dr hab. Michał Śmiałek, prof. uczelni
Politechnika Warszawska
michal.smialek@ee.pw.edu.pl

Warszawa, 12.07.2021

Reviewer's opinion
on the PhD dissertation authored by
Tarek Alkhaeir
entitled:
Design patterns and code smells. Relationship and impact on selected software quality metrics

1. Overview of the dissertation

The dissertation has 88 pages and consists of 14 chapters (called "sections" within the dissertation) and references. It is written in good English with minor flaws which do not compromise understanding by the reader. Chapters 1-3 contain abstracts (in English and Polish) and the introduction. The introduction discusses the current status of research in the field under consideration but does not provide any formally specified research hypothesis or objective. Chapters 4 and 5 constitute a short introduction to design patterns and code smells and methods of their detection. These chapters basically contain a list of patterns and smells with their short descriptions and a list of detection techniques with descriptions and references. They also introduce two existing detection tools used within the dissertation.

Chapter 6 contains an overview of related work. This is divided into five sections that deal with research on various relationships between design patterns, code smells, changeability, and defects. Chapter 7 is a concise presentation of abbreviations used in the following sections. It can be noted that the list is not complete and is amended in specific sections of the following chapters.

Chapters 8, 9 and 10 constitute the main contribution of the dissertation. They present three experiments to determine relationships between design patterns, code smells, defects and changeability. The structure of each of these chapters is similar. They begin with the experimental design presentation with research questions (three per experiment) and input data sets. Then, they present details of specific statistical tests applied to the data sets to validate the hypotheses associated with each of the research questions. A discussion of the experiment results concludes this. It should be noted that Chapters 8 and 9 contain results presented already in 2 papers published in journals (Information and Software Technology and IEEE Access, respectively).

Chapters 11 to 14 provide summary remarks. This includes a summary conclusion, discussion on the dissertation's contribution, discussion on the validity of the experiments and a short discussion on possible future work.

The literature section contains 114 positions, all of them appropriately referenced in the text. However, vast majority of references are not complete, lacking, e.g. conference/journal titles, page numbers, publishers, or having inconsistent formatting.

From the editorial point of view, the dissertation is provided in A4 format, with the body of the text being in B5 format. Generally, formatting of text, figures and tables is correct. However, there can be found numerous flaws. This includes incorrect section numbering, tables extending onto the margins, broken references ([?]), inconsistent font sizes in tables, and others. More details on this are given in Section 6 of this review.

2. Problem and its impact

The dissertation treats an interesting and current topic of determining the relationships between design patterns, code smells, defects in code and changeability of code. Unfortunately, the overall problem is not specified clearly and distinctively within the dissertation (e.g. research hypothesis). Thus it has to be deduced from its contents, mostly including the introduction and the concluding chapters.

It is not the role of the reviewer to define the research hypothesis. However, during the review, I have assumed that the overall hypothesis is that there are statistically significant correlations between the four main variables of the dissertation: occurrences of design patterns, occurrences of code smells, number of code defects and code changeability. Furthermore, these correlations can lead to practical recommendations for software developers regarding design (e.g. usage of specific patterns) and maintenance (e.g. code refactoring) of new and existing software systems.

Considering the above, it can be stated that the problem discussed in the dissertation is definitely a scientific one. Its solution (finding the correlations) necessitates the usage of scientific methods (experimentation, statistical analysis). Moreover, the problem has a significant practical impact, as its solution can give software developers important recommendations that could alter software engineering practice.

The above considerations lead to the following remark regarding the dissertation.

> *Remark 1.* The overall problem (research hypothesis) of the dissertation should be clearly presented as a formal statement. Validation of the formulated hypothesis should be formally assessed in the concluding sections (cf. Section 10.4). This might involve, e.g. a table showing all the correlations found during the reviewed research. Moreover, the impact on software development practice should be clearly linked to the results of research. This means clear indications of which specific correlations found in research indicate which recommendations of changes in practice (cf. Section 12.2).

