Ingeniería y competitividad
ISSN: 0123-3033
ISSN: 2027-8284
Facultad de Ingeniería, Universidad del Valle

Bastidas, María I.; Pardo, César J.; Ardila, Carlos A.
A Systematic Literature Mapping: risk-based testing in software development
Ingeniería y competitividad, vol. 23, no. 1, e9503, 2021, January-June
Facultad de Ingeniería, Universidad del Valle

DOI: https://doi.org/10.25100/iyc.v23i1.9503

ISSN: 0123-3033 – ISSN-e: 2027-8284
INGENIERIA Y COMPETITIVIDAD
REVISTA CIENTÍFICA Y TECNOLÓGICA
Vol. 23 No. 1-2021 - DOI: 10.25100/iyc.23i1.9503

SYSTEMS ENGINEERING

A Systematic Literature Mapping: risk-based testing in software development

INGENIERÍA DE SISTEMAS

Un mapeo sistemático de literatura: pruebas basadas en riesgo en el desarrollo de software

María I. Bastidas[1§], César J. Pardo[1], Carlos A. Ardila[1]

[1]Universidad del Cauca, Faculty of Electronic Engineering and Telecommunications, Systems Engineering Program, Information Technology Research and Development Group (GTI), Popayán, Colombia

[§]*mariaisabelb@unicauca.edu.co,cpardo@unicauca.edu.co,cardila@unicauca.edu.co*

Recibido: 28 de noviembre de 2019 – **Aceptado:** 03 de julio de 2020 – **Publicado online**:

Abstract

Risk-based testing (RBT) is a type of test that helps identify product risks from the start of development, incorporating techniques that allow them to be identified and classified according to their impact and probability to create test cases for those selected requirements. However, in software development organizations the identified risks are related to the planning or cost of the project to guarantee product delivery and do not consider other risks as input for the creation of test cases and quality evaluation. of the product. Therefore, the objective of systematic mapping is based on identifying and determining the state of the art of publications related to RBT used in the software industry, in addition to metrics that incorporate or evaluate the performance of these types of tests and their benefits. The results show the proposals found on the software industry RBT and the importance of use as other types of software testing. Also, we present a preview of the Framework to support the RBT in global software development.

Keywords: *Risk-based testing, Risk assessment, Software testing, Systematic mapping, Test management.*

Resumen

Las pruebas basadas en riesgos (PBR) son un tipo de prueba que ayuda a identificar los riesgos del producto desde el inicio de desarrollo, incorporando técnicas que permitan su identificación y ser clasificados según su impacto y probabilidad, de modo que permitan crear casos de prueba para aquellos requerimientos seleccionados. Sin embargo, en las organizaciones de desarrollo software los riesgos que se identifican tienen relación con la planificación o coste del proyecto para garantizar la entrega del producto y no consideran otros riesgos como elementos de entrada para la creación de casos de prueba y evaluación de la calidad del producto. Por lo tanto, el objetivo del mapeo sistemático se basa en identificar y determinar el estado del arte de las publicaciones relacionas con PBR utilizadas en la industria software, además de métricas que incorporen o evalúen el desempeño de este tipo de pruebas y sus beneficios. Los resultados obtenidos demuestran las propuestas encontradas sobre las PBR en la industria software y la importancia de

Cómo citar:
Bastidas, M. I., Pardo, C. J., & Ardila, C. A. (2020). Un mapeo sistemático de literatura: pruebas basadas en riesgo en el desarrollo de software. INGENIERÍA Y COMPETITIVIDAD, 23(1), 1-17. https://doi.org/10.25100/iyc.v23i1.9503.

uso como otro tipo de pruebas software. Así mismo, presentamos una vista previa de Framework para soportar las PBR en el desarrollo de software global.

Palabras clave: *Evaluación de riesgos, Gestión de pruebas, Mapeo sistemático, Pruebas basadas en riesgos, Pruebas de software.*

1. Introduction

Software development organizations use software testing to evaluate product quality during the development life cycle of their products/services [(1)]. A testing type that has started to be incorporated and studied is risk-based testing (RBT), which focuses on testing activities on those areas that trigger the most critical software system situations [(2,3)]. According to the international standard ISO/IEC/IEEE 29119, risks should be considered as a fundamental part of the testing process [(4)], where a risk element in the testing context is any tested value element, for example, a requirement, a component or an error [(2)]. In this case, when risks are identified they are prioritized according to both their probability and impact, and test cases are projected based on strategies for the identified risk factors treatment [(5)]. Therefore, RBT is a testing type that considers the software product risks to solve decision problems in all testing process phases, i.e., test planning, design, execution, and test evaluation [(2)].

Incorporating RBT into software projects from the first stages of product development will allow timely follow-up of testing until it is guaranteed the risk identified in the final product is not affected [(4)] and optimize the resource allocation such as budget, time and people [(6)]. On the other hand, it helps to mitigate the risks associated with the product, identifying those critical areas that may require it, and providing support for decision-making in the management of the project [(7)]. In this way, organizations can develop their software systems with more confidence and profitability, delivering a quality product, and reducing additional development work costs [(8)].

It is possible to observe that the proposals found have focused their efforts mainly on identifying the product risks from the requirements, analyzing them, and evaluating them to create methods and/or procedures that allow it to be incorporated into software development [(9)]. However, and although some of them propose some process elements, these are not detailed or described comprehensively, for example, in [(10)] a model is proposed which describes a set of phases and activities to be considered in the RBT, however, it does not describe in detail the activities presented, and the input and output artifacts. From the analysis of the literature, it has been possible to observe that research on RBT has great potential for application and cost savings [(6)].

