Table 1 displays the information about the experiment sessions. The experiment sessions are in the first column, followed by the executed disciplines, the number of subjects that participated with the blocks by basic and intermediate levels, and the date of execution. In Session 1, two participants had the results disregarded, as long as they did not follow the recommendations regarding the non-use of electronic devices and did not avoid communicating with the other subjects during the experiment execution.

**Table 1. Experiment Sessions.**

Session	Course	Subjects	BL	IL	Date
1	V&V	20	18	2	11/09/2018
2	ESE	12	4	8	12/09/2018
3	PSVI	4	2	2	13/09/2018
<b>Total</b>		<b>36</b>	<b>24</b>	<b>12</b>	

**BL:** Basic Level, **IL:** Intermediate Level

The first activity Consent Term<sup>3</sup> provision for experiment execution. The subjects who signed the term were selected to participate in the other activities. We informed the

<sup>3</sup>Consent Term and study artifacts available in: <https://bit.ly/2wRInwF>

subjects about the "non-evaluative" character of the experiment, *i.e.* that the Subjects were not under evaluation. Afterward, we included the subjects in two homogeneous groups, A and B, according to the result of the questionnaire previously applied. In the first stage, we show the video Software Concerns Separation training. This activity lasted approximately 12 minutes. Then, we display the Theme/Doc approach training, after a random choice. This activity lasted for approximately 9 minutes. The last activity of the training was the presentation of the video about the ObasCIId approach, which lasted approximately 12 minutes. It is important to emphasize that the difference in duration between the training sessions of the approaches is due to the differences between them because the ObasCIId approach requires more activities to be carried out to identify the software concerns and for this reason, it has impacted training duration.

At the end of the training phase, the execution stage began. For this, the subjects received the necessary artifacts to perform the tasks. The main activity to be undertaken was to identify software concerns in the software requirement specification, using the approaches. In the former part of the experiment execution, Group A should use the ObasCIId approach and Group B, the Theme/Doc approach. In the latter part of the experiment execution, Group A used the Theme/Doc approach and or Group B to ObasCIId. Therefore, we provided the material presented in the training sessions to the subjects in a printed document. Also, Group A received the software requirement specification (SRS) of the Course Management System and the Software Concerns Catalog. For Group B, the SRS of the OC Tool was also delivered. In the latter part of the experiment execution, Group A received the SRS of the OC Tool and Group B, the SRS of the Course Management System plus the Software Concerns Catalog. The subjects answered the questionnaire about the perception of its utility and ease of use after performing the concerns identification. The three sessions occurred in the same way, following the same activities in the same execution order.

## **2.8. Threats to Validity**

According to Cook [Cook and Campbell 1979], threats are categorized into: Conclusion, Internal, Construct, and External Validity.

### **2.8.1. Conclusion Validity**

It encompasses questions related to results analysis, *i.e.* whether the conclusions reached are correct. The Paired T-test was used to compare the measured values. The *Shapiro-Wilk* test was applied to verify the normality of the data. The fact that there is no relationship between the approaches analyzed and our research, that help us to mitigate this threat. Objective measurements that do not depend on the subjective judgment will be made because the effectiveness of the approaches will be measured. The qualitative metrics collected to verify the perceived utility and ease of use only complement the quantitative metrics, not being the main objective of measurement. We standardize procedures for all subjects. The sessions of the experiment followed the same execution order, having the same artifacts and duration time. The experiment was performed in a controlled environment, avoiding external interactions to experiment, such as interruptions, the exit of the experiment environment, access to electronic devices, etc. We advised the participants to

ban the use of electronic devices and parallel conversations during the experiment. No participant was allowed to leave the experiment environment during the execution.

### **2.8.2. Internal Validity**

We performed the experiment at a time when students were not too overwhelmed with semester projects, papers, or exams. We validated all the artifacts by a Ph.D. specialist in the Aspect-Oriented Requirements Engineering area. We do not select participants with any type of involvement in similar experiments, even the sample used in the pilot experiment was discarded for the real experiment.

### **2.8.3. Construct Validity**

As the experiment follows a paired design, all subjects performed the two treatments. However, this fact was not a threat because there was no learning between the approaches execution. For learning verification, we analyzed experiment data as a random design and, the results remained similar. Hence, we verified that there was no evidence of learning between the execution of the approaches. We do not inform the subjects were about the experiment design details. Also, we inform that the experiment would have no impact on students' grades and that they were not under evaluation but participating in an experiment. Subjects were not provided detailed information about the experiment, only the instructions were given on the tasks to be performed.