3. Contribution

The contribution of the dissertation is presented in Chapter 12. The research contribution can be summarised as providing a coherent study on the interrelations between design patterns and code smells and the effects these interrelations have on the quality of code (number of defects) and code changeability. Similarly to the problem statement, this summary is not provided in the dissertation. However, the individual contributions presented in Chapter 12 are identified correctly by the PhD candidate. Altogether, it can be stated that these individual contributions add up to validate the overall research hypothesis of the dissertation (although not articulated explicitly). They show that the studied correlations are statistically significant, leading to important recommendations for the software engineering practice.

Chapter 12 of the dissertation also presents contributions for the practice. These contributions are identified correctly. The presented research can be applied to design and programming practice and the further development of code metrics tools. In particular, the existing tools can apply the research results to extend their functionality by providing correlated metrics that involve both design patterns and code smells.

Considering the above, it can be stated that the dissertation constitutes a significant and original contribution to the field of software engineering in general and software metrics in particular. Though, it should be noted that the dissertation includes research results already presented in two research papers. This encompasses two of the three presented experiments. Thus, the dissertation can be treated in part as a thematically coherent series of articles. Both articles were published in well-respected journals (Information and Software Technology and IEEE Access) and have already a significant citation metric. On the other hand, the dissertation extends this 'series' with an additional significant experiment and provides a new edition of the previous work. It should also be noted that the mentioned two papers were

written by the PhD candidate together with the supporting supervisor. In one case, the supervisor was listed as the main author. Thus, it would be crucial to determine the actual PhD candidate's contribution to this research.

The above considerations lead to the following remark regarding the dissertation's contribution.

Remark 2. The overall contribution of the dissertation should be clearly presented and linked with the overall research hypothesis. The level of contribution of the PhD candidate in work published together with the supporting supervisor should be clearly explained.

4. Correctness

The three experiments that constitute the main contribution of the dissertation have a similar structure. First, the requirements for an experiment are formulated in the form of three research questions. The first two questions pertain to general relationships between the major metrics (number of design patterns, number of code smells, defects, changeability). The last question pertains to the relationships involving specific design patterns and code smells. Following this, the experiment is set up by selecting software systems to be analysed (2, 10 and 3 respectively). All these systems are taken from publicly and freely available source code repositories containing related data (e.g. bug reports). Then, appropriate tools are used to collect metrics from the analysed systems. This involves mostly existing tools (SSA system by Tsantalis and InCode) but also a script to collect changeability metrics developed especially for the third experiment. Based on the collected metrics, appropriate statistical analysis is then performed. For each of the research questions, a null hypothesis and alternative hypotheses are formulated. Typically, the normality of data distribution for the collected variables is determined through QQ-plots and/or Shapiro-Wilk test (the data were found not to be normally distributed in most of the considered cases). This is followed by various other statistical tests, depending on the particular hypothesis to be validated. Finally, the results of these tests are discussed, and certain conclusions are drawn. Threats to the validity of the results are presented for all the experiments in Chapter 13.

Generally, the experiment setup and results are sound and in line with the current research practice. The specific tests are appropriate, and the results are well justified. The discussion of results is satisfactory and presents valid interpretation of the results. However, certain remarks can be made regarding specific details.

Remark 3. Each of the presented experiments operates on a different set of software systems. It is not explained anywhere in the dissertation why is this the case. This is associated with two main issues. First, it is not explained why the particular set of systems is used for a specific experiment and not for the others. Second, it is not explained why the three experiments did not operate on all the systems used in the experiments. This might be justified, e.g. by lack of certain data in the sets (e.g. data on defects), but there is no clear explanation provided.

Remark 4. Experiment no. 1 is already five years old. Thus, it would be quite natural to extend the study with the newest data (new releases of the two systems). Currently, this is just a repetition of the older paper, while the reviewed dissertation could (should) offer more up-to-date results. This could be an important new contribution of the PhD candidate, going beyond the work authored with the supporting supervisor.

Remark 5. The dissertation does not present specific procedures for collecting code metrics. It mentions the tools used but there is no information on how these tools were used and what specific parameters of these tools were applied. This significantly impedes replication of these experiments. The dissertation could – for instance – present the process of downloading the source data and running the tools with specific parameters to collect the metrics specified in the experiments' setup.

