

**FRAMEWORK OF INFORMAL
SPECIFICATION VALIDATION WITH VISUAL
SPECIFICATION AND HUMAN FACTOR
INVOLVEMENT TO IMPROVE
REQUIREMENT SPECIFICATION QUALITY**

FAUZIAH BINTI ZAINUDDIN

DOCTOR OF PHILOSOPHY

UNIVERSITI MALAYSIA PAHANG



SUPERVISOR'S DECLARATION

We hereby declare that we have checked this thesis and in our opinion, this thesis is adequate in terms of scope and quality for the award of the degree of Doctor of Philosophy.

(Supervisor's Signature)

Full Name : DR. RUZAINI BIN ABDULLAH ARSHAH

Position : ASSOCIATE PROFESSOR

Date : 17 MAY 2019

(Co-supervisor's Signature)

Full Name : DR. ROZLINA BIN MOHAMAD

Position : SENIOR LECTURER

Date : 17 MAY 2019



STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

(Student's Signature)

Full Name : FAUZIAH BINTI ZAINUDDIN

ID Number : PCS16002

Date : 17 MAY 2019

**FRAMEWORK OF INFORMAL SPECIFICATION VALIDATION WITH VISUAL
SPECIFICATION AND HUMAN FACTOR INVOLVEMENT TO IMPROVE
REQUIREMENT SPECIFICATION QUALITY**

FAUZIAH BINTI ZAINUDDIN

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Doctor of Philosophy

Faculty of Computer Systems & Software Engineering
UNIVERSITI MALAYSIA PAHANG

MAY 2019

ACKNOWLEDGEMENTS

Alhamdulillah. All praises to Allah who has blessed me with the mercy and strength in completing this thesis. This thesis would not have been possible to complete without the assistance of many people who gave their support in different ways. I would like to thank my supervisors, Assoc. Prof. Dr. Ruzaini bin Abdullah Arshah and Dr. Rozlina bin Mohamad, for their suggestions, continued encouragement, and patience to guide me through my research. I have learned a lot from them.

I would like to thank all my colleagues from the Faculty of Computer Systems & Software Engineering, Universiti Malaysia Pahang, especially to the management of the Faculty for giving me the opportunity to further my study and for providing space and time in supporting me to complete the study.

I would like to express my deepest gratitude to my husband and twin girls, and especially to my parents who encouraged me to further my study for the sake of knowledge. I have always needed to work hard to achieve my goals in life and they have always been there for me with unwavering support. I dedicate this work to my parents and my husband, to honor their love, prayer, and support during those years.

ABSTRAK

Mutakhir ini, Kejuruteraan Keperluan telah mencetuskan minat yang mendalam dikalangan penyelidik. Penyelidikan dalam kejuruteraan perisian telah mengenal pasti bahawa kegagalan dan kelemahan sistem perisian biasanya berpunca daripada kelemahan keperluan spesifikasi yang dijalankan. Keadaan ini mengakibatkan impak yang serius ke atas kualiti spesifikasi tidak formal seperti ketidaksiapan, kesilapan, ketidakjelasan, kesukaran untuk mengesah dan ketidakfahaman. Secara praktisnya, spesifikasi tidak formal menjadi keutamaan pengamal untuk melakukan dokumentasi keperluan sistem berbanding spesifikasi semi atau formal. Sebaliknya, senario penyelidikan semasa menunjukkan lebih banyak kajian dilakukan ke atas pengesahan spesifikasi semi atau formal. Di mana teknik visualisasi berjaya diadaptasi untuk membantu faktor manusia bagi tujuan pengesahan kedua-dua jenis spesifikasi tersebut. Jelas, pengesahan spesifikasi tidak formal menggunakan teknik yang sama masih belum diterokai dan mampu memberi impak yang sama ke atas kualiti spesifikasi tidak formal. Untuk penukaran teks kepada visual, rujukan yang sistematik diperlukan bagi menentukan “apa” yang hendak divisualkan dan “bagaimana” untuk mengvisualkannya. Justeru, kajian mencadangkan satu kerangka pengesahan spesifikasi tidak formal yang merangkumi medium komunikasi (visualisasi) dan penglibatan secara langsung faktor manusia dalam memenuhi kualiti spesifikasi yang dimahukan. Kajian ini dimulakan dengan mengenal pasti bagaimana faktor manusia mampu mempengaruhi keperluan proses kejuruteraan keperluan, tahap bahasa spesifikasi yang telah disahkan oleh para pengamal, pendekatan Ulasan Kesusasteraan Bersistematis digunakan untuk meneroka faktor-faktor tersebut. Berdasarkan kepada penemuan kajian, satu kerangka kerja konseptual telah dibangunkan oleh penyelidik. Prosedur transformasi dalam kerangka kerja tersebut diformulasi untuk membantu transisi daripada teks kepada visual. Diakhir kajian, kerangka kerja yang dibangunkan diuji kesahihannya menggunakan kaedah kombinasi formulasi statistik (penerangan secara berturutan) iaitu pengesahan melibatkan kajian kes dan pandangan pakar. Kerangka kerja Kejuruteraan Keperluan Tiga Dimensi dan Taksonomi Perisian Visualisasi menjadi asas kepada Kerangka Kerja Spesifikasi Keperluan Dan Pengesahan Faktor Manusia (REsHFv). Ini membolehkan spesifikasi tidak formal boleh ditentusahkan sebagai lengkap, betul, jelas serta boleh difahami. Penilaian pengesahan mendapati, kerangka yang dibina boleh diaplikasikan dan menghasilkan spesifikasi tidak formal yang berkualiti untuk sesebuah projek, peningkatan peratusan untuk setiap kualiti spesifikasi yang dikaji adalah di antara 55% hingga 30%. Ujian t-Test (daripada kajian kes) dilakukan dengan dua sampel (Perisian Keperluan Spesifikasi – markah penuh SRS) untuk mendapatkan nilai-p ujian yang kurang daripada 0.05 ($p < 0.05$). Ini menunjukkan keputusan yang signifikan untuk menyimpulkan bahawa kualiti Spesifikasi Keperluan Perisian, pembina SRS dengan Spesifikasi Visualisasi Perisian mampu menyediakan kualiti SRS lebih baik pada tahap signifikan sebanyak 5%. Hasil pandangan panel pakar, turut menyokong keputusan dan mengesahkan kerangka kerja yang dibina (REsHFv) yang boleh diaplikasikan dalam kejuruteraan perisian sebenar. Kajian ini menyumbang kepada pembangunan satu kerangka kerja untuk spesifikasi tidak formal yang berkualiti dengan mengambil kira faktor manusia dan perisian spesifikasi visual bagi tujuan pengesahan spesifikasi tidak formal.