This paper is a conference extension presented in [(11)], in contrast to the document presented above, we show the new results of the search made in other search engines such as IEEE Xplore, Redalyc and Google Scholar (section 3.2). Likewise, a table is added to present the classification used to establish a glossary of terms that enables to clarify the heterogeneity of the definitions regarding RBT (section 4.2), describing the metrics established by other authors to incorporate or evaluate RBT (section 4.3), extending the list of benefits and limitations from the new primary studies included (section 4.4). Additionally, the discussion of results has been updated (section 5.5), a preview of a Framework to support RBT in global software development is presented (section 6). Finally, the conclusions and future works are presented

(section 7). Considering the previous information, the importance and interest of the academic community in RBT benefits, this paper presents a systematic mapping of the literature on RBT proposals and related work. In addition to this introduction, the article is organized as follows: Section 2 presents related work. Section 3 carries out the planning of review. Section 4 presents the execution of the review on the selected sources. Section 5, the results obtained are analyzed and interpreted. Section 6 presents a framework preview to support RBT in global software development. Finally, section 7 presents conclusions and future work.

2. Related work

Software testing helps to improve product quality throughout the development lifecycle. Therefore, incorporating some testing types such as risk-based (RBT) testing allows the detection of errors in early stages allowing their correction and lower cost [(12)]. RBT is a testing-based approach to risk management [(13)], which considers the impact and likelihood of risk.

Besides, proposals have been found such as procedures, methods, approaches, models, methodologies, taxonomies, techniques, and frameworks. For example, a taxonomy of RBT provides a framework for understanding approaches to RBT and adapting them to specific purposes by including three types of approaches: risk drivers, risk assessment, and RBT [(9)]. It has also been possible to find a taxonomy where categorization is made between standards and approaches presented to incorporate RBT [(4)]. In [(14)], a light approach is presented to estimate the risk probability in software testing, using phases: (i) risk elements definition, (ii) probability, (iii) impact estimation, (iv) risk values calculation, (v) risk levels determination, (vi) testing strategy definition, (vii) testing strategy refinement. In [(15)], an approach through a quality assessment based on the quality and control model called QuaMoco is presented, creating two approaches: Approach 1: the quality assessment of each component and Approach 2: directly using the metrics at the lowest level of the quality model hierarchy. This type of proposal allows evidence solutions that help to incorporate this type of testing in the software industry. Although it has been possible to find related jobs, these are not detailed at the process element level to consider RBT related tasks or activities.

3. Research Protocol

Systematic mapping is a method for researching, collecting, and categorizing all existing information about a specific research topic. This systematic mapping has been carried out following the guidelines presented in the following works: Piattini et al. (16), Bocco et al. (17), Kitchenham (18), Petersen et al. (19) and Budgen et al (20). Systematic mapping is established in three stages: Planning, Execution, and Documentation. The first two stages are described in the following subsections and the documentation stage corresponds to section 4.

3.1. Planning Stage

This stage describes the sub-sections of each of the activities carried out: 3.1.1) Establishment of the research questions, 3.1.2) Definition of the search strategy, 3.1.3) Establishment of the selection criteria for the primary studies, 3.1.4) Establishment of the quality assessment criteria, 3.1.5) Definition of the data extraction strategy, 3.1.6) Synthesis methods selection.

3.1.1 Research questions

The main objective of this systematic mapping is based on identifying through the state of art publications related to RBT and their contribution to the software industry. Therefore, the research questions are described in Table 1.

Through the questions described above, it was possible to group the information according to what was asked in each one of them, allowing to identify the proposals related to RBT in software development and to identify the benefits and limitations, as well as metrics that allow evaluating this type of testing. Likewise, through the state of art, we can identify the solutions, existing deficiencies, and opportunities to propose new lines/future research work.

Table 1. *Research Questions*

N°	Research question	Motivation
1	**Q1.** What is meant by risk-based testing in the scientific community?	To know the definition of risk-based testing according to review papers.
2	**Q2.** What studies on risk-based testing exist?	To determine the number of publications since 2000 to April 2020, regarding risk-based testing for the software industry.
3	**Q3.** What metrics have been proposed for risk-based testing?	To determine the metrics that were used and the context in which they were applied.
4	**Q4.** What are the benefits and limitations that were presented in the proposals for risk-based testing?	To determine what are the benefits of creating proposals and the limitations that have been presented for their implementation.

3.1.2. Search Strategy

To carry out the automated information search, the following databases were used: Scopus Springer, IEEE Xplore, Redalyc and Google Scholar, performing a combination with the logical "AND" connector on the identified keywords: software, testing, RBT, since this is a specific type of software testing. Before the search chain in the scientific database application, the grey literature was consulted, which consisted of reports, companies' products, and services catalogs, documentation outside the indexed magazines, evidence that there is a great interest in this subject for this testing type. Therefore, the chain was made up of two parts, one related to software testing and the other to RBT. The basic search string that was adapted when running the review on the search engines is as follows: “Software testing" AND "Risk-based Testing”.

Since there are few relevant works in RBT processes, it has been chosen not to use the logical operator OR among the keywords because there would be non-coherent results related to the research topic and it is undesired to omit jobs that may be useful for our research. Furthermore, the research on the publications of the last two decades (since early 2000 to April 2020) was carried out and the studies found showed advances in this area. On the other hand, it is remarked that the subject is being investigated as part of a test process that helps companies identify risks in product development [(2)] and propose types of contributions for RBT, increasing interest in part of the scientific community.

3.1.3. Selection Criteria for Primary Studies

The title, abstract, and keywords of each study collected by the automated search will be evaluated to determine whether they are included among the potential studies that will be analyzed later. Consequently, only the studies that meet the following criteria will be considered: (i) English language papers referring to RBT and (ii) papers published since 2000 to April 2020 in magazines, conferences, and books. As a factor of exclusion, there was an exhaustive analysis of the abstracts, future works, and conclusions of each one of the studies. In some cases, (where there was no clarity with the aforementioned) it was necessary to extend to a more detailed reading in other sections of the study.