### **2.8.4. External Validity**

We performed the study with Software Engineering students who represent a significant sample for the area since the primary responsibility for identifying software concerns in a project is the Software Engineer. However, the fact of carrying out the experiment with students is considered a threat, which could not be mitigated. We used documentation of traditional software requirement specifications found in the industry. Furthermore, the artifacts were validated with a Ph.D. specialist in the area.

## **2.9. Results**

After the subjects of the experiment performed the identification of concerns activities, the correction stage was passed, where concerns identified by the subjects and concerns presented in the template were confronted. The correction was conducted in pairs as a means to attaining a consensus. Figure 1 displays the results of the experiment. The red boxes refer to the ObasCIId approach and the blue boxes show the results of the Theme/Doc approach and also show the results according to the levels: basic and intermediate. It is possible to notice that essentially the Theme/Doc approach obtained better rates for the *F-measure*, regardless of the blocking performed. However, the difference between treatments is lower among subjects classified at the intermediate level. The ObasCIId approach has a smaller variability compared to Theme/Doc. The lower limit for the basic level is quite similar, 0.20 for ObasCIId and 0.22 for Theme/Doc, at the intermediate level the difference is larger, 0.33 for ObasCIId and 0.72 for Theme/Doc.

The group that presents the greatest variability between the first and third quartile is the basic level in regard to Theme/Doc approach. The least dispersed group is the intermediate level that executed Theme/Doc approach, amidst a difference of 0.22 between the first and third quartile. This group also presented the highest value for the lower limit, 0.72. A further relevant aspect is that at all levels and treatments the upper limit was the maximum possible value, which means that in all groups there were subjects who were able to identify all the interests presented in the software under analysis. Based on the data reported, we verified that both treatments obtained better results with groups with greater experience in Requirements Engineering and Aspect-Oriented Requirements Engineering, and the treatments did present significant differences to their effectiveness.

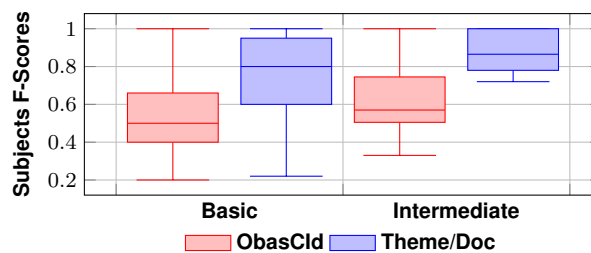
## 2.10. Hypothesis Testing

According to [Graybill et al. 1998], to be able to accept or reject a particular hypothesis, it is necessary to perform a procedure called Hypothesis Test or Significance Test. The first procedure was the analysis of experiment samples to verify the normality among data distribution. For this task, we used the *Shapiro-Wilk* test, in which one tries to reject the null hypothesis to infer the normality of the data [Wohlin et al. 2012]. Table 2 displays the values of  $W$  and  $p$ -value obtained with the *Shapiro-Wilk* test for the experiment. As  $p$ -value = 0.05 and  $W_{(calculated)} \geq W_{(0,05; 10)}$ , it is possible to state with a significance level of 5% that the sample comes from a normal population, that is, there is 95% chance of the sample being normal.

**Table 2. Shapiro-Wilk Test Results**

	Session 1	Session 2	Session 3	Total
<b>W</b>	0.9711	0.9616	0.8935	0.9279
<b><i>p</i>-value</b>	0.7789	0.8064	0.3999	0.2013

Based on the results of the normality test, we have chosen to perform the *Paired Sample T-Test* to check the  $H_0$  of the experiment. Applying this test it was obtained a  $p$ -value =  $2.001e-05$  ( $p$ -value < 0.05), *i.e.* it was possible to reject  $H_0$  and state that there are differences between effectiveness of the ObasCIId and Theme/Doc approaches. According to the data displayed in the Figure 1, we may accept that the  $H_2$ : Theme/Doc approach has more effectiveness than the ObasCIId approach. Presented results were obtained for a given sample from a specific population under a given context. Hence, their data do not depict a representative sample and cannot be generalized.



