Software testing: A tertiary survey of the literature

Rosa Cobeñas1, Julio Quispe2, Abraham Dávila3

1Universidad Privada de Ciencias Aplicadas, Lima, Perú,

2Universidad Nacional Tecnológica de Lima Sur, Lima, Perú

3Pontificia Universidad Católica del Perú, Lima, Perú

pcsircob@upc.edu.pe, jquispe@untels.edu.pe, abraham.davila@pucp.edu.pe

**Abstract.** The continuous evolution software emphasizes the importance of testing to ensure the quality of the product being developed. However, the dispersion of secondary studies hinders the understanding of the testing domain. The objective of this paper is to organize and synthesize the secondary studies available in the literature on software testing. To achieve this, a tertiary literature survey was conducted following systematic mapping guidelines. A total of 900 articles were sourced from four relevant databases. Of these, 102 secondary studies were selected for analysis. These studies revealed that the most extensively researched topics are test techniques and test processes. Additionally, the publication rate has consistently averaged over 10 secondary studies per year over the past nine years.

**Keywords:** tertiary study, software testing, survey of the literature, secondary study

1 Introduction

The rapid technological development and deployment of various information technology (IT) solutions globally [1] have significantly impacted the business environment, government sector, and society in general [2]. This has made software and system solutions critical to societal development. In this context, software testing is a vital activity in the system life cycle [3] and software life cycle [4] because it ensures the quality of the product or service.

Correspondingly, in the field of research and development, numerous primary and secondary studies addressing various research domains have been published. These include artificial intelligence [5], [6], [7], [8]; mutation testing [9], [10], [11]; software test automation [12], [13]; and cloud testing [14], [15].

Notwithstanding the extensive research, only Garousi's [16] is relevant as a tertiary study in the software testing domain. However, after eight years, it is required to organize secondary studies to date. Thus, this research aims to conduct a tertiary study to organize secondary studies selected (SSS) and address a series of research questions related to software testing. Researchers will be able to identify current knowledge topics in the field of testing, aligned with SWEBOK (Software Engineering Body of Knowledge). This article is structured as follows: Section 2 presents related work; Section 3 describes the study design; Section 4 outlines the study results and discussion; and Section 5 presents the conclusions.

2 Related Work

In a preliminary search, three articles on tertiary reviews were identified. The one by Garousi [16] was the closest to the present study.

Article [17] reports a tertiary study focused on the process of test case selection and prioritization in the field of software testing. It includes the trends, approaches used, tools, and challenges related to software test case selection and prioritization. The study includes an analysis of the evolution of studies on the selection and prioritization of test cases, highlighting effective methodologies and areas where more research is needed.

Article [18] reports an investigation of the areas, tools, and challenges in model-based testing. In addition, it highlights the importance of thoroughly analyzing the techniques used in secondary studies for test case selection and prioritization. In addition, the authors point out that more empirical evidence is still needed.

Finally, [16] identified secondary studies predominantly focus on methods such as model-based testing, search-based testing, and mutation testing, whereas other techniques such as combinatorial testing, symbolic execution, and randomized testing are addressed less frequently.

3 Tertiary Study Design

This section presents the planning of the tertiary study [19], [20] considering the guidelines of systematic mapping available in the literature [21], [22].

3.1. Planning—Identification and Scope of Need

A first relevant tertiary study is the Garousi study [16], which was considered until 2015. However, a significant number of secondary studies have been published since the Garousi study [16]; in order to consider the new publications (from 2015 to 2024) a new tertiary study was performed. The research questions established and their motivation are shown in Table 1. Additionally, it was considered that the RQ3 is the same research question from the first Garousi study [16].

For the RQ3 the classification topics considered are:

* Topics of SWEBOK (Software Engineering Body of Knowledge): software testing fundamentals, software testing tools, test level, test process, test techniques, test related measures.
* From Garousi study [16]:
  + Software testing method: Model-based, mutation, search-based, combinatorial, symbolic execution, random, other.
  + Type of Software Under Test (SUT): Service-Oriented Architecture (SOA)/web-service, product-line, GUI (graphical user interface), web-application, component-based software, aspect-oriented software, cloud, protocol, mobile, concurrent.
  + Software testing type/phase: Regression, unit, integration, system, acceptance testing.