With the analysis of the documents, measures of importance, and contributions of the subject, it was possible to do the selection for the primary studies. On the other hand, those studies that meet some of the following exclusion criteria will be ignored: (i) duplicate studies (always considering the most complete and recent paper), (ii) studies whose main contribution is not related to RBT, (iii) studies that contemplate the

topics superficially. For this research, there were 3 evaluators: a principal who defined the objectives and research questions; two researchers with extensive experience in conducting systematic mapping, reviewing iteratively and incrementally review of each question and response that allowed the better organization of information to provide a quality document and understanding to the reader.

3.1.4. Quality Evaluation Criteria

To obtain the best results from selected studies, quality studies will be measured to determine which are the most important and relevant RBT. To this end, a questionnaire was made considering the research questions mentioned previously with a score of three values -1 (No), 0 (Partially), and +1 (Yes). The questionnaire consists of the evaluation criteria presented in Table 2 and Table 3 presents the definition of each of the evaluation criteria defined to evaluate the primary studies in detail.

The sum of the score of each criterion will conform to the final quality score about the study. The purpose of this quality assessment is not to exclude papers of low relevance, but to present to the reader the most representative and relevant studies considered in the development of this review. This is why some of the papers resulting in a relatively low score, such as [2], [21], [12], [23], [5], and [31] are not excluded because in our opinion, contribute to our investigation.

Table 2. *Evaluation Criteria*

N°	Evaluation Criteria
A	The study presents a clear definition of risk-based testing.
B	The study presents a detailed description of how to incorporate risk-based testing.
C	The study contains detailed steps on how to implement each of the proposals with risk-based testing.
D	The study exposes the results obtained after performing risk-based testing in a clear and detailed way.
E	The study has been cited by other authors.

3.1.5. Data extraction strategy

The data extraction strategy will be based on a series of possible answers for each of the research questions already defined. This allowed ensuring the application of the same data extraction criteria for all the selected works. Table 4 establishes each of the strategies that are evidenced in the defined research questions.

3.1.6. Selection of synthesis methods

For the data synthesis making, it was decided to use the information representation through tables, numbers, and/or percentage and/or study references selected and classified according to the possible ones for each of the research questions. The systematic mapping started in 2018 and ended in April 2020.

Table 3. *Evaluation criteria applied to primary studies*

Evaluation criteria	**Primary Study Reference**																							
	(7)	(8)	(2)	(21)	(9)	(22)	(12)	(14)	(23)	(24)	(25)	(26)	(13)	(27)	(28)	(4)	(29)	(10)	(5)	(30)	(31)	(32)	(33)
A	1	0	1	0	1	0	0	1	0	1	1	0	1	1	0	0	1	1	0	1	0	1	1
B	0	0	-1	0	0	0	-1	0	-1	0	0	0	0	1	1	0	0	1	1	1	1	1	1
C	0	0	-1	0	0	0	-1	0	-1	0	0	1	0	0	0	1	0	0	0	0	0	0	0
D	1	1	0	1	0	1	0	1	-1	1	1	0	1	0	0	0	1	0	0	0	0	1	1
E	1	1	1	-1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	0	0	0
Score	**3**	**2**	**0**	**0**	**2**	**2**	**-1**	**3**	**-2**	**3**	**3**	**2**	**3**	**3**	**1**	**2**	**3**	**3**	**1**	**3**	**1**	**3**	**3**

3.2. Implementation Stage

In the implementation stage, the application of the revision protocol defined in the previous stage was carried out. The number of iterations performed was two, one iteration for each established search source. Table 5 presents the total number of studies found, relevant studies, repeated ones, and primary studies found in the search sources: Scopus, Springer, IEE Xplore, Redalyc, and Google Scholar.

Table 4. *Classification scheme*

N°	Research question	Answers
1	What is meant by risk-based testing in the scientific community?	Review of definitions in related works.
2	What studies on risk-based testing exist?	The proposal, case study, surveys, experiments, systematic review, among others.
3	What metrics have been proposed for risk-based testing?	Metrics at the level of requirements, functional, architecture, development, security, progress, probability of failure. Metrics to evaluate risk-based testing.
4	What are the benefits and limitations that were presented in the proposals for risk-based testing?	Benefits in terms of cost, time, productivity, efficiency.

Table 5. *Count of studies found in each search source*

N°	Search sources	Studies found	Relevant studies	Relevant repeated studies	Selected Primary Studies
1	Scopus	58	17	0	16
2	Springer	85	11	7	1
3	IEEE Xplore	43	7	5	5
4	Redalyc	12	6	5	0
5	Google Scholar	26	2	2	1
	Total	**224**	**43**	**19**	**23**

4. Results

The results obtained for each of the research questions are shown below, as well as the systematic mapping in general.

4.1. What is meant by risk-based testing in the scientific community?

In the systematic review of the 23 papers that were studied, it can be noticed that only 66.7% of the analyzed studies use a common or unique definition for the term “risk-based tests”. The definitions replacing the term RBT according to the paper reference, quantity, and percentage of these studies are shown in Table 6.

4.2. What studies on risk-based testing exist?

In order to give a better organization to the articles found on RBT, there is a use of concepts that allow the best identification of each one of them and to classify them for better understanding. Several of these concepts were obtained from ontological definitions described in [34,35] and others from the same revised article. In Table 7, the description of each classification type submitted is detailed, according to definitions presented by authors in ontologies. Likewise, a detailed description of each classification term meaning is presented.