**Figure 1. Experiment results.**



## 2.11. Perceived Utility and Ease of Use

Qualitative metrics were collected regarding the perceived utility and ease of use of the approaches. These metrics were not included in the protocol presented in Section 2, because they are subjective. The reference model used to formulate the questions presented in the questionnaire was the TAM (Technology Text Acceptance) Model, proposed by Davis [Davis 1993]. The questionnaire was organized in 12 questions, 6 for each of the approaches under analysis. Figure 2 shows questionnaire results.

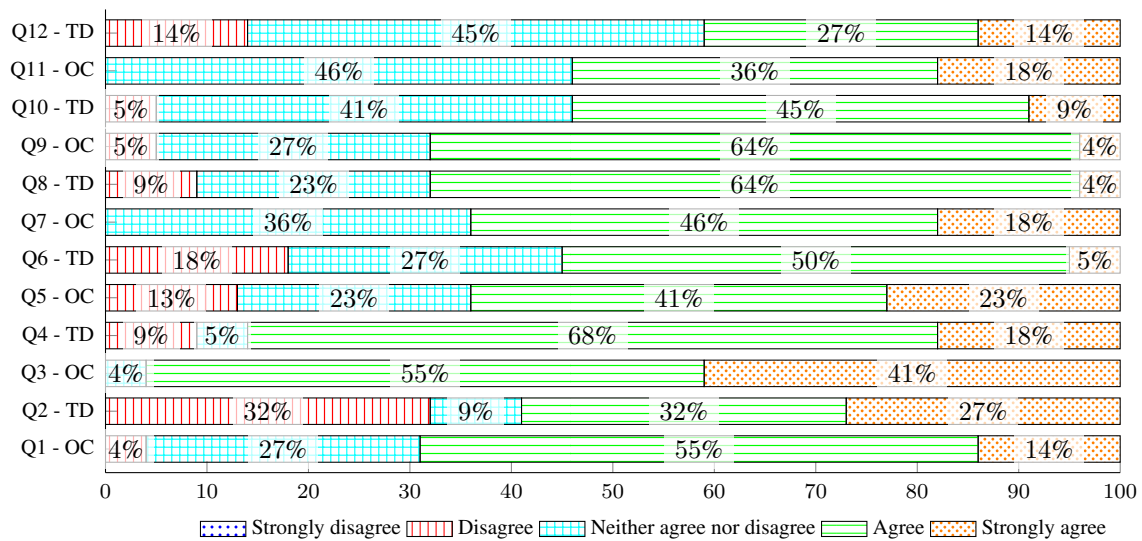


Figure 2. Questionnaire Results

The following are the questions that we use: - The approach is easy to use; - The approach is useful for identifying and classifying crosscutting concerns in requirements documents; - I enjoyed working with the approach; - Using the approach can increase my performance during work related to the identification and classification of crosscutting concerns; - The approach produces the results I expect from an approach to identifying and classifying crosscutting concerns; - I will recommend using the approach.

To answer the questionnaire the subjects should choose one of 5 possible answers: Strongly Disagree, Disagree, Neither Agree nor Disagree, Agree, Strongly Agree. In general, the ObasCId approach presented better results because it shows a higher percentage of responses that obtained total agreement, that is, answers marked with “Strongly Agree”. Both approaches were well accepted, according to questions Q6 and Q12, which refer to using the approaches recommendation. The Theme/Doc approach had better results in the ease of use questions, questions Q1 and Q7, obtaining 27.3% of responses *Strongly Agree* versus 13.6% of ObasCId. In questions Q5 and Q11, regarding the expected results, although the Theme/Doc approach obtains a higher result of “Strongly Agree” answers (9.1%), it had an expressive number of subjects that remained neutral, answering “I Neither Agree nor Disagree” and the fact that 63.6% of subjects respond “I Agree” to the same question, but related to the approach ObasCId, may represent that the latter had better expected results.

### 3. Final Remarks

Some general considerations about obtained results. With the Theme/Doc approach, the crosscutting concern participants noticed greater difficulty in identifying were the ones related to usability, a non-functional requirement. Therefore, in our experiment, the Theme/Doc approach proved to be more effective for identifying concerns, not necessarily crosscutting concerns. The difficulty in executing the Theme/Doc approach may have been part of Action View modeling, which requires a certain understanding and special attention to its creation to be correctly modeled.