ABSTRACT

Requirement engineering has attracted a great deal of attention from researchers and practitioners in recent years. Researchers endeavor in software engineering had identified that failure and deficiency of software system often rooted to requirement specification undertaken. Previous studies indicate that there are issues in human factor involvement especially in validating informal specification. Such phenomena would later cause serious impacts to quality of informal specification such as incompleteness, incorrectness, ambiguity, difficult to verify and incomprehensible. In practice, informal specification is preferred by practitioners to perform documentation for system requirement compared to semi- or formal specification. In contrast, current research scenario shown more studies were conducted for semi- or formal specification validation; whereby, visualization technique was successfully adopted in assisting human factor for validation purposes. Apparently, informal specification validation using the same technique is not yet explored; and obviously visualization is able to give similar impact to informal specification quality as it does for semi- and formal specification validation. Consequently, to transform text into visual, the process required systematic assistance to specify “what” to be visualized and “how” to visualize them. Therefore, this study addresses these issues by proposing a framework of informal specification validation that includes communication medium (visualization) and direct human factor involvement in satisfying the intended specification quality attributes (with respect to human factor perspective). The study starts with identifying the nature of how human factor influences the requirement engineering process, levels of specification language validated by practitioners, current mechanisms in assisting human factor, recent specification language in validation perspective and identifying quality attributes for requirement specification that are significant to human factor involvement using *Systematic Literature Review* approach. Based on the findings, a conceptual framework was developed. Transformation procedure in the proposed framework was then formulated to assist the transformation from text to visual. Finally, this study adopts mix-methods (explanatory sequential) for framework validation purposes that include case studies and expert review. Putting *The Three Dimensions of Requirements Engineering* framework and the *Software Visualization Taxonomy* as the basis, the *Requirement Specification with Human Factor Validation* (REsHFv) framework was proposed to deal with informal specification validation by human factor involvement using visualization as the medium in order to achieve complete, correct, unambiguous, comprehensible and verifiable informal specification. From the validation assessment, the framework was proven to be applicable and was able to produce good quality of informal specification for a project, the percentage of the quality improvement is between 55% to 30% for each quality attributes. The t-Test (from the case study) for paired two-sample (Software Requirement Specification - SRS total marks) for mean *p*-value for this test is less than 0.05 (*p* < 0.05). Hence, this implies that there is a significant evidence to conclude that the SRS quality constructed with Software Visual Specification (SVS) is able to provide improved quality of SRS at 5% significance level. Meanwhile, experts reviewed that the results indicate the proposed framework is applicable and relevant in real software engineering setting. This research contributes to the development framework for quality informal specification by incorporating human factor and software visual specification for informal specification validation purposes. Thus, the proposed REsHFv framework is expected to provide beneficial impact to the quality of informal specification resulted from direct human factor involvement during specification validation.

TABLE OF CONTENT

DECLARATION

TITLE PAGE

ACKNOWLEDGEMENTS	ii
-------------------------	----

ABSTRAK	iii
----------------	-----

ABSTRACT	iv
-----------------	----

TABLE OF CONTENT	v
-------------------------	---

LIST OF TABLES	ix
-----------------------	----

LIST OF FIGURES	xi
------------------------	----

LIST OF ABBREVIATIONS	xiv
------------------------------	-----

CHAPTER 1 INTRODUCTION	1
-------------------------------	---

1.1 Introduction	1
------------------	---

1.2 Research Background	3
-------------------------	---

1.3 Problem Statements	4
------------------------	---

1.4 Research Questions	7
------------------------	---

1.5 Research Objectives	7
-------------------------	---

1.6 Research Scope	9
--------------------	---

1.7 Significant of the Study	9
------------------------------	---

1.8 Organization of the Thesis	10
--------------------------------	----

CHAPTER 2 LITERATURE REVIEW	11
------------------------------------	----

2.1 Introduction	11
------------------	----

2.2 Requirement Engineering Process	12
-------------------------------------	----

2.3 Requirement Specification	14
-------------------------------	----

2.3.1	Specification Language	15
2.3.2	Requirement Specification Validation	17
2.3.3	Techniques for Validating Requirement Specification	22
2.4	Human Factor Involvement in Requirement Engineering	26
2.4.1	Existing Theories on Human Factor Involvement in Requirement Validation	27
2.4.2	Human Factor Involvement in Requirement Specification Validation	29
2.5	Quality Attributes for Requirement Specification (with respect to Human Involvement)	33
2.6	Software Visualization Taxonomy	35
2.6.1	Use of Software Visualization in SE and RE	39
2.7	Existing Software Engineering Research Framework Validation	40
2.8	Systematic Literature Review (SLR) - Summary	42
2.9	Summary	43
CHAPTER 3 METHODOLOGY		47
3.1	Introduction	47
3.2	Research Methodology	47
3.3	Research Operational Framework	50
3.3.1	Phase I – Problem Identification	50
3.3.2	Phase II – Establishment of the Integration Framework	56
3.3.3	Phase III – Develop Transformation Procedure	60
3.3.4	Phase IV – Framework Validation	61
3.4	Summary	66
CHAPTER 4 REQUIREMENT SPECIFICATION HUMAN FACTOR (VALIDATION) – RESHFV FRAMEWORK		67

4.1	Introduction	67
4.2	Informal Specification	71
4.3	Transformation Procedure	72
	4.3.1 Software Visualization Taxonomy – Extended version (SVT-Ev)	73
	4.3.2 Software Visual Specification - structure (SVS-s)	78
	4.3.3 Visualization Tool	83
4.4	Informal Specification Validation (ISV)	84
4.5	Software Visual Specification	87
4.6	Validated Informal Specification	87
4.7	Summary	88
CHAPTER 5 FRAMEWORK VALIDATION		89
5.1	Introduction	89
5.2	Framework Validation – Target Components	89
5.3	Framework Validation Instrument – Design and Development	91
	5.3.1 Experiment Set Up - Case Study	93
	5.3.2 Experiment Set Up - Expert Review	96
5.4	Result from The Case Study	97
	5.4.1 SVS Sample from Case Study	98
	5.4.2 Results from Stakeholders	100
	5.4.3 Result from System Analyst	104
	5.4.4 Result Analysis for Case Study	107
5.5	Expert Review Report and Feedback	120
	5.5.1 Expert Review Section A	122
	5.5.2 Expert Review Section B	128
	5.5.3 Result Analysis for Expert Review	131

5.6 Summary	132
CHAPTER 6 CONCLUSION	134
6.1 Introduction	134
6.2 Synthesis	134
6.3 Discussion on the Findings	136
6.4 Research Contributions	136
6.4.1 Contribution to Theory	136
6.4.2 Contribution to Practice	138
6.5 Future Works	138
REFERENCES	140
APPENDIX A THE SOFTWARE VISUALIZATION TAXONOMY CATEGORIES	157
APPENDIX B CASE STUDY INSTRUMENT SET UP	160
APPENDIX C CASE STUDY - QUESTIONNAIRE	162
APPENDIX D EXPERT REVIEW INSTRUMENT SET UP	176
APPENDIX E EXPERT PANEL QUESTIONNAIR	178
APPENDIX F REPORT ON STUDY SELECTION OF SYSTEMATIC LITERATURE REVIEW (SLR) FOR THE CHALLENGE AND PRACTICE OF HUMAN FACTOR INVOLVEMENT IN REQUIREMENT SPECIFICATION VALIDATION	186
APPENDIX G SVS ADD-IN TOOL FRAMEWORK	189