In the time window established and presented in Figure 1, since year 2000 onwards, there is an increasing interest in RBT, with research increasing from 2012 to the present. The percentage of studies according to the classification type per year is: (i) 22.7% corresponding to approaches: 2012 [30], 2017 [14], 2018 [22], 2020 [32,33]; (ii) 18.2% corresponds to case study: 2000 [21], 2010 [29], 2014 [2], 2016 [12]; (iii) 9.1% corresponds to Taxonomy: 2014 [9], 2019 [4]; (iv) 9.1% corresponds to Techniques: 2005 [26], 2018 [28]. (v) 40.9% corresponds to one article per year in Framework 2014 [27], Tools

2007 [31], Method 2016 [8], Methodology 2013 [25], Model 2012 [13], Prediction of Defects 2016 [23], Procedure 2014 [7], Process 2010 [10] and Exploratory Review 2016 [24]. Finally, in the years from 2001 to 2004, 2006, 2008, 2009, 2011, and 2015 no related studies are presented or there was no research and publication.

Table 8 shows the twenty types of studies found on RBT during systematic mapping. In addition, it can be seen that the approaches represent 23% of the studies, the techniques represent 9% of the studies, 18% corresponds to works where case studies were conducted, 9% corresponds to taxonomies and 45% corresponds to a study that contains: a procedure, a method, prediction of defects, exploratory review, a model, a methodology, process, and framework.

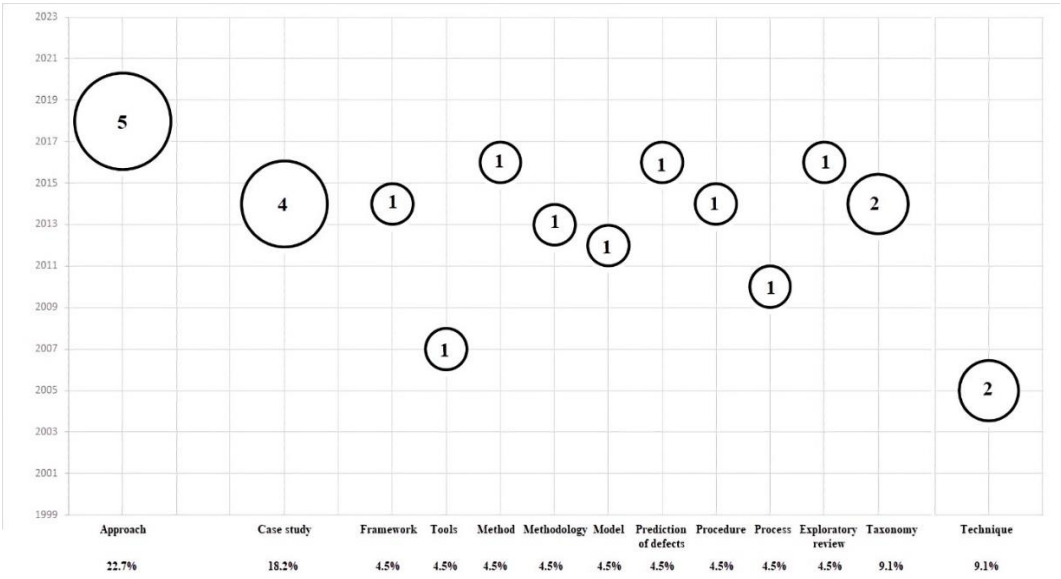

Figure 1. *Publications per year*

Table 6. *Definition of risk-based testing*

N°	Definition	Reference Papers	#	%
1	Test-based approach for risk management.	(7)	1	7,1
2	It is a type of software test that considers the risks of the software product as the guiding factor for solving decision problems in the design, selection, and prioritization of test cases.	(13,25, 27)	3	21,4
3	It is a type of software test that explicitly considers the risks of the software product as the guiding factor for solving decision problems at all stages of the testing process, that is, the planning, design, implementation, and evaluation of the test.	(2, 24)	2	14,2
4	It is a test approach that considers the risks of the software product as the guiding factor to support decisions at all stages of the testing process.	(9,12,14, 22, 23)	5	35,7
5	Addresses the explicit use of risk management activities within the test process.	(29)	1	7,1
6	It consists of activities for the identification, analysis, and mitigation of risk factors associated with software product requirements, giving priority to efforts and allocating resources for software components that need to be further tested.	(10)	1	7,1
7	It is an approach that consists of a set of activities related to the identification	(5)	1	7,1

of risk factors related to software requirements.

***Table 7**. Glossary of Concepts*

Ref. Primary study	Concept	Ontological definition	Ref. concept
(14,22,30)	Approach	It is a research method, a way of thinking, which emphasizes the total system instead of component subsystems. It strives to optimize the effectiveness of the total system instead of improving the effectiveness of closed systems.	(22)
(2,12,21,29)	Case study	It is an empirical investigation that studies a contemporary phenomenon in its real context, where the boundaries between the phenomenon and the context are not accurately shown, and in which multiple sources of evidence are used.	(36)
(27)	Framework	The software structure is composed of customizable and interchangeable components for the development of a tool.	(37)
(31)	Tools	The tools automatize the implementation of certain activities.	(38)
(8)	Method	A method is a procedure that is generally oriented towards a specific purpose.	(34)
(25)	Methodology	The methodology is transformed into a discipline that studies, analyses, promotes, and cleanses the method.	(25)
(13)	Quality model	Set of measurable concepts and the relationships between them that provide the basis for specifying the quality requirements and assessing the quality of the entities of a given entity class.	(39)
(23)	Defect Prediction	These models are useful tools for testing software.	(23)
(7)	Procedure	Specified way to carry out an activity or process (ISO 9000).	(40)
(10)	Process	A consistent set of policies, organizational structures, technologies; procedures, purposes, objectives, and work products necessary to design, develop, implement, and maintain a software product.	(40)
(24)	Exploratory review	The process by which a text is analyzed in order to identify its grammatical structure, based on a formal grammar.	(24)
(4,9)	Taxonomy	It is a type of controlled vocabulary in which all terms are connected by some structural model (hierarchical, arboreal, faceted, etc.) and specially oriented to the navigation systems, organization, and search of website content.	(35)
(26,28)	Technique	Different ways of applying a method.	(34)

4.3. What metrics have been proposed for risk-based testing?

Table 9 describes the proposed metrics in general: [5], [7], [8], [21], and [13]. In that sense, not all 23 primary papers analyzed propose metrics, only 23.8% of the proposals do it (5 papers). Table 10 shows the metrics for assessing the RBT process and Table 11 shows the Metrics for assessing RBT activities.