**Table 1.** Research questions and their motivation.

|  |  |
| --- | --- |
| Research question | Motivation |
| RQ1 How many SSSs were published between 2015 to today? | The purpose of this question is to determine the academic community's interest in the domain at the time. |
| RQ-2 In which media have the SSSs been published? | This question helps identify the most representative media in the domain. |
| RQ-3 What software testing specific areas that have been investigated in the SSSs? | This question aims to characterize the research in the testing domain. In this case, the topics identified in the testing chapter of SWEBOK and the classification from Garousi’s study [16] were used. |
| RQ-4 What are the citation levels of the SSSs? | This question provides insight into the influence of the SSSs from Google Scholar, Web of Science, and Scopus. |

3.2 Planning—Identification of the study

To start the selection process of secondary studies, a semi-automatic search strategy was employed on four relevant databases: Scopus, IEEE Xplore, ACM Digital Library, and Web of Science (WoS). The search string was constructed using the population intervention (PI) method [22]. The terms for population (P) were "systems testing" OR "software testing". Meanwhile, those for intervention (I) were "review of studies" OR "structured review" OR "systematic review" OR "literature review" OR "literature analysis" OR "in-depth survey" OR "literature survey" OR "meta-analysis" OR "past studies" OR "subject matter expert analysis of research" OR "empirical body of knowledge" OR "overview of existing research" OR "body of published research" OR "study aggregation" OR "study synthesis" OR "critical review" OR "mapping study" OR "mapping literature". This was based on [20] and the previous experience with other literature reviews. In the population (P), the term system testing was introduced to broaden the search, but then it was decided to leave only software testing to compare some results with the Garousi’s study.

The following inclusion criteria (IC) and exclusion criteria (EC) were considered:

* IC.1. Secondary studies focused on software testing.
* IC.2. Secondary literature review studies.
* IC.3. Secondary studies written in English.
* EC1. Secondary studies unrelated to software testing.
* EC2. Tertiary or primary education studies.
* EC.3. Duplicate studies.
* EC.4. Studies unavailable in full text.

The quality assessment was performed according to [19] using established criteria and the acceptance threshold was set at > 2.5 (see Table 2).

**Table 2.** Criteria for evaluating secondary studies; adapted from [19]

|  |  |  |
| --- | --- | --- |
| Criteria | Score | Description |
| Are the review’s inclusion and exclusion criteria described and appropriate? | 1: Y | The inclusion criteria are defined explicitly in the document. |
| 0,5: Partially | The inclusion criteria are implied. |
| 0: No | The inclusion criteria are not defined and cannot be inferred conveniently. |
| Is the literature search likely to have covered all relevant studies? | 1: Y | The authors have searched at least four digital libraries and included additional search strategies or identified and referenced all relevant journals. |
| 0,5: Partially | The authors have searched three–four digital libraries without additional search strategies, or have searched a defined albeit restricted set of journals and conference proceedings |
| 0: No | The authors have searched up to two digital libraries or an exceptionally restricted set of journals. |
| Did the reviewers assess the quality/validity of the included studies? | 1: Y | The authors have explicitly defined the quality criteria and extracted these from each primary study. |
| 0,5: Partially | The research question involves quality issues that are addressed by the study. |
| 0: No | An explicit assessment of the quality of individual documents has not been attempted. |
| Were the basic data/studies adequately described? | 1: Y | Detailed information on each document is presented. |
| 0,5: Partially | Only summarized information on individual documents is presented. |
| 0: No | The results of the individual studies are not specified. |

3.3. Planning—Selection, extraction, and categorization process

In this study, the following IC and EC were applied in stages to ensure a thorough and systematic selection of secondary studies:

* First stage: The search strings are executed according to each database. The titles of conferences, proceedings, symposiums, and workshops are eliminated. Duplicates across databases (EC.3) are removed.
* Second stage: Studies from 2015 onward are selected. Each title is reviewed, and IC.1, IC.2, IC.3, and EC.1 are applied.
* Third stage: Each abstract is read, and EC.2 and EC.3 are applied.
* Fourth stage: The secondary study is downloaded, and a rapid reading of the secondary study is performed. EC.1, EC.2, and EC.4 are applied.
* Fifth stage: The quality of the selected studies is evaluated using the established criteria. Only studies with acceptance scores higher than 2.5 are included.