LIST OF TABLES

Table 2.1	Current works on requirement specification validation with direct human factor involvement	20
Table 2.2	Detail distribution for requirement specification using cognitive analysis	22
Table 2.3	Summary of specification validation techniques	23
Table 2.4	Papers distribution details	30
Table 2.5	Human factor involvement details	30
Table 2.6	Requirement specification quality attributes	33
Table 2.7	Distribution of quality attributes found in the papers	35
Table 2.8	Visualization definitions	35
Table 2.9	List of relevant studies on requirement specification from SLR	44
Table 3.1	Search source	53
Table 3.2	Inclusion and exclusion criteria	55
Table 4.1	List of newly identified visual vocabularies and presentation criterion	75
Table 4.2	Newly identified visual vocabularies and presentation criteria vs source/s details	76
Table 4.3	SVS-s execution procedure algorithm	82
Table 4.4	Algorithm ISV review process with SVS	85
Table 5.1	Internal consistency reliability - Cronbach Alpha	92
Table 5.2	Iteration 1: list of questionnaire items with mean scores for group A and B	101
Table 5.3	Iteration 2: list of questionnaire items with mean scores for group A and B	103
Table 5.4	Section B: list of questionnaire items with mean scores for group A and B	106
Table 5.5	Section C: list of questionnaire items with mean scores for group A and B	106
Table 5.6	Section D: list of questionnaire items with mean scores for group A and B	107
Table 5.7	Marks comparison (SRS content and specification quality)	111
Table 5.8	t-Test: Two-Sample assuming unequal variances	112
Table 5.9	Comparing Mean Score Value for each SRS Quality attributes (Iteration 2 versus Iteration 1)	118
Table 5.10	Mean score for items related to SRS quality attributes for iteration 1 and 2	119

Table 5.11	t-Test: paired two sample for means	119
Table 5.12	Item's content validity index from expert panel for REsHFv framework (Part I: REsHFv Framework)	123
Table 5.13	Item content validity index from expert panel for SVT-Ev (PART II: Software Visualization Taxonomy -Extended Version (SVT-Ev))	125
Table 5.14	Item content validity index from expert panels (PART III: SVS-structure (SVS-s) diagram)	127
Table 5.15	Item content validity index from expert panel for SVS add-in tool in Microsoft PowerPoint (PART IV: SVS add-ins Tool in Microsoft PowerPoint)	128
Table 5.16	Item content validity index from expert panels for REsHFv framework execution (PART V: REsHFv Framework Execution)	129
Table 5.17	Case study comparison (PART VI: CASE STUDY A and CASE STUDY B comparison)	131
Table A- 6.1	Software Visualization Taxonomy Characteristics Source Mapping	158
Table C- 6.2	Sections Of System Analyst Questionnaire And Corresponding Research Questions And Dependent Variables	162
Table C- 6.3	Sections Of Stakeholders Questionnaire And Corresponding Research Questions And Dependent Variables	166
Table E-6.4	Expert review questionnaire and corresponding research questions and dependent variables	178
Table F-6.5	Summary of Included and Excluded data	186
Table F-6.6	Quality Criteria for Study Selection	187
Table F-6.7	Summary Of The Most Relevant Studies	187
Table G-6.8	User Interface Basic Components	192
Table G-6.9	Single Animation Templates	193
Table G-6.10	Composite Animation Templates	193
Table G-6.11	Domain-Based Animation Templates	194
Table G-6.12	System Skin	194

LIST OF FIGURES

Figure 1.1	Research question mapping to research objective	8
Figure 2.1	Literature review framework	11
Figure 2.2	Requirement engineering process	12
Figure 2.3	Specification language analysis	17
Figure 2.4	Validation techniques and detail distribution	22
Figure 2.5	Human factor involvement vs specification presentation style	23
Figure 2.6	Human factor involvement in requirement specification	29
Figure 2.7	Research on requirement specification addressing human factor involvement	32
Figure 2.8	Software visualization taxonomy formation	37
Figure 2.9	Software visualization taxonomy main categories	38
Figure 3.1	Research methodology	49
Figure 3.2	Mapping for research question and objective for its research contribution	50
Figure 3.3	Review protocol	51
Figure 3.4	The Three Dimensions of Requirement Engineering (Pohl, 1993)	58
Figure 3.5	Research Aim in The Three Dimensions of Requirement Engineering	58
Figure 3.6	Work flow for integration framework construction	59
Figure 3.7	Explanatory sequential for mixed methods approach	61
Figure 4.1	Fundamental framework for REsHFv	67
Figure 4.2	Initial requirement engineering (specification) human factor (validation) – REsHFv framework	68
Figure 4.3	Construct REsHFv from the Integration Framework Construction Workflow	70
Figure 4.4	REsHFv framework	71
Figure 4.5	Software requirement specification standard structure by IEEE Std 830-1998	72
Figure 4.6	Software visualization taxonomy –extended version (SVT-Ev)	77
Figure 4.7	Mapping of software visual specification - structure (SVS-s) attributes from SVT-Ev characteristics	79
Figure 4.8	Software visual specification - structure (SVS-s)	81
Figure 4.9	Derivation of characteristics in SVT-Ev into SVS ribbon tab	83
Figure 4.10	SVS-structure ribbon tab	83

Figure 4.11	Informal specification validation (ISV) review process (with SVS assistance)	86
Figure 5.1	Framework's components to be validated	90
Figure 5.2	Framework evaluation phase	92
Figure 5.3	Case study execution process	95
Figure 5.4	Expert review implementation procedure	97
Figure 5.5	SVS sample from group A	98
Figure 5.6	SVS sample from group B	99
Figure 5.7	Iteration 1 - SRS review by stakeholder	102
Figure 5.8	Case study respondents' information	105
Figure 5.9	Dialogue diagram usefulness	108
Figure 5.10	SVS-s usefulness	109
Figure 5.11	SVS construction tool criteria	110
Figure 5.12	SVS usefulness in SRS construction	113
Figure 5.13	SVS usefulness in SRS validation	114
Figure 5.14	Stakeholder involvement in SRS validation–part 1	115
Figure 5.15	Stakeholder involvement in SRS validation–part 2	116
Figure 5.16	Stakeholder Involvement in SRS Validation–Part 3	117
Figure 5.17	SRS quality mean score comparison for iteration 1 and 2	118
Figure 5.18	Expert Demographic Information	121
Figure 5.19	Description level rating distribution for REsHFvfFramework components	124
Figure 5.20	Description level rating distribution for SVT-Ev	126
Figure 5.21	Summary of means I-CVI from expert panels	132
Figure A 6.1	Software Visualization Taxonomy (form category) by (Price et al., 1993)	157
Figure C- 6.2	Questionnaire Iteration 1 – FORMAL REVIEW by Stakeholder	172
Figure C- 6.3	Questionnaire Iteration 2 – FORMAL REVIEW by Stakeholder	173
Figure C- 6.4	Questionnaire for System Analyst (page 1of 2)	174
Figure C- 6.5	Questionnaire for System Analyst (page 2 of 2)	175
Figure E-6.6	Expert Panel Information Details	181
Figure E-6.7	Section A – Part I	182
Figure E-6.8	Section A – Part II	182
Figure E-6.9	Section A – Part III	183
Figure E-6.10	Section A – Part IV	183