4.4. What are the benefits and limitations that were presented in the proposals for risk-based testing?

In the literature reviewed, it is observed that RBT for software development companies have some benefits such as: (i) it helps to make the quality of the deliverables more reliable; (ii) optimization in the testing process; (iii) quality in the product release; (iv) variables improvement such as confidence and profitability of the organizations; (v) it helps to detect the most critical defects from the beginning; (vi) cost and time reduction; (vii) compliance with the production deadline is reduced; (viii) identification of the software parts that are most likely to fail; (ix) it helps test managers to make better use of their limited time and resources; and (x) the use of fuzzy expert system facilitates more realistic risk estimates. In addition, the following challenges were identified: (i) time in risk assessment when

systems are complex; (ii) availability of experts during the risk estimation process; and (iii) use of case study with controllable scope to evaluate proposals submitted.

4.5. Result of systematic mapping

Once analyzed each of the questions in the systematic mapping, the following are identified:

(i) In the definitions of the term "risk-based testing", it can be identified that: in general, product risks are a guiding factor to support the entire testing process during the development life cycle; (ii) Most of the proposals studied, perform literature reviews or systematic mapping on RBT to be able to define some type of solution for software development organizations. Likewise, some of the authors carry out a case study to demonstrate the importance, benefits, and contributions to the software industry, which shows a great interest in this type of evidence; (iii) For case studies they consider literature review or systematic mapping to be able to evaluate in companies the use of RBT without making any detailed proposals; (iv) In [9] the taxonomy is not made real case studies to apply this technique and it is not manifested as future work. It is not clear what the necessary steps are or where to start to contribute in each of the papers proposed to look at the context; (v) In [12] it isn't clear what the variables in it were to consider to analyze and buy the results of the open interviews and the documents delivered by the SME companies; and (vi) In [23] there is not an example of experience with case studies in industrial projects, only empirical studies that provide evidence of defects to support software testing activities.

5. Discussion

5.1. Main Observations

The systematic mapping goal is to know the current proposals or initiatives about RBT. Once the studies found have been analyzed, the following is observed: (i) Very few studies are evidenced in relation to RBT, making a systematic mapping since 2000 to the present to be able to identify the importance of the topic in the scientific field, demonstrating that it is a line of research that is still being investigated; (ii) Due to the proposals presented in some papers, the authors have made efforts to carry out case studies in order to demonstrate the benefits that an organization can have when applying RBT in small and large organizations; (iii) The metrics proposed in some studies help define software estimation, identifying possible risks at the level of architecture, functionality, requirements, development, and security.

In this case, the metrics were used according to the need of the study or proposal to be developed. However, there are metrics to evaluate RBT that help identify the quality of this in product development, increasing its delivery quality; (iv) Some proposals propose solutions to be applied to the software industry including elements such as phases or activities, but not all define roles, input and output artifacts.

Table 8. *Classification of the types of proposals in risk-based tests according to ontological concepts.*

Type of author classification	Description of the proposal	Ref.	%
Approach	Through the quality assessment based on the QuaMoco [15] quality model (Quality Modelling and Control), two approaches are made to be integrated into RBT: quality assessment of each component and direct use of RBT of the metrics at the lowest level of the quality model hierarchy.	(22)	22.72
	A light approach is proposed for the estimation of risk probabilities in risk-based software tests, promoting its implementation without specific prerequisites.	(14)	

	A PRISMA approach is proposed [41] that contemplates the creation of a product risk matrix.	(30)	
	Proposes an effective approach to managing risky components and proposes a general design of RBT.	(32)	
	Proposes a semi-automatic risk-based test case prioritization approach based on software modification information and method (function) invocation relationship.	(33)	
Case study	A case study is carried out with three industry organizations to provide improvements in the use of test risks for each one of them.	(2)	18.18
	The metrics method is used in case studies to predict failures while considering metrics that have already been established.	(21)	
	A case study is conducted through the conclusions of RBT in large companies obtained in the paper [2], the advantages for SMEs are identified through the related case study.	(12)	
	A case study is carried out through an RBT Process approach [10] to (i) check if RBT can find defects faster than a non-risk-based approach; (ii) check if the defects discovered are those that have high severity.	(29)	
Framework	A framework for the RBT process that configures and provides feedback for the risk assessment model.	(27)	4.50
Tools	Design and implementation of a risk assessment tool called QUART-ET (Rapid risk assessment for engineering tests) to facilitate the risk management process.	(31)	4.50
Method	Prioritization method of test cases based on RBT and prioritization of test cases using a fuzzy system [42].	(8)	4.50
Methodology	The generic test methodology is based on risk and a procedure on how it can be introduced into a test process.	(25)	4.50
Model	It presents a risk assessment model and a risk assessment procedure based on a generic risk test process.	(13)	4.50
Defect Prediction	Through the method of prediction [43] and RBT, a series of requirements are made to have a better prediction in tests.	(23)	4.50
Procedure	The procedure is defined from the review of different authors' proposals and incorporating the stages proposed by the ISTQB [44].	(7)	4.50
Process	Approach to build a software testing process model based on the risks of artifacts, guides, activities, and metrics, with the support of tools and evaluated by case studies.	(10)	4.50
Exploratory review	It explores how risk estimation is carried out in RBT approaches.	(24)	4.50
Taxonomy	It presents a taxonomy of RBT that provides a framework for understanding, categorizing, and evaluating.	(9)	9.10
	It presents a taxonomy of RBT to the current test standards among them; ISO/IEC/IEEE 29119 [45], ETSI EG [46], and OWASP [47].	(4)	
Technique	RBT techniques for application in test planning.	(26)	9.10
	Introduces an FMEA (Failure Mode Analysis and Effects) [48] risk-based technique with metrics for software testing.	(28)	

Table 9. *Metrics identified for risk-based testing*

Paper	Metric	Description
(7)	Functional point of view	It allows identifying the user's requirements and relevant derivative acceptance criteria to establish priorities in the tests.
	Architectural point of view	It allows identifying the components, shared libraries, and the implementation that are part of the architecture.
	Development point of view	It allows to identify the technological knowledge level, available tools support, or quality assurance measures.
(8)	Requirement Complexity (RC)	It allows to identify the requirements that need complex functionalities that tend to introduce more failures during implementation.
	Requirement Size (RS)	It allows to identify the size of the functions that could affect the number of failures in a system.