A data extraction form was established to answer the research questions. This structure included the study identifier, article title, year of publication (RQ1), name of place of publication (RQ2), type of place of publication (RQ2), topic area of knowledge (RQ3), and number of citations (RQ4). For RQ3, the first-level structure of SWEBOK v3 was used to categorize the secondary studies. It includes the following elements: software testing fundamentals, test levels, test techniques, test-related measures, test process, and software testing tools. For RQ4, the citations were obtained from Google Scholar, Web of Science, and Scopus to prevent bias.

3.4. Validity threat analysis

In SMS (systematic mapping study) and SLR (systematic literature review) studies, validity considerations are important [22]. Potential threats, such as inaccuracy in data extraction and limitation of the set of studies, due to selected search terms and academic engines, as well as bias in inclusion or exclusion criteria, should also be addressed and mitigated. These threats are analyzed according to the four standard validity types [21], [22], [23].

* Internal validity, for this purpose, search engines, terms and inclusion and exclusion criteria are precisely defined to ensure the reproducibility of the results. In the process, challenges may arise related to the selection of studies and biases applied.
* Construct validity, for this purpose a formal search is established and the inclusion of relevant and extensive academic databases. It is expected that the data collection will be complete and that any omissions will be negligible. In addition, the application of inclusion and exclusion criteria may be affected by personal experiences.
* Validity of conclusion, for this it is based on recognized standards such as SWEBOK v3 that allows to support the identified themes. The review of secondary studies by authors reduces bias in data extraction and resolve discrepancies by consensus.
* External validity, for this, establishes that the generalization of results has value only within the specific scope of secondary studies in software tests; considering secondary studies in English and limiting to the four digital databases considered.

4 Study Results and Discussion

This section presents the selection process and the answers to the research questions. Appendix A: Selection Process details the development of the selection process.

The search string was executed in February 2024 and adapted to each database according to its specific constraints. The records obtained were consolidated and processed in a spreadsheet. The complete list is presented in Appendix B: Selected Secondary Studies. The process (see Fig. 1) was as follows:

* First stage: The search strings were executed and the result is shown in Fig 1.
* Second stage: The titles were reviewed. This reduced the count to 193 articles.
* Third stage: The abstracts were read. This further narrowed the selection to 189 articles.
* Fourth stage: A rapid reading of each study was conducted. This left 159 articles.
* Fifth stage: The scores of these articles according to the criteria [19] are presented in Appendix C: Evaluation of secondary studies. Finally, this left 102 articles as secondary study selected (SSS).

Diagrama

Descripción generada automáticamente

**Fig. 1.** Selection process.

The answers to the research questions are presented below:

4.1. RQ-1 How has the number of SSS publications evolved?

The temporal evolution of the number of SSSs from 2015 to 2024 is presented in Fig. 2. An upward trend is observed from 2015 to 2019. However, a decrease from 19 to 15 secondary studies is observable from 2019. Subsequently, the number of studies per year reduced to six by 2023. Excluding the partial data for 2024, the average number of publications has been 11.1 articles per year.

With regard to the evolution of the SSSs, our research revealed a large number of publications. This is similar to what was reported in Garousi’s study [16]. For example, the annual average was 12.5 articles from 2015 to 2023, and it was 13.5 articles between 2009 and 2014. These numbers are essentially similar notwithstanding the one-point difference. In addition, [16] includes secondary studies since 2009. It showed a continuous increase to a peak of 22 SSSs in 2014. In contrast, our study revealed a maximum of 19 SSSs for the period from 2015 to 2024.

4.2. RQ-2: In what media have the SSSs been published?

Table 3 lists the journals and conferences with at least two publications. Meanwhile, Appendix C provides the complete list of SSS publication sources. "Information and Software Technology" leads with 18 published secondary studies. It is followed by "ACM International Conference Proceeding Series" with 11 and "Journal of Systems and Software" with 6 published studies. Meanwhile, Fig. 3 shows the distribution of selected studies according to the publication source from 2015 to 2024. It emphasizes the predominance of specialized journals and academic conferences. In 2019, the largest number of publications was recorded, with 10 secondary studies published in journals and 9 in conferences. Since 2016, the number of articles published in journals has been higher than those published in conferences. This is in line with the trend reported by [16].