Figure E-6.11 Section B (Page 1 of 2) – Part V	184
Figure E-6.12 Section B (Page 2 of 2) – Part V	185
Figure G-6.13 SVS add-in system components	189
Figure G-6.14 GUI for SVS add-in tool in Microsoft PowerPoint	191
Figure G-6.15 Derivation of characteristics in SVT-Ev into SVS Ribbon Tab	192
Figure G-6.16 SVS-structure Ribbon Tab	195

REFERENCES

- Abad, Z. S. H., Ruhe, G., & Noaeen, M. (2016). Requirements engineering visualization: A systematic literature review. *Proceedings of the IEEE 24th International Requirement Engineering Conference (RE)*, 6–15.
- Abelein, U., & Paech, B. (2015). Understanding the influence of user participation and involvement on system success – a systematic mapping study. *Empirical Software Engineering*, 20(1), 28–81.
- Achimugu, P., Selamat, A., Ibrahim, R., & Naz, M. (2014). A systematic literature review of software requirements prioritization research. *Information and Software Technology*, 56(6), 568–585.
- Add-in Express .Net. (2010). *Official Guide – Add-in Express 2010 for Microsoft Office and .Net*.
- Agerholm, S., & Larsen, P. G. (1998). A lightweight approach to formal methods. *Proceedings of the International Workshop on Current Trends in Applied Formal Methods*, 168–183.
- Alhogail, A. (2015). Design and validation of information security culture framework. *Computers in Human Behavior*, 49(2015), 567–575.
- Alzain, A. M., Clark, S., & Jwaid, A. (2016). A study of the reliability and validity of the first Arabic learning styles instrument (ALSI). *Proceedings of the World Congress Sustainable Technologies (WCST)*, 29–34.
- Amálio, N., & Glodt, C. (2014). A tool for visual and formal modelling of software designs. *Science of Computer Programming*, 98, 52–79.
- Ambreen, T., Ikram, N., Usman, M., & Niazi, M. (2016). Empirical research in requirements engineering: trends and opportunities. *Requirements Engineering*, 23(1), 63–95.
- Apple Computer, I. (1992). *Macintosh human interface guidelines*. Addison-Wesley Publishing Company, USA.
- Apple Computer, I. (2017). IOS Human interface guidelines (online). Retrieved from <https://developer.apple.com/design/> on September 13, 2017.

- Assasi, N., Tarride, J.-E., O'Reilly, D., & Schwartz, L. (2016). Steps toward improving ethical evaluation in health technology assessment: a proposed framework. *BMC Medical Ethics*, 17(1), 34.
- Bahill, A. T., & Henderson, S. J. (2005). Requirements development, verification, and validation exhibited in famous failures. *Systems Engineering*, 8(1), 1–14.
- Ball, T., & Eick, S. G. (1996). Software visualization in the large. *Computer*, 29(4), 33–43.
- Bano, M., & Ikram, N. (2014). Addressing the challenges of alignment of requirements and services: A vision for user-centered method. *Requirements Engineering*, 83–89.
- Bano, M., & Zowghi, D. (2013). Users' involvement in requirements engineering and system success. *Proceedings of the 3rd International Workshop on Empirical Requirements Engineering (EmpiRE)*, 23–31.
- Batool, A., Motla, Y. H., Hamid, B., Asghar, S., Riaz, M., Mukhtar, M., & Ahmed, M. (2013). Comparative study of traditional requirement engineering and agile requirement engineering. *Proceedings of the 15th International Conference on Advanced Communications Technology (ICACT)*, 1006–1014.
- Beecham, S., Hall, T., Britton, C., Cottee, M., & Rainer, A. (2005). Using an expert panel to validate a requirements process improvement model. *Journal of Systems and Software*, 76(3), 251–275.
- Belfo, F. (2012). People, organizational and technological dimensions of software requirements specification. *Procedia Technology*, 5, 310–318.
- Belhajjame, K., Paton, N. W., Fernandes, A. A. A., Hedeler, C., & Embury, S. M. (2011). User feedback as a first class citizen in information integration systems. *Proceedings of the 5th Biennial Conference on Innovative Data Systems Research CIDR*, 175–183.
- Berander, P. (2004). Using students as subjects in requirements prioritization. *Proceedings of the International Symposium on Empirical Software Engineering ISESE '04*, 167–176. IEEE.
- Berenbach, B., Paulish, D. J., Kazmeier, J., & Rudorfer, A. (2009). *Software & systems requirements engineering: in practice*. The McGraw-Hill Companies.

- Bergin, J., Brodlie, K., Patiño-Martínez, M., McNally, M., Naps, T., Rodger, S., ... Khuri, S. (1996). An overview of visualization: its use and design: report of the working group in visualization. *ACM SIGCSE Bulletin*, 24, 192–200.
- Björndal, P., Rissanen, M. J., & Murphy, S. (2011). Lessons learned from using Personas and scenarios for requirements specification of next-generation industrial robots. *Proceedings of the International Conference of Design, User Experience and Usability*, 378–387.
- Blaine Price, R. B. and I. S. (1998). An introduction to software visualization. *Software Visualization*, 3–27.
- Bowen, J., & Reeves, S. (2007). Formal models for informal GUI designs. *Electronic Notes in Theoretical Computer Science*, 183, 57–72.
- Brath, R., & Banissi, E. (2016). Evaluation of visualization by critiques. *Proceedings of the 6th Workshop on Beyond Time and Errors on Novel Evaluation Methods for Visualization*, 19–26.
- Bures, T., Hnetyrnka, P., Kroha, P., & Simko., V. (2012). Requirement specifications using natural languages. (Technical Report). D3S-TR-2012-05.
- Card, S. K., Mackinlay, J., & Shneiderman, B. (1999). Reading in information visualization: using vision to Think. Morgan Kaufmann (Online). Retrieved from papers2://publication/uuid/A8A1FFDB-DA15-4926-9FE6-FF51757C7B0A
- Chang, S.-K. (1990). *Principles of visual programming systems*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA
- Chen, D., Chen, W., & Kavi, K. M. (2002). Visual requirement representation. *Journal of Systems and Software*, 61(2), 129–143.
- Cheng, B. H. C., Atlee, J. M., & Joanne, M. (2007). Research directions in Requirements Engineering. *Future of Software Engineering*, 285–303.
- Cimatti, A., Roveri, M., Susi, A., & Tonetta, S. (2012). Validation of requirements for hybrid systems: A formal approach. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 21(4), 22.
- Clancy, T. (2014). *The Standish group chaos report*. Project Smart.