5. Knowledge of the candidate

The reviewed dissertation treats three main areas of research – design patterns and their detection, code smells and their detection, and statistical analysis of code. The first two areas are discussed in Chapters 4 and 5, respectively. This discussion is somewhat limited, and consists of a presentation of a list of design patterns (only the GoF patterns) and code smells (indicated as "well known" ones). Moreover, some detection techniques and the two tools used in the experiment are briefly described.

Chapter 6 of the dissertation contains an overview of related work. This concentrates on presenting past research regarding five types of relationships that involve design patterns, code smells, defects and changeability. The selected literature is relevant and covers all the aspects of the dissertation, but seems limited to up to around the years 2018/19. A simple forward search on the presented literature shows newer relevant literature, like for instance: P. Piotrowski, L. Madeyski – "Software defect prediction using bad code smells: A systematic literature review" (2020), or K. Sultana, Z. Kodabux – "Examining the relationship of code and architectural smells with software vulnerabilities" (2020).

The dissertation does not contain any introduction to statistical methods. The methods used in Chapters 8-10 are indicated by their names and usually referenced with their main source. However, in some cases, only the method name is given (see, e.g. the mentioning of the ANOVA method on page 70). Moreover, the various variables in the statistical analysis (e.g. W, z, p, σ) are not explained. It seems to be assumed that the reader should always refer to the appropriate source, thus making the dissertation not self-contained.

In summary, it can be stated that the PhD candidate shows satisfactory knowledge and understanding of the relevant areas of the ICT discipline in general and Software Engineering in particular. However, this knowledge is not fully reflected in the dissertation through an elaborated presentation of appropriate areas of research and practice. Yet still, the overall contents of the dissertation shows that the PhD candidate has a good understanding of the field and can efficiently use relevant research techniques and current research results.

Considering the above, the following remarks can be formulated.

Remark 6. The dissertation lists several techniques for detecting design patterns and code smells. However, their descriptions are very brief and, in most cases, do not include any examples (references to literature, tools, etc.). Moreover, there is no elaborated comparison of the effectiveness of these techniques. Thus it would be very desirable to provide a more detailed discussion on the selection of techniques and rationale behind the choice of the techniques used in the dissertation.

Remark 7. The dissertation briefly mentions two tools used to detect design patterns and code smells. However, it does not provide any elaborated information on the tools and their alternatives. No discussion was made on why the particular tools were used, apart from very brief references to literature. Moreover, the dissertation does not provide any details on the tool to measure code changeability. It can be deduced from Chapter 12 (see page 78) that the tool constitutes some set of own developed scripts. Yet, there is no discussion on why such an approach was taken and what are the characteristics of the tool.

Remark 8. The dissertation lacks an elaborated presentation of the statistical methods that were used in the experiments. It would be very desirable to provide a detailed discussion on the characteristics of the methods and the rationale behind their selection (compared to other methods).

Remark 9. An update to the literature that includes newer research would be very desired. This should also involve updates to the related work chapter.

6. Other remarks

In this section, I provide several smaller remarks pertaining to the contents and editing of the dissertation.