Gráfico, Gráfico de líneas

Descripción generada automáticamente

**Fig. 2.** Publications by year.

**Table 3.** Venues that have more than one publication

|  |  |  |  |
| --- | --- | --- | --- |
| Venue | J/C1 | Q2 | Study |
| 1. Information and Software Technology | J | 18 | S005, S013, S017, S018, S028, S029, S034, S040, S043, S044, S045, S050, S062, S081, S082, S085, S096, S102 |
| 2. ACM International Conference Proceeding Series | C | 11 | S008, S023, S024, S038, S048, S053, S073, S078, S088, S097, S101 |
| 3. Journal of Systems and Software | J | 6 | S007, S011, S052, S055, S066, S091 |
| 4. Computer Science Review | J | 4 | S002, S025, S033, S090 |
| 5. Software Testing Verification and Reliability | J | 4 | S027, S057, S059, S100 |
| 6. ACM Computing Surveys | J | 3 | S012, S022, S093 |
| 7. Empirical Software Engineering | J | 3 | S036, S046, S098 |
| 8. IEEE Access | J | 3 | S003, S016, S031 |
| 9. Intern. Conf. on Software Testing, Verification and Validation Workshops | C | 2 | S035, S058 |
| 10. Ibero-American Conference on Software Engineering | C | 2 | S010, S076 |
| 11. Euro-micro Conf. on Software Engineering and Advanced Applications | C | 2 | S067, S083 |
| 12. Software Quality Journal | J | 2 | S032, S060 |

1 J is Journal and C is Conference.

2 Q is quantity of secondary studies

Gráfico, Gráfico de líneas

Descripción generada automáticamente

**Fig. 3.** Secondary Studies Selected by type of publication source.

4.3. RQ-3: What software testing specific areas that have been investigated in the SSSs?

In this question, the topics identified in the testing chapter of SWEBOK and the topics used in the Garousi's study [16] were used for the same questions.

Table 4 presents the distribution of studies according to the software testing knowledge area topics of SWEBOK. The column title “Q” shows the quantity of SSS, while the column “F (%)” shows the cumulative frequency. Test technique is the most important topic reported in the SSS, and according to the Pareto analysis, the test process and test-related measures are also the important topics reported in the SSS. This indicates their significant relevance and discussion in the academic and professional literature on software testing.

**Table 4.** SSS organized by topics of SWEBOK

|  |  |  |  |
| --- | --- | --- | --- |
| Topic of SWEBOK | Q | F (%) | Studies |
| Test Techniques | 51 | 50% | S001, S002, S003, S004, S007, S011, S013, S015, S019, S020, S022, S024, S025, S031, S032, S033, S039, S043, S044, S045, S048, S050, S052, S053, S055, S057, S058, S059, S060, S061, S063, S065, S066, S068, S069, S070, S072, S074, S075, S079, S082, S083, S084, S085, S087, S088, S093, S094, S096, S097, S102 |
| Test Process | 16 | 66 | S006, S009, S014, S016, S021, S027, S034, S036, S037, S038, S040, S080, S086, S089, S092, S100 |
| Test-Related Measures | 13 | 78 | S005, S008, S017, S023, S026, S028, S029, S035, S056, S062, S064, S078, S098 |
| Software Testing Tools | 11 | 89 | S030, S041, S049, S051, S054, S067, S076, S090, S095, S099, S101 |
| Software Testing Fundamentals | 10 | 99 | S012, S018, S042, S046, S047, S071, S073, S077, S081, S091 |
| Test Levels | 1 |  | S010 |

This, in turn, implies that the effectiveness and application of specific testing techniques are critical areas of interest and focus for researchers and practitioners. In contrast, topics such as testing levels are discussed considerably less. The analysis provides valuable insights for identifying the most relevant and potentially underdeveloped areas of research and practice in the field of software testing. This would facilitate a more informed direction for future research and practice in the discipline. In particular, since 2023, eight SSS have been published, of which three are related to artificial intelligence and metaheuristics (S084, S094, S074), two related to security (S094, S089) and five related to some aspect of testing in general (S014, S035, S085, S093).