- Cooling, J. E., & Hughes, T. S. (1994). Making formal specifications accessible through the use of animation prototyping. *Microprocessors and Microsystems*, 18(7), 385–392.
- Cooper, J. R., Lee, S. W., Gandhi, R. A., & Gotel, O. (2009). Requirements engineering visualization: A survey on the state-of-the-art. *Proceedings of the 4th International Workshop on Requirements Engineering Visualization*, 46–55.
- Crear, J. (2009). *Chaos summary 2009*. Boston MA: Standish Group.
- Cusumano, M., MacCormack, A., Kemerer, C. F., & Crandall, B. (2003). Software development worldwide: The state of the practice. *IEEE Software*, 20(6), 28–34.
- Cypher, A. (1993). *Watch what I do: programming by demonstration*. MIT Press.
- Davis, L. L. (1992). Instrument review: Getting the most from a panel of experts. *Applied Nursing Research*, 5(4), 194–197.
- Dawes, J. (2008). Do data characteristics change according to the number of scale points used? An experiment using 5-point, 7-point and 10-point scales. *International Journal of Market Research*, 50(1), 61–104.
- De Oliveira, K. M., Bacha, F., Mnasser, H., & Abed, M. (2013). Transportation ontology definition and application for the content personalization of user interfaces. *Expert Systems with Applications*, 40(8), 3145–3159.
- Denger, Christian, and T. O. (2005). Quality assurance in requirements engineering. *Engineering and Managing Software Requirements*, 163–185.
- Developer, G. (n.d.) (2017). GNOME Human Interface Guidelines (Online). Retrieved from <https://developer.gnome.org/hig/stable/> on September 13, 2017.
- Dix, A. (2009). *Human-computer Interaction*. Springer US.
- Duke, D. J., Brodlie, K. W., & Duce, D. A. (2004). Building an ontology of visualization. *Proceedings of the Conference on Visualization '04*, 598–7.
- Ebrahiminejad, S., Tehrani, M., Megat, N., Zainuddin, M., & Takavar, T. (2014). Heuristic evaluation for Virtual Museum on smartphone. *Proceedings of the 3rd International Conference on User Science and Engineering (i-USER)*, 227–231.

- Ermel, C., Holscher, K., Kuske, S., & Ziemann, P. (2005). Animated simulation of integrated UML behavioral models based on graph transformation. *Proceedings of the IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC'05)*, 125–133.
- Ernst, N. A., Yu, Y., & Mylopoulos, J. (2006). Visualizing non-functional requirements. *Proceedings of the 1st International Workshop on Requirements Engineering Visualization (REV'06-RE'06 Workshop)*, 2–2.
- Fabbrini, F., Fusani, M., Gnesi, S., & Lami, G. (2001). An automatic quality evaluation for natural language requirements. *Proceedings of the 7th International Workshop on Requirements Engineering: Foundation for Software Quality REFSQ*, 4–5.
- Feather, M. S., Cornford, S. L., Kiper, J. D., & Menzies, T. (2006). Experiences using visualization techniques to present requirements, risks to them, and options for risk mitigation. *Proceedings of the 1st International Workshop on Requirements Engineering Visualization (REV'06-RE'06 Workshop)*, 10–10.
- Feja, S., Speck, A., & Kiel, C. (2011). BAM: a requirements validation and verification framework for business process models. *Proceedings of the 11th International Conference on Quality Software*, 186–191.
- Field, A. (2009). *Discovering statistics using SPSS*. Sage publications.
- Fricker, S. A., Schneider, K., Fotrousi, F., & Thuemmler, C. (2016). Workshop videos for requirements communication. *Requirements Engineering*, 21(4), 521–552.
- Gallagher, K., Hatch, A., Munro, M., Society, I. C., Hatch, A., & Munro, M. (2008). Software architecture visualization: An evaluation framework and its application. *IEEE Transactions on Software Engineering*, 34(2), 260–270.
- Garrett, J. J. (2010). *Elements of user experience, the: user-centered design for the web and beyond*. Pearson Education.
- Gemino, A. (2004). Empirical comparisons of animation and narration in requirements validation. *Requirements Engineering*, 9(3), 153–168.
- Gemino, A., & Wand, Y. (2003). Evaluating Modeling Techniques Based on Models of Learning. *Communications of the ACM*, 46(10), 79.

Génova, G., Fuentes, J. M., Llorens, J., Hurtado, O., & Moreno, V. (2013). A framework to measure and improve the quality of textual requirements. *Requirements Engineering*, 18(1), 25–41.

Ghazel, M., & Yang, J. (2015). A pattern-based method for refining and formalizing informal specifications in critical control systems. *Journal of Innovation in Digital Ecosystems*, 2(1–2), 32–44.

Giese, M., & Heldal, R. (2004). From informal to formal specifications in UML. *Proceedings of the International Conference on the Unified Modeling Language*, 197–211.

Glinert, E. P. (1990a). *Visual programming environments: applications and issues*. IEEE Computer Society Press.

Glinert, E. P. (1990b). *Visual programming environments: paradigms and systems*. IEEE Computer Society Press.

Glinz, M. (2007). On Non-Functional Requirements. *Proceedings of the 15th IEEE International Requirements Engineering Conference (RE 2007)*, 21–26.

Grant, C., & Osanloo, A. (2014). Understanding, selecting, and integrating a theoretical framework in dissertation research: Creating the blueprint for your “house.” *Administrative Issues Journal*, 4(2), 4.

Guerra, E., Lara, J. De, & Di, P. (2008). Visual specification of measurements and redesigns for domain specific visual languages. *Journal of Visual Languages and Computing*, 19(3), 399–425., 19, 399–425.

Hall, A. (1990). Seven myths of formal methods. *IEEE Software*, 7(5), 11–19.

Hansen, S. W., Robinson, W. N., & Lyytinen, K. J. (2012). Computing requirements: Cognitive approaches to distributed requirements engineering. *Proceedings of the 45th Hawaii International Conference on System Sciences*, 5224–5233.

Hanson, W. E., Creswell, J. W., Clark, V. L. P., Petska, K. S., & Creswell, J. D. (2005). Mixed methods research designs in counselling psychology. *Journal of Counselling Psychology*, 52(2), 224.

Hassanzadeh, A., & Namdarian, L. (2011). Developing a framework for evaluating service oriented architecture governance (SOAG). *Knowledge-Based Systems*, 24(5), 716–730.

Hazel, D., & Traynor, O. (1998). Requirements engineering and verification using specification animation. *Proceedings of the 13th IEEE International Conference on Automated Software Engineering*, 302–305.

Hoffer, J. A. (2012). *Modern Systems Analysis and Design (6/e)*. Pearson Education India.