	Requirement Modification Status (RMS)	It allows to see the general modification status of each requirement. RMS represents the degree of modification of a requirement by comparing the same requirement with the previous version.
	Potential Security Threats (PST)	It allows to see the potential security threats (PST), it is used as an indicator of the security-related risks that reside in the requirements.
(21)	Metrics for Progress Monitoring	It allows to identify the number of planned tests, implemented and completed, the number of failures by function, the number of hours used in the tests for fault found, and the number of hours of use in the default setting (to correct the error and return the retest function).
	Metrics to predict the probability of failures	It allows to identify the change in functionality since the previous launch, the size of the function (that is, lines of code number), the complexity (this could be functional complexity or structural complexity), and the quality of the design documentation.
(13)	Automated metrics	It allows to define code complexity metrics.
	Semi-automated metrics	It allows to measure the functional complexity, for example.
	Manual metrics	It allows the frequency of use and the importance for the user.

Table 10. *Metrics identified to assess risk-based testing*

Paper	Metric	Description
(5)	For risk tests case	It allows to verify how many risks have been mitigated. It allows to verify how many risks have been mitigated by requirement.
	Identify prioritized risks	It allows to verify the prioritized risks with the highest level of requirements.
	Identify risk category	It allows to verify the risk classification according to categories or taxonomies.
	Identify treated risks	It allows to verify how much the risks, decrease in each iteration or test cycle.
	Verify risk reduction	It allows to verify the risk reduction leverage.
	Effort identification	It allows supporting planning by providing effort estimates.
	Defect identification	Indicates the quality of the RBT.
	Identify the effectiveness of RBT	Effectiveness of the RBT.
	Identify unidentified defects	Defect unnoticed with RBT.
	Effort required	The effort required to find a defect in RBT cases.

Table 11*. Metrics identified to assess risk-based testing activities*

Paper	Metric	Description
(5)	To time identification	It allows to know the average time taken to analyze a requirement with a certain number of lines.
	To identify the productivity of RBT	It allows to verify how productive are risk identification meetings.
	To identify the productivity of RBT (low factor)	It allows to identify the number of low-risk factors identified.
	To identify the same risk exposure	It allows identification of the same risk exposure.
	To assess risk identification activity	It allows assessing the useful identified risks / meaningful to design test cases.

However, in RTBProcess, it is denoted that although there are roles and activities, the inputs and outputs of the artifacts presented in the model are not clear; and (v) Some authors research on a case study to incorporate their solution in the software industry.

In this way, to determine the benefits, advantages, and limitations found in relation to obtaining data and information during its implementation. Nonetheless, the necessary tools they used to collect the information when applying the types of proposals in those studies are not displayed.

5.2. Limitations of systematic mapping

When search string was made, it was necessary to use the words “risk-based testing” only as these yielded strong results in engines such as

Springer, IEEE Xplore Digital Library, Redalyc, and Google Scholar. Likewise, the search for papers was carried out until April 2020 to extend this paper.

5.3. Transcendence for Research and Practice

This systematic mapping is of great importance for IT personnel who want to incorporate RBT in development projects, allowing the identification of product risks, carrying out test cases, and assuring product quality. For researchers who wish to continue with this line of research, it is an area that has a greater interest in terms of product quality. In addition, there are several works in the future posed by the authors of the papers to continue working on this topic. Organizations will benefit from using this type of evidence and see that there are more initiatives or proposals that help incorporate RBT evidence.

6. A framework to support risk-based testing in the development of global software

From the systematic mapping, we have identified a set of fundamental process elements such as roles, products, activities, and tools that allow identifying the contribution of each proposal, considering the software product risks and the general testing phases defined in the ISO 29119. In review and categorization of these elements, a first phase of the proposed framework development will be carried out that belongs to a process of RBT development for global software development teams, following the 3C (Communication, Coordination, and Cooperation) collaboration model [49], as this is a development approach that is currently used.

Furthermore, this process will help companies to incorporate product risks and test cases in an