Based on the classification used in Garousi’s study [16], the SSS were organized by software testing methods in Table 5, type of SUT in Table 6, and software testing phase in Table 7. The Garousi’s study did not include any studies that focused on a long domain in software testing in the established categories.

**Table 5.** Classification by software testing methods

|  |  |  |  |
| --- | --- | --- | --- |
| Software testing method | Q | Studies | Garousi |
| Model-based | 8 | S021, S023, S031, S032, S033, S034, S043, S051, | 15 |
| Mutation | 6 | S002, S013, S025, S059, S066, S088 | 7 |
| Search base | 0 | --- | 7 |
| Combinatorial | 2 | S008, S058 | 4 |
| Symbolic execution | 1 | --- | 2 |
| Random | 0 | --- | 2 |
| Agent-Based | 1 | S068 | N.C. |
| Requirement-based | 1 | S075 | N.C. |
| Bayesian-based test generation | 1 | S077 | N.C. |

**Table 6.** Type of software under testing

|  |  |  |  |
| --- | --- | --- | --- |
| Type of SUT | Q | Studies | Garousi |
| Mobile | 7 | S011, S022, S024, S028, S059, S067, S076 | 2 |
| Cloud | 4 | S012, S052, S065, S080 | 2 |
| Web-application | 2 | S001, S007 | 3 |
| Concurrent | 2 | S069, S082 | 2 |
| SOA/web-service | 2 | S090, S092 | 9 |
| Adaptative systems | 2 | S078, S100 | N.C. |
| Testing non-testable systems | 1 | S060 | N.C. |
| Internet of things | 1 | S073 | N.C. |
| Microservices | 1 | S047 | N.C. |
| Product-line | 1 | S021 | 8 |
| GUI | 0 |  | 5 |
| Protocol | 0 |  | 2 |
| Aspect-oriented software | 0 |  | 2 |
| Component-based software | 0 |  | 2 |

**Table 7.** Software testing phase

|  |  |  |  |
| --- | --- | --- | --- |
| Software testing phase | Quantity | Studies | Garousi |
| Regression | 5 | S006, S037, S044, S068, S093 | 11 |
| Integration | 2 | S010, S085 | 3 |
| Unit | 0 |  | 2 |
| System | 0 |  | 0 |
| Acceptance testing | 1 | S084 | 1 |

Table 5 shows that model-based and mutation methods are the most frequently reported in SSS, which aligns with what they report in [16]. However, in the case of related search-based there is a major difference. The Garousi study test case generation work has been considered as search-based. In this study, it was not generalized, and based on other authors, it was included independently, in such situation are: agent-based [24], requirement-based [25] and Bayesian-based test generation [26]. The abbreviation N.C. means that this method was not considered in [16]. In the Garousi study, there is a small mistake in the software test method table; because 67 studies were reported in search-based, but they should only be seven according to the studies referenced in the same table.

Table 6 shows that the most frequently reported types of SUT are mobile and cloud, with 7 and 4 SSS respectively. In contrast to [16]. it was observed that the SOA/Web services and Product line types were less studied, and even no secondary study of GUI was reported. The current emphasis is in line with the level of development of these types of software over the past few years.

Table 7 shows that the software testing phase reported in SSS is consistent with the testing phase reported in the study [16].

4.4. RQ-4: What are the levels of SSS cited?

To determine the level of SSS cited, we first obtained only the top 10 from each digital database, such as Google, Scopus, and WoS. After that, the table was created from the original lists, and empty fields appeared. These empty fields were filled with NC (not considered). Finally, it was averaged and ordered in a descending form.

It can be observed that the final top 10 (average column) is identical to those of Google column. All of them have over 100 citations on Google Scholar. The complete list is presented in Appendix D: Citations of SSSs in Google Scholar. Study S102 from 2016 is prominent with 238 citations. This number is attributed to its subject matter and publication date. Similarly, S044 from 2018 and S011 from 2016 also have large citation counts of 230 and 190, respectively. The large number of citations for S044 can be attributed to its age and relevant topic. Meanwhile, the contemporary subject matter of S011 has contributed to its significant citation count.