Hofmann, H. F., & Lehner, F. (2001). Requirements engineering as a success factor in software projects. *IEEE Software*, (4), 58–66.

Hong, Y., & Nam, T. (2010). A method to get rich feedbacks from users in an interview for design concept decision. *Proceedings of the CHI'10 Extended Abstracts on Human Factors in Computing Systems (CHI EA '10)*, 3907–3912.

Hsu, C., & Ohio, T. (2007). The Delphi technique: making sense of consensus. *Practical Assessment, Research and Evaluation*, 12(10), 1–8.

Hull, E., Jackson, K., & Dick, J. (2005). *Requirements Engineering*. Springer London.

Hyrkäs, K., Appelqvist-Schmidlechner, K., & Oksa, L. (2003). Validating an instrument for clinical supervision using an expert panel. *International Journal of Nursing Studies*, 40(6), 619–625.

Ibrahim, N., Kadir, W., Wan, M. N., & Deris, S. (2014). Documenting requirements specifications using natural language requirements boilerplates. *Proceedings of the 8th. Malaysian Software Engineering Conference (MySEC)*, 19–24.

Ibriwesh, I., Ho, S.-B., Chai, I., & Tan, C.-H. (2017). A controlled experiment on comparison of data perspectives for software requirements documentation. *Arabian Journal for Science and Engineering*, 42(8), 3175–3189.

Iee, E. (1990). *IEEE standard glossary of software engineering terminology*. ANSI/ IEEE Std 729-1983

Iee, E. (1998). *IEEE Recommended practice for software requirements specifications*. IEEE Std 830-1998.

Imenda, S. (2014). Is there a conceptual difference between theoretical and conceptual frameworks? *Journal of Social Sciences*, 38(2), 185–195.

Inayat, I., Salwah, S., Marczak, S., Daneva, M., & Shamshirband, S. (2015). A systematic literature review on agile requirements engineering practices and challenges. *Computers in Human Behaviour*, 51, 915–929.

International Organization for Standardization. (1998). *ISO 9241-11: Ergonomic requirements for office work with visual display terminals (VDTs) - part 11: guidance on usability*. Geneva Switzerland.

Ivankova, N. V., Creswell, J. W., & Stick, S. L. (2006). Using mixed-methods sequential explanatory design: From theory to practice. *Field Methods*, 18(1), 3–20.

Jelemenská, K., Čičák, P., & Dúcky, V. (2011). Interactive presentation towards students' engagement. *Procedia - Social and Behavioural Sciences*, 29, 1645–1653.

Johann, T., & Maalej, W. (2015). Democratic mass participation of users in requirements engineering? *Proceedings of the IEEE 23rd International Requirements Engineering Conference (RE)*, 256–261.

Johnson, R. B. B., Onwuegbuzie, A. J. A. J., & Turner, L. A. L. A. (2007). Toward a definition of mixed methods research. *Journal of Mixed Methods Research*, 1(2), 112–133.

Kabaale, E., & Kituyi, G. M. (2015). A theoretical framework for requirements engineering and process improvement in small and medium software companies. *Business Process Management Journal*, 21(1), 80–99.

Kahraman, G., & Bilgen, S. (2015). A framework for qualitative assessment of domain-specific languages. *Software and Systems Modelling*, 1505–1526.

Kamalrudin, M., & Grundy, J. (2011). Generating essential user interface prototypes to validate requirements. *Proceedings of the 26th IEEE/ACM International Conference on Automated Software Engineering (ASE 2011)*, 564–567.

Kamalrudin, M., & Sidek, S. (2015). A review on software requirements validation and consistency management. *International Journal of Software Engineering and Its Application*, 9(10), 39–58.

Kamimori, S., Ogata, S., & Kaijiri, K. (2015). Automatic method of generating a web prototype employing live interactive widget to validate functional usability requirements. *Proceedings of the 3rd International Conference on Applied Computing and Information Technology/2nd International Conference on Computational Science and Intelligence*, 8–13.

- Kamsties, E., & Peach, B. (2000). Taming ambiguity in natural language requirements. *Proceedings of the 13th International Conference on Software and Systems Engineering and Applications*.
- Karlsson, L., Dahlstedt, Å. G., Regnell, B., Natt och Dag, J., & Persson, A. (2007). Requirements engineering challenges in market-driven software development - An interview study with practitioners. *Information and Software Technology*, 49(6), 588–604.
- Kassab, M., Neill, C., & Laplante, P. (2014). State of practice in requirements engineering: contemporary data. *Innovations in Systems and Software Engineering*, 10(4), 235–241.
- Keele, S. (2007). Guidelines for performing systematic literature reviews in software engineering (Vol. 5). (Technical report). Ver. 2.3 EBSE Technical Report. EBSE.
- Kennard, R., & Leaney, J. (2010). Towards a general purpose architecture for UI generation. *Journal of Systems and Software*, 83(10), 1896–1906.
- Kennard, R., & Leaney, J. (2011). Is there convergence in the field of UI generation? *Journal of Systems and Software*, 84(12), 2079–2087.
- Khanom, S., Heimbürger, A., & Kärkkäinen, T. (2015). Can icons enhance requirements engineering work? *Journal of Visual Language and Computing*, 28, 147–162.
- Kilicay-Ergin, N., & Laplante, P. A. (2013). An online graduate requirements engineering course. *IEEE Transactions on Education*, 56(2), 208–216.
- Kitchenham, B. (1996). Evaluating software engineering methods and tool part 1: The evaluation context and evaluation methods. *ACM SIGSOFT Software Engineering Notes*, 21(1), 11–14.
- Kitchenham, B., Pearl Brereton, O., Budgen, D., Turner, M., Bailey, J., & Linkman, S. (2009). Systematic literature reviews in software engineering - A systematic literature review. *Information and Software Technology*, 51(1), 7–15.
- Kitchenham, B., & Pfleeger, S. L. (2008). Personal opinion survey. *Guide to Advance Empirical Software Engineering*, 63–92.
- Kitchenham, B., Pickard, L., & Lawrence, S. (1995). Case studies for method and tool evaluation. *IEEE Software*, 12(4), 52–62.