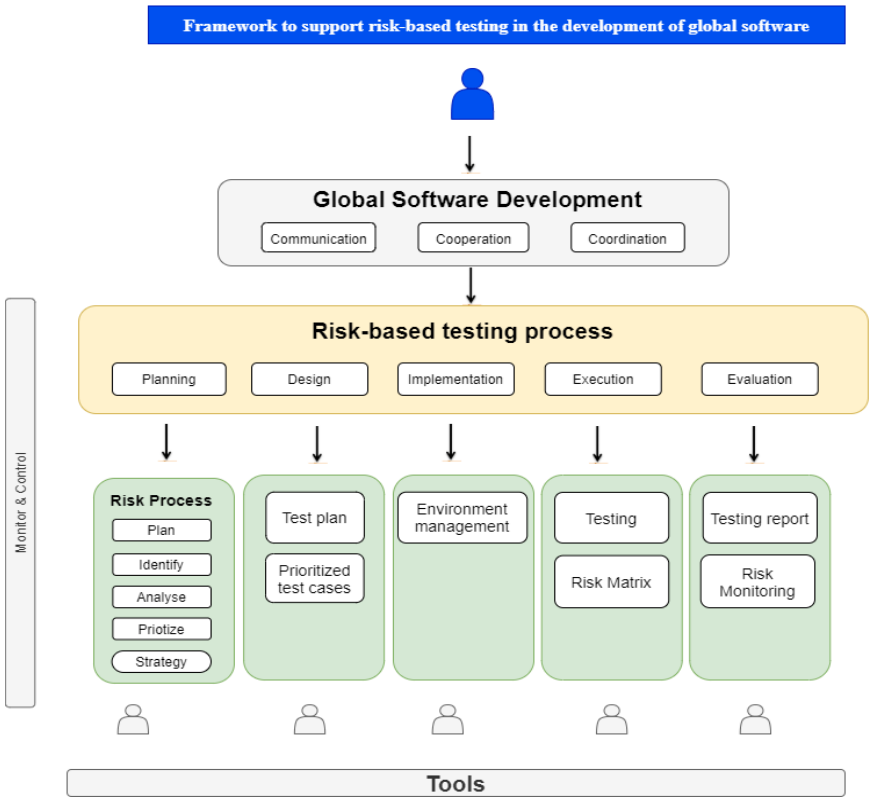

Figure 2. *Preview of the proposed process*

agile and efficient way. The framework consists of the following elements:

(i) Practices of Global software development at the level of communication, coordination, and cooperation; (ii) Risk-based testing process for global software development that includes the phases: planning, design, implementation, execution, and evaluation. In addition, establish a set of roles and input and output artifacts that allow monitoring and control over the software tests that are generated. Likewise, within the risk-based testing process, in the planning section is Product Risk Management, which includes the identification, analysis, prioritization and strategy of risks in the development of the software product; and (iii) Software tools that include a set of guides or techniques that help execute the activities of the testing process model and tests to carry out a series of activities and make the documentation that helps to use this proposed process. Figure 2 shows a preview of the framework composed of the following elements: risk process model, test process model, and global development of software and tools.

7. Conclusions and Future Work

In recent years, RBT begins to be an interesting topic for organizations as it is a type of software test that involves product risks as an essential part of the product's life cycle. Although this is a relatively emergent issue, there are proposals that provide a solution to perform this type of testing in the software industry. However, it is not clear how to incorporate this type of evidence in organizations, there are specific roles, tasks, and activities, the inputs and outputs artifacts that allow it to be implemented or incorporated are not clear and does not specify the type of organization to which it is executed in the case study.

The results obtained in this systematic mapping demonstrate the importance of RBT and the benefits that organizations can have when using the proposals proposed by the authors. On the other hand, other proposals incorporate test case prioritization techniques, which provide benefits in terms of adjusting your test efforts by time, cost, and budget. In addition, the use of metrics helps to identify the risks that may arise during the architecture, analysis, and development phase of the software cycle, thus achieving, listing risks to be evaluated, performing test cases, executing and continuing to monitor. Considering the proposals found in the systematic review of the literature and considering the deficiencies found, we have presented a detailed summary of our research proposal, which defines a framework to support RBT in the development of global software.

8. Acknowledgments

The professors Ph.D. César Pardo and MSc. Carlos Ardila gratefully acknowledges the contribution of the University of Cauca, where they work as Associate Professor and Full Professor, respectively.

9. Funding Statement

The author(s) received no specific funding for this work.

10. References

(1) Mera-Paz J. Análisis del proceso de pruebas de calidad de software. Ingeniería Solidaria. 2016;12(20):163–76.https://doi.org/10.16925/in.v12i20.1482.

(2) Felderer M, Ramler R. A multiple case study on risk-based testing in industry. International Journal on Software Tools

for Technology Transfer. 2014;16(5):609–25. https://doi.org/10.1007/s10009-014-0328-z.

(3) Gerrard P, Thompson N. Risk-Based e-Business Testing. 1st ed. Norwood (USA): Artech House Publishers; 2002. 368 p.

(4) Großmann J, Felderer M, Viehmann J, Schieferdecker IK. A Taxonomy to Assess and Tailor Risk-Based Testing in Recent Testing Standards. IEEE Software. 2019;37(1):40-49. https://doi.org/10.1109/MS.2019.2915297.

5) Souza E, Gusmao C, Alves K, Venancio J, Melo R. Measurement and Control for Risk-based Test Cases and Activities. In: 10th Latin American Test Workshop - LATW. Rio de Janeiro, Brasil: IEEE Institute of Electrical and Electronics Engineers; 2009. p. 210–5.

(6) Schieferdecker IK. Model-Based Testing. IEEE Software. 2012;29(1):14–8. https://doi.org/10.1109/MS.2012.13.

(7) Felderer M, Ramler R. Integrating risk-based testing in industrial test processes. Software Quality Journal. 2014;22:543–75. https://doi.org/10.1007/s11219-013-9226-y.

(8) Hettiarachchi C, Do H, Choi B. Risk-based test case prioritization using a fuzzy expert system. Information and Software Technology. 2016;69(C):1–15. https://doi.org/10.1016/j.infsof.2015.08.008.

(9) Felderer M, Schieferdecker IK. A taxonomy of risk-based testing. International Journal on Software Tools for Technology Transfer. 2014;16(5):559–68. https://doi.org/10.1007/s10009-014-0332-3.

(10) Souza E. RBTProcess: Modelo de Processo de Teste de Software baseado em Riscos [master's thesis]. Recife, Brasil: Universidade de Pernambuco; 2008. 147 p.

(11) Bastidas MI, Calvache CJP, Ardila CA. Risk-Based Testing: Preliminary Findings Obtained From A Systematic Mapping Study of the Literature. In: XIV Jornadas Iberoamericanas de Ingeniería de Software e Ingeniería del Conocimiento (JIISIC'2019). Cañas, Costa Rica: Escuela Superior Politécnica de Chimborazo; 2019. p. 107–19.