Regarding the ObasCId approach, the crosscutting concern with a lower number of correct answers of the participants was related to usability in the OCT requirements and student selection in CMS specification. The greatest difficulty encountered in implementing this approach was related to support from the concerns catalog since it induces some participants to mark false positives, as the catalog contains some concerns that might not necessarily be mapped.

Based on our experiment results, we believe that Theme/Doc approaches are due to some characteristics that make it slightly simpler to be applied than the ObasCId approach. Perhaps the catalog of concerns, available at ObasCId, despite being useful, induces the participant not yet familiar with the approach, which was our case, to elicit some false positives. However, we emphasize that for our context, with our representative portion of the selected population, these are the comparative evidence found. Moreover, we did not intend to refute or replicate the experiment carried out by Junior and Penteadó [Parreira Jr. 2018], since we have different protocols and objectives.

The general objective of this paper was to provide empirical evidence about relevant studies in this context. ObasCId and Theme/Doc approaches were evaluated through an experiment, as means to measure and compare their effectiveness. The main contributions of this work are the outcomes of the empirical study conducted, which were supported by well-defined and replicable protocol. Obtained empirical evidence illustrates that, in a given scenario and context, there are significant differences between selected approaches effectiveness. The results obtained through qualitative metrics are related to perceived utility and ease of use of approaches listed above, which ones might also be elicited as a contribution to this research area. To the best of our knowledge, no studies were found in the Literature, so far, performing these measurements given the presented context and scenario. As further research, we would suggest extending the results of the controlled experiment performed with a different sample, for instance, by replicating it in an industrial environment, since we applied our experiment in an academic environment. It is also possible to conduct empirical studies in the area of crosscutting concerns in other stages of the software development life cycle.

### References

- Baniassad, E. and Clarke, S. (2004). Finding aspects in requirements with theme/doc. *Early Aspects: Aspect-Oriented Requirements Engineering and Architecture Design*, pages 15–22.
- Bourque, P., Fairley, R. E., et al. (2014). *Guide to the software engineering body of knowledge (SWEBOK (R)): Version 3.0*. IEEE Computer Society Press.

- Caldiera, V. R. B.-G. and Rombach, H. D. (1994). Goal question metric paradigm. *Encyclopedia of software engineering*, 1:528–532.
- Cook, T. and Campbell, D. (1979). *Quasi-Experimentation: Design and Analysis Issues for Field Settings*. Houghton Mifflin.
- Davis, F. D. (1993). User acceptance of information technology: system characteristics, user perceptions and behavioral impacts. 38(3):475 – 487.
- Dijkstra, E. W. (1976). *A Discipline of Programming*. Prentice Hall.
- Graybill, F., Iyer, H., and Burdick, R. (1998). *Applied Statistics: A First Course in Inference*. Data Warehousing Institute Series from. Prentice Hall.
- IEEE (1990). Standard Glossary of Software Engineering Terminology.
- Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Lopes, C. V., Loingtier, J.-M., and Irwin, J. (1997). *Aspect-oriented programming*. Springer Berlin Heidelberg.
- Meyer, B. (1997). *Object-oriented Software Construction*. Prentice Hall.
- Monteiro, S. D. (2017). Sistemas de recuperação da informação e o conceito de relevância nos mecanismos de busca: semântica e significação.
- Parnas, D. L. (1972). *On the criteria to be used in decomposing systems into modules*. Communications of ACM.
- Parreira Jr., R. A. D. P. (2018). ObasCIId(-Tool): an ontologically based approach for concern identification and classification and its computational support. *Journal of the Brazilian Computer Society*, 24(1):3.
- RESENDE, A. M. (2007). *Um método para Identificação e Definição de Aspectos Iniciais*. 209f. PhD thesis, Tese de Doutorado-Instituto Tecnológico de Aeronáutica, São José dos Campos.
- Santanna, C. (2008). *On the Modularity of Aspect-Oriented Design: a concern-driven measurement approach*. PhD thesis, Centre for Telematics and Information Technology, University of Twente, Enschede, The Netherlands.
- Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., and Wesslén, A. (2012). *Experimentation in Software Engineering*. Springer, Norwell, MA, USA.