- Kosara, R. (2007). Visualization criticism - The missing link between information visualization and art. *Proceedings of the 11th International Conference Information Visualization (IV'07)*, 631–636.
- Kosara, R., & Mackinlay, J. (2013). Storytelling: The Next Step for Visualization. *IEEE Computer*, 46(5), 44–50.
- Kujala, S., Kauppinen, M., Lehtola, L., & Kojo, T. (2005). The role of user involvement in requirements quality and project success. *Proceedings of the 3rd IEEE International Conference on Requirements Engineering (RE'05)*, 75–84.
- Lamsweerde, A. V. (2000). Formal specification: a roadmap. *Proceedings of the Conference on the Future of Software Engineering*, 147–159.
- Lawshe, C. H. (1975). A quantitative approach to content validity 1. *Personnel Psychology*, (28), 563–575.
- Leonardi, C., Sabatucci, L., Susi, A., & Zancanaro, M. (2010). Ahab's leg: Exploring the issues of communicating semi-formal requirements to the final users. *Proceedings of the International Conference on Advanced Information Systems Engineering*, 455–469.
- Lethbridge, T. C., Singer, J., & Forward, A. (2003). How software engineers use documentation: The state of the practice. *IEEE Software*, 20(6), 35–39.
- Li, D., Li, X., Liu, J., & Liu, Z. (2008). Validation of requirement models by automatic prototyping. *Innovation System Software Engineering*, 4(3), 241–248.
- Lindner, S., Büttner, P., Taentzer, G., Vaupel, S., & Russwinkel, N. (2014). *Towards an efficient evaluation of the usability of Android apps by cognitive models*. Kognitive Systeme III, DuEPublico.
- Liu, S. (2007). Utilizing test case generation to inspect formal specifications for completeness and feasibility. *Proceedings of the 10th IEEE High Assurance Systems Engineering Symposium (HASE'07)*, 349–356.
- Łobaziewicz, M. (2015). The design of B2B system user interface for mobile systems. *Procedia Computer Science*, 65, 1124–1133.
- Lucassen, G., Robeer, M., Dalpiaz, F., van der Werf, J. M. E. M., & Brinkkemper, S. (2017). Extracting conceptual models from user stories with Visual Narrator. *Requirements Engineering*, 22(3), 339–358.

- Luna, E. R., Rossi, G., & Garrigos, I. (2011). WebSpec: a visual language for specifying interaction and navigation requirements in web applications. *Requirements Engineering*, 16(4), 297–321.
- Maalem, S., & Zarour, N. (2016). Challenge of validation in requirements engineering. *Journal of Innovation in Digital Ecosystems*, 3(1), 15–21.
- Macaulay, L. (1993). Requirements Capture as a Cooperative Activity. *Proceedings of the IEEE International Symposium*, 174–181.
- Maletic, J. I., Marcus, A., & Collard, M. L. (2002). A task oriented view of software visualization. *Proceedings of the 1st International Workshop on Visualizing Software for Understanding and Analysis*, 32–40.
- Mandel, T. (1997). *The elements of user interface design*. John Wiley & Sons.
- Marcus, A., Xie, X., & Poshyvanyk, D. (2005). When and how to visualize traceability links? *Proceedings of the 3rd International Workshop On Traceability in Emerging Forms of Software Engineering*, 56.
- Martins, G., L. E., & Gorschek, T. (2016). Requirements engineering for safety-critical systems: A systematic literature review. *Information and Software Technology*, 75, 71–89.
- Martins, N., Da Veiga, A., & Eloff, J. H. (2007). Information security culture – validation of an assessment instrument. *Southern African Business Review*, 11(1), 147–166.
- Mat, A. (2009). Applying SOFL to construct formal specification an automatic automobile driving simulation system. *Software Technology and Engineering*, 42–48.
- Mattmann, I., Gramlich, S., & Kloberdanz, H. (2016). Getting Requirements Fit for Purpose - Improvement of Requirement Quality for Requirement Standardization. *Procedia CIRP*, 50, 466–471.
- Méry, D., Lorraine, U. De, & Nancy, V. (2012). Critical systems development methodology using formal techniques. *Proceedings of the 3rd Symposium On Information and Communication Technology*, 3–12.
- Microsoft. (2017). UX checklist for desktop applications (Online). Retrieved from [https://msdn.microsoft.com/en-us/library/windows/desktop/dn742479\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/dn742479(v=vs.85).aspx) on September 13, 2017,

Microsystems, S. (2001). *Java look and feel design guidelines: advanced topics*. Addison-Wesley Professional.

Mit, E. (2015). Formalize the software quality measurement for heterogeneous requirements. *Proceeding of the 9th International Conference on IT in Asia (CITA)*, 1–4.

Moody, D. L., & Heymans, P. (2010). Visual syntax does matter: improving the cognitive effectiveness of the i * visual notation. *Requirements Engineering*, 141–175.

Mullen, P. M. (2003). Delphi: Myths and reality. *Journal of Health Organization and Management*, 17(1), 37–52.

Myers, B. (1986). Visual programming, programming by example, and program visualization: a taxonomy. *ACM Sigchi Bulletin*, 17 (4), 59–66.

Myers, B. (1990). Taxonomies of visual programming and program visualization. *Visual Languages and Computing*, 1(1), 97–123.

Naskovska, K., Lau, S., Aboughazala, A., Haardt, M., & Haueisen, J. (2017). Joint MEG-EEG signal decomposition using the coupled SECSI framework: Validation on a controlled experiment. *Proceeding of the IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, 1–5.

Neill, C. J., & Laplante, P. A. (2003). Requirements engineering: the state of the practice. *IEEE Software*, 20(6), 40–45.

Nicolás, J., & Toval, A. (2009). On the generation of requirements specifications from software engineering models: A systematic literature review. *Information and Software Technology*, 51(9), 1291–1307.

Nielsen, J. (2005). Ten Usability Heuristics.

Ogata, S., & Matsuura, S. (2010). Evaluation of a use-case-driven requirements analysis tool employing web UI prototype generation. *WSEAS Transactions on Information Science and Applications*, 7(2), 273–282.

Okoli, C., & Pawlowski, S. D. (2004). The Delphi method as a research tool: an example, design considerations and applications. *Information and Management*, 42(1), 15–29.

- Oppermann, Reinhard, H. R. (1997). Software evaluation using the 9241 evaluator. *Behaviour and Information Technology*, 16(4), 232–245.
- Ott, D. (2012). Defects in natural language requirement specifications at Mercedes-Benz: An investigation using a combination of legacy data and expert opinion. *Proceeding of the 20th IEEE International Requirements Engineering Conference (RE)*, 291–296.
- Paech, B., Koenig, T., Borner, L., & Aurum, A. (2005). An analysis of empirical requirements engineering survey data. *Engineering and Managing Software Requirements*, 427–452.
- Petre, M. (1995). Why looking isn't always seeing: readership skills and graphical programming. *Communications of the ACM*, 36(6), 33–44.
- Pfleeger, S. L., & Kitchenham, B. (1996). Software quality: the elusive target. *IEEE Software*, 12–21.
- Plomp, C. J., & Mayora-Ibarra, O. (2002). A generic widget vocabulary for the generation of graphical and speech-driven user interfaces. *International Journal of Speech Technology*, 5(1), 39–47.
- Pohl, K. (1993). The three dimensions of requirements engineering. *Proceeding of the International Conference on Advanced Information Systems Engineering*, 275–292.
- Pohl, K. (2010). *Requirements engineering: fundamentals, principles, and techniques*. Springer Publishing Company, Incorporated.
- Pohl, K., & Rupp, C. (2011). *Requirement engineering fundamentals*. Rocky Nook Inc.
- Polit, D. F., & Beck, C. T. (2006). The content validity index: are you sure you know what's being reported? Critique and recommendations. *Research in Nursing and Health*, 29(5), 487–497.
- Polit, D. F., Beck, T., & Owen, S. V. (2007). Focus on research methods is the CVI an acceptable indicator of content validity. *Res Nurs Health*, 30, 459–467.
- Price, B. A., Baecker, R. M., & Small, I. S. (1993). A principled taxonomy of software visualization. *Journal of Visual Languages and Computing*, 4(3), 211–266.