6 Conclusions

This paper presents a tertiary survey of the literature on software testing. Through a systematic process, we obtained 900 records from four relevant databases and ultimately selected 102 articles as SSSs. These studies were used to address and contrast four key research questions with related works.

**Table 8.** Top SSS cited based on the average of top 10 cited in relevant digital database

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SSS | Average | Google | Scopus | WoS |
| S044 | 169 | 230 | 154 | 124 |
| S102 | 151 | 238 | 117 | 99 |
| S011 | 133 | 190 | 111 | 98 |
| S046 | 130 | 185 | 106 | 100 |
| S033 | 122 | 122 | NC | NC |
| S050 | 108 | 108 | NC | NC |
| S030 | 101 | 165 | 83 | 56 |
| S036 | 91 | 149 | 67 | 57 |
| S040 | 80 | 108 | 71 | 60 |
| S003 | 78 | 105 | 70 | 58 |
| S058 | 60 | NC | 60 | NC |
| S013 | 57 | NC | 62 | 52 |
| S096 | 51 | NC | NC | 51 |

With regard to the evolution of SSSs, it has been established that the volume of publications has averaged 13 articles per year over the past two decades. This reflects the scientific community’s sustained interest in this critical area. Similarly, these results indicate that although the domain is a mature field of research, it continues to evolve with technological advancements. With regard to the publication medium, it was determined that the largest number of SSSs was published in journals, with 23 SSSs published in the journal "Information and Software Technology" and 17 in "ACM International Conference Proceeding Series". With regard to the research topics, using the SWEBOK V3 categorization, we identified that "test techniques" has been an area of significant interest for over two decades, whereas "test process" has gained relevance in the past decade. According to Table 8, it was determined that the most cited articles are S044 from 2018 (titled *"Test case prioritization approaches in regression testing: A systematic literature review"*)with 230 citations and S102 from 2016 (titled *"When and what to automate in software testing? A multi-vocal literature review"*)with 238 citations.

It can be concluded that this study has provided an updated perspective of the software testing domain and thereby, feasible future research directions. In addition, the breadth and depth of the topics reflected in the number of SSSs emphasize the importance of continued evaluation and advancement in this field.

Acknowledgments.

This work was partially supported by the Software Engineering Development and Research Group of the Department of Engineering of Pontificia Universidad Católica del Perú and by Universidad Nacional Tecnológica de Lima Sur.

Appendix [URL](https://drive.google.com/drive/folders/1asHiPydqdK-YyZM7uaFxsWk7X0-exGiw?usp=drive_link)

Appendix A: Selection Process: details the development of the selection process.

Appendix B: Selected secondary studies.

Appendix C: Evaluation of secondary education.

Appendix D: SSS citations in Google Scholar.

References

1. Castro, G., Fustiel, Y., Palacios, F., Cabrera, I., Cárdenas, R.: Internacionalización y localización del software en el ámbito mundial. INNOVA Research Journal. 2, 99–111 (2017). https://doi.org/10.33890/innova.v2.n8.1.2017.347.

2. Enríquez, A., Sáenz, C.: Gobierno Digital. Pieza clave para la consolidación de Estados democráticos en los países del SICA. (2022).

3. ISO/IEC/IEEE: ISO/IEC/IEEE 15288:2017 Systems and Software Engineering -- System Life Cycle Processes. , Geneva (2017).

4. ISO/IEC/IEEE: ISO/IEC/IEEE 12207:2017 Systems and Software Engineering -- Software Life Cycle Processes. , Geneva (2017).

5. De Lima, L., Peres, L., Grégio, A., Silva. Fabiano: A systematic literature mapping of artificial intelligence planning in software testing. (2020).

6. Lima, R., Rosado, A., Ribeiro, J.: Artificial intelligence applied to software testing a: literature review conference. (2020).

7. Trudova, A., Dolezel, M., Buchalcevova, A.: Artificial Intelligence in Software Test Automation: A Systematic Literature Review. (2020).