Contents remarks

Page 10, bullet point 1: "AST" and "XMI" necessitate some explanation and references.
Page 11, reference to [12]: this is not a primary reference about anti-patterns; why was it chosen?
Page 11, reference to [100]: this also seems not a primary reference on technical debt.
Page 11 and 12, Table 2: the table claims to present "well-known code smells" but there is no explanation what is the source of this knowledge; what are the references that describe these code smells and what was the method for selecting them?
Page 13, references to [62]: this reference indeed contains an overview of code smell detection techniques; however, a question arises why is it used as a main (and only) reference for metric-based approaches.
Page 17, Notation: why doesn't this chapter contain all the abbreviations used in the dissertation? It would be much more convenient for the reader to have all the notation explanations in one place and not scattered throughout the monograph (Sections 8.1.2, 9.1.2, etc.).
Page 19, Table 2: the 'classes' and 'kLOC' columns should be clearly labelled as ranges (from-to); in the paragraph below the table, it is mentioned that the most recent version (BTW: the most recent version in 2021 is 3.8.1 and not 3.2.3) of Maven has 92 kLOC while the table mentions the range from 53 to 57 kLOC.
Page 20, Section 8.2: the purpose of this section (4 lines long) is not clear; it seems to be inserted by mistake.
Page 21, 'Shapiro-Wilk': no reference (cf. reference for Wilcoxon test on page 23).
Page 21, 'QQ-plots': no reference; what is the purpose of creating these plots? why not perform the Shapiro-Wilk test directly?
Page 23, `null hypothesis about normality': the hypothesis is nowhere defined, but it is rejected (and can be mistaken with H0 by a casual reader).
Page 30, "random sample of classes": how many classes were inspected? what was the procedure?
Page 33, "Weka": what are the 'very small values' applied to this algorithm?
Page 35, "relate the findings to other studies": what are these studies? the only reference is made to the study in [52].
Pages 41 and 48: what is the reason for this division into Sections 9.2 and 9.3?
Page 42, "exposed/unexposed cases/non-cases": these formulations should be clearly explained, as the reader might not be familiar with the OR test; they should also be mapped onto the actual variables that were used in the test; what are the examples of, e.g. unexposed cases?
Page 42, Tables 18 and 19: there is no discussion similar to that made for tables 16 and 17 previously on this page
Page 43, Table 16: no explanation of the variables; where is log(OR) used?
Page 45, "not normally distributed": for completeness of discussion, it should be explained how the tables show this.
Page 59, "CHURN" formula: what does "$f_{release(f)}$" mean?
Page 59, last paragraph: how does the manual validation look like? what were the sample classes, what was the statistical analysis? where are the results?
Page 61, "Kruskal-Wallis": there is no reference for this test; why this particular test was used? no explanation on the test was provided.
Page 62, "< 0.00": the variable is less than zero?
Page 64, Table 38: it should be clearly explained what is the variable that median and mean refer to.

Page 70, "ANOVA": what is this method about? no reference is given.
Page 73 (and others): example of the discussed classes should be given, either in text or as an appendix.

Editorial remarks

Page 19: Sec 7 → Section 7
Page 19, Table 3: protrudes onto the margin
Page 21: SDP! = SnDPp → SDP ≠ SnDPp
Pages 21-22, Tables 5 and 4: incorrect numbering (reverse)
Page 25, Table 6: protrudes onto the margin
Page 30, footnotes: placed on a wrong page
Page 44, Table 19: Defects → DEF
Page 45, "Table 34": erroneous reference, should be "Table 21"
Page 47, Table 47: the table fits within the margins, but the text is very small
Page 49, Table 27; Page 50, Table 28; Page 51, Table 29; Page 62, Table 36; Page 64, Table 39; Page 65, Table 41; Page 67, Table 43; Page 68, Table 46: tables protrude onto the margin
Page 72: several broken references, text protruding onto the margin
Page 79-88: numerous flaws in references – lacking page numbers, journal names, conference names, publishers, etc.; very inconsistent format of references.

7. Conclusion

Taking into account what I have presented above and the requirements imposed by Article 13 of *the Act of 14 March 2003 of the Polish Parliament on the Academic Degrees and the Academic Title* (with amendments), my evaluation of the dissertation according to the three basic criteria is the following:

A. Does the dissertation present an original solution to a scientific problem? (the selected option is marked with **X**)

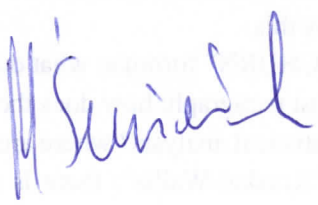
[X]	[ ]	[ ]	[ ]	[ ]
Definitely YES	*Rather yes*	*Hard to say*	*Rather no*	*Definitely NO*

B. After reading the dissertation, would you agree that the candidate has general theoretical knowledge and understanding of the discipline of **Information and Communication Technology**, and particularly the area of software engineering?

[ ]	[X]	[ ]	[ ]	[ ]
Definitely YES	*Rather yes*	*Hard to say*	*Rather no*	*Definitely NO*

C. Does the dissertation support the claim that the candidate is able to conduct scientific work?

[ ]	[X]	[ ]	[ ]	[ ]
Definitely YES	*Rather yes*	*Hard to say*	*Rather no*	*Definitely NO*



Signature