Räbiger, S., & Spiliopoulou, M. (2015). A framework for validating the merit of properties that predict the influence of a twitter user. *Expert Systems with Applications*, 42(5), 2824–2834.

Ramón, Ó. S., Cuadrado, J. S., Molina, J. G., & Vanderdonckt, J. (2016). A layout inference algorithm for graphical user interfaces. *Information and Software Technology*, 70, 155–175.

Rodrigues, A. (2014). Quality of requirement specification: preliminary of an automatic validation approach. *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, 1021–1022.

Rogers, Y., Sharp, H., & Preece, J. (2011). *Interaction design: Beyond human-computer interaction*. John Wiley & Sons.

Roman, G. C., & Cox, K. C. (1993). A taxonomy of program visualization systems. *Computer*, 26(12), 11–24.

Rudd, J., Stern, K., & Isensee, S. (1996). Low vs. high-fidelity prototyping debate. *Interactions*, 3(1), 76–85.

Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering*, 14(2), 131–164.

Sabatucci, L., Ceccato, M., Marchetto, A., & Susi, A. (2015). Ahab's legs in scenario-based requirements validation: An experiment to study communication mistakes. *Journal of Systems and Software*, 109, 124–136.

Salman, I., Misirli, A. T., & Juristo, N. (2015). Are students representatives of professionals in software engineering experiments? *Proceedings of the IEEE/ACM 37th IEEE International Conference on Software Engineering*, 666–676.

Sanchez-Gordon, M. L., de Amescua, A., O'Connor, R. V., & Larrucea, X. (2017). A standard-based framework to integrate software work in small settings. *Computer Standards and Interfaces*, 54, 162–175.

Sefelin, R., Tscheligi, M., & Giller, V. (2003). Paper prototyping—what is it good for?: a comparison of paper-and computer-based low-fidelity prototyping. *Proceedings of the CHI'03 Extended Abstracts On Human Factors in Computing Systems*, 778–779.

Shneiderman, B. (2010). *Designing the user interface: strategies for effective human-computer interaction*. Pearson Education India.

- Shu, N. C. (1988). *Visual programming*. New York: Van Nostrand Reinhold.
- Siegemund, K., Zhao, Y., Pan, J. Z., & Aßmann, U. (2012). Measure software requirement specifications by ontology reasoning. *Proceeding of 8th International Workshop on Semantic Web Enabled Software Engineering (SWESE'2012)*.
- Singer, J., Storey, M., & Damian, D. (2002). Selecting empirical methods for software engineering research. *Guide to Advanced Empirical Software Engineering*, 285–311.
- Sommerville, I. (1998). *Requirements Engineering: Processes and techniques* (Vol. 10). Wiley.
- Sommerville, I. (2016). *Software Engineering GE*. Pearson Australia Pty Limited.
- Spichkova, M. (2014). Design of Formal Language and interfaces: “formal” does not mean “unreadable.” *Proceedings of the Emerging Research and Trends in Interactivity and the Human-Computer Interface*, 301–314.
- Stasko, J. T., & Patterson, C. (1992). Understanding and characterizing software visualization systems. *Proceedings of the IEEE Workshop on Visual Languages*, 3–10.
- Stephen Haag, M.K. Raja, and L. L. S. (1996). Quality function deployment usage in software development. *Communications of the ACM*, 39(1), 42–49.
- Storey, M. A., Fracchia, F., & Müller, H. (1999). Cognitive design elements to support the construction of a mental model during software exploration. *Journal of Systems and Software*, 44(3), 171–185.
- Thitisathienkul, P., & Prompoon, N. (2015). Quality assessment method for software requirements specifications based on document characteristics and its structure. *Proceedings of the 2nd International Conference on Trustworthy Systems and Their Applications*, 51–60.
- Tichy, W. F. (2000). Hints for reviewing empirical work in software engineering. *Empirical Software Engineering*, 5(4), 309–312.
- Tiwari, S., Rathore, S. S., Gupta, S., Gogate, V., & Gupta, A. (2012). Analysis of use case requirements using sfta and sfmea techniques. *Proceedings of the IEEE 17th International Conference on Engineering of Complex Computer Systems*, 29–38.

- Tory, M. (2004). Rethinking visualization: A high-level taxonomy. *Proceedings of the IEEE Symposium on Information Visualization*, 151–158.
- Tory, M., & Möller, T. (2005). Evaluating visualizations: do expert reviews work? *IEEE Computer Graphics and Applications*, 25(5), 8–11.
- Uusitalo, L., Lehikoinen, A., Helle, I., & Myrberg, K. (2015). An overview of methods to evaluate uncertainty of deterministic models in decision support. *Environmental Modelling and Software*, 63, 24–31.
- Velásquez, I., Caro, A., & Rodríguez, A. (2018). Kontun: A Framework for recommendation of authentication schemes and methods. *Information and Software Technology*, 96, 27–37.
- Véras, P. C., Villani, E., Maria, A., & Vieira, M. (2014). A benchmarking process to assess software requirements documentation for space applications. *Journal of Systems and Software*, 100, 103–116.
- Wleringa, R., & Dubois, E. (1998). Integrating semi-formal and formal software specification techniques. *Information Systems*, 23(3–4), 159–178.
- Wohlin, C., Runeson, P., Martin, H., Magnus, O. C., Bjorn, R., & Anders, W. (2012). *Experimentation in software engineering*. Springer Berlin.
- Wolf, W. (2003). A decade of hardware/software codesign. *Computer*, (4), 38–43.
- Würfel, D., Lutz, R., & Diehl, S. (2016). Grounded requirements engineering: An approach to use case driven requirements engineering. *Journal of Systems and Software*, 117, 645–657.
- Yaman, S. G., Suvola, T., Riungu-Kalliosaari, L., Hokkanen, L., Kuvaja, P., Oivo, M., & Männistö, T. (2016). Customer involvement in continuous deployment: a systematic literature review. *Proceedings of the International Working Conference on Requirements Engineering: Foundation for Software Quality*, 249–265.
- Yang, C., Liang, P., Avgeriou, P., Eliasson, U., Heldal, R., Pelliccione, P., & Bi, T. (2017). An industrial case study on an architectural assumption documentation framework. *Journal of Systems and Software*, 134, 190–210.
- Yin, R. K. (1994). *Case study research - design and methods. Applied social research methods series (Vol. 5)*. Sage Publications.

Zafar, S., Farooq-khan, N., & Ahmed, M. (2015). Requirements simulation for early validation using Behavior Trees and Datalog. *Information and Software Technology*, 61, 52–70.