8. Islam, M., Khan, F., Alam, S., Hasan, M.: Artificial intelligence in software testing: a systematic review. (2023).

9. Silva, H., Lima, J., Vergilio, S., Endo, A.: A mapping study on mutation testing for mobile applications. Software Testing Verification and Reliability. 32, (2022). https://doi.org/10.1002/stvr.1801.

10. Jeevarathinam, R., Thanamani. Antony: A survey on mutation testing methods, fault classifications and automatic test cases generation. (2011).

11. Haugset, B., Hanssen, G.: Automated acceptance testing: a literature review and an industrial case study. (2008).

12. Musthafa, F., Mansur, S., Wibawanto, A.: Automated software testing on mobile applications: a review with special focus on android platform. Presented at the (2020).

13. Mendez.Porras, A., Quesada-López, C., Jenkins, M.: Automated testing of mobile applications: A systematic map and review. (2015).

14. Bertolino, A., De Angelis, G., Gallego, M., García, B., Gortázar, F., Lonetti, F., Marchetti, E.: A Systematic Review on Cloud Testing. ACMComput. Surv. (2019).

15. Ahmad, A., Brereton, P., Andras, P.: A Systematic Mapping Study of Empirical Studies on Software Cloud Testing Methods. In: 2017 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C). pp. 555–562. IEEE (2017). https://doi.org/10.1109/QRS-C.2017.94.

16. Garousi, V., Mäntylä, M. V.: A systematic literature review of literature reviews in software testing. Inf Softw Technol. 80, 195–216 (2016). https://doi.org/10.1016/j.infsof.2016.09.002.

17. Singhal, S., Jatana, N., Suri, B., Misra, S., Fernandez-Sanz, L.: Systematic literature review on test case selection and prioritization: A tertiary study. Applied Sciences (Switzerland). 11, (2021). https://doi.org/10.3390/app112412121.

18. Villalobos-Arias, L., Quesada-López, C., Martinez, A., Jenkins, M.: A tertiary study on model-based testing areas, tools and challenges: Preliminary results. In: Avances en Ingenieria de Software a Nivel Iberoamericano, CIbSE 2018. pp. 15–28 (2018).

19. Kitchenham, B., Brereton, P., Budgen, D., Turner, M., Bailey, J., Linkman, S.: Protocol for a Tertiary study of Systematic Literature Reviews and Evidence-based Guidelines in IT and Software Engineering. (2007).

20. Kitchenham, B., Pretorius, R., Budgen, D., Brereton, O.P., Turner, M., Niazi, M., Linkman, S.: Systematic literature reviews in software engineering – A tertiary study. Inf Softw Technol. 52, 792–805 (2010). https://doi.org/10.1016/J.INFSOF.2010.03.006.

21. Petersen, K., Feldt, R., Mujtaba, S., Mattsson, M.: Systematic Mapping Studies in Software Engineering. International Journal of Software Engineering & Knowledge Engineering. 17, 33–55 (2007).

22. Petersen, K., Vakkalanka, S., Kuzniarz, L.: Guidelines for conducting systematic mapping studies in software engineering: An update. Inf Softw Technol. 64, 1–18 (2015). https://doi.org/10.1016/j.infsof.2015.03.007.

23. Claes Wohlin, Per Runeson, Martin Bost, Magnus c. Ohlsson, Bjorn Regnell, Anders Wesslin: Experimentation In Software Engineering - An Introduction. (1996).

24. Kumaresen, P.P., Frasheri, M., Enoiu, E.P.: Agent-Based Software Testing: A Definition and Systematic Mapping Study. Proceedings - Companion of the 2020 IEEE 20th International Conference on Software Quality, Reliability, and Security, QRS-C 2020. 24–31 (2020). https://doi.org/10.1109/QRS-C51114.2020.00016.

25. Whalen, M.W., Heimdahl, M.P.E., Rajan, A., Miller, S.R.: Coverage metrics for requirements-based testing. Proceedings of the 2006 International Symposium on Software Testing and Analysis, ISSTA 2006. 2006, 25–35 (2006). https://doi.org/10.1145/1146238.1146242.

26. Asgari Araghi, M., Rafe, V., Khendek, F.: Using data mining techniques to generate test cases from graph transformation systems specifications. Automated Software Engineering. 31, (2024). https://doi.org/10.1007/s10515-024-00417-0.