(12) Felderer M, Ramler R. Risk orientation in software testing processes of small and medium enterprises: an exploratory and comparative study. Software Quality Journal. 2016;24(3):519–48. https://doi.org/10.1007/s11219-015-9289-z.

(13) Felderer M, Haisjackl C, Breu R, Motz J. Integrating Manual and Automatic Risk Assessment for Risk-Based Testing. In: Biffl S, Winkler D, Bergsmann J, editors. 4th International Conference, SWQD 2012 LNBIP 94. Berlin, Heidelberg: Springer Cham; 2012. p. 159–80.

(14) Ramler R, Felderer M, Leitner M. A Lightweight Approach for Estimating Probability in Risk-Based Software Testing. Großmann J, Felderer M, Seehusen F, editors. 4th International Workshop, RISK 2016 LNCS 10224. Graz, Austria: Springer, Cham; 2017.

(15) Wagner S, Lochmann K, Heinemann L, Kläs M, Trendowicz A, Plösch R, et al. The quamoco product quality modeling and assessment approach. In: ICSE '12: Proceedings of the 34th International Conference on Software Engineering. Zurich, Switzerland: IEEE Press; 2012. p. 1133–1142.

(16) Patón-Romero JD, Piattini M. Modelos de madurez de Green IT: un mapeo sistemático. International Journal of Information Systems and Software Engineering for Big Companies (IJISEBC). 2017;4(2):53–61.

(17) Genero M, Cruz-Lemus J, Piattini M. Métodos de investigación en ingeniería del software. 1st ed. España: RA-MA; 2014.

(18) Kitchenham B, Charters S. Guidelines for performing Systematic Literature Reviews in Software Engineering - Technical Report EBSE. UK: Keele University and Durham University; 2007.

(19) Petersen K, Feldt R, Mujtaba S, Mattsson M. Systematic mapping studies in software engineering. In: EASE'08 Proceedings of the 12th international conference on Evaluation and Assessment in Software Engineering. Swindon (UK): BCS Learning & Development Ltd; 2008. p. 68–77.

(20) Budgen D, Turner M, Brereton P, Kitchenham B. Using mapping studies in software engineering. In: Proceedings of PPIG 2008. Lancaster: Lancaster University; 2008. p. 195–204.

(21) Amland S. Risk-based testing: Risk analysis fundamentals and metrics for software testing including a financial application case study. Journal of Systems and Software. 2000;53(3):287–95. https://doi.org/10.1016/S0164-1212(00)00019-4.

(22) Foidl H, Felderer M. Integrating software quality models into risk-based testing. Software Quality Journal. 2018;26(2):809–47. https://doi.org/10.1007/s11219-016-9345-3.

(23) Ramler R, Felderer M. Requirements for Integrating Defect Prediction and Risk-Based Testing. In: 42nd EUROMICRO Conference on Software Engineering and Advanced Applications. Limassol: IEEE Press; 2016. p. 359–62.

(24) Felderer M, Haisjackl C, Pekar V, Breu R. An Exploratory Study on Risk Estimation in Risk-Based Testing Approaches. In: Winkler D, Biffl S, Bergsmann J, editors. 7th International Conference, SWQD 2015 LNBIP 200. Berlin, Heidelberg: Springer, Cham; 2015. p. 32–43.

(25) Felderer M, Ramler R. Experiences and Challenges of Introducing Risk-Based Testing in an Industrial Project. In: Winkler D, Biffl S, Bergsmann J, editors. 5th International Conference, SWQD 2013 LNBIP 133. Vienna, Austria: Springer, Cham; 2013. p. 10–29.

(26) Redmill F. Theory and practice of risk-based testing. Software Testing Verification & Reliability. 2005;15(1):3–20.

(27) Felderer M, Haisjackl C, Pekar V, Breu R. A Risk Assessment Framework for Software Testing. In: Margaria T, Steffen B, editors. 6th International Symposium,

ISoLA 2014 LNCS 8803 - Part II. Corfu, Greece: Springer, Cham; 2014. p. 292–308.

(28) Chamarthi R, Reddy A. Empirical Methodology of Testing Using FMEA and Quality Metrics. In: 2018 International Conference on Inventive Research in Computing Applications (ICIRCA 2018). Coimbatore, India: IEEE Press; 2018. p. 85–90.

(29) Souza EP, Gusmao C. Risk-Based Testing: A Case Study. In: 7th International Conference of Information Technology: New Generations (ITNG). Las Vegas, NV: IEEE Press; 2010. p. 1032–7.

(30) Veenendaal E van. The PRISMA Approach: Practical Risk-Based Testing. 1st ed. UTN Publishers; 2012. 136 p.

(31) Sherrell L, Bowen S, Puppala H. A Tool for Risk-Based Testing. Tennessee: University of Memphis; 2007.

(32) Dahiya O, Solanki K, Dhankhar A. Risk-Based Testing: Identifying, Assessing, Mitigating & Managing Risks Efficiently in Software Testing. Int J Adv Res Eng Technol. 2020;11(3):192–203.

(33) Jahan H, Feng Z, Hasan-Mahmud S. Risk-Based Test Case Prioritization by Correlating System Methods and Their Associated Risks. To be published in Arab J Sci Eng [Preprint] 2020. https://doi.org/10.1007/s13369-020-04472-z.

(34) Pardo C, Pino FJ, García F, Piattini M, Baldassarre MT. An ontology for the harmonization of multiple standards and models. Computer Standards & Interfaces. 2012;34(1):48–59.

(35) Centelles M. Taxonomías para la categorización y la organización de la información en sitios web. Hipertext.net [Internet]. 2005;(3). Available from: https://www.upf.edu/hipertextnet/numero-3/taxonomias.html.

(36) Yin R. Case Study Research, Design and Methods, Applied social research Methods Series. 4th ed. Thousand Oaks, CA: SAGE Publications, Inc; 2009. 59–60 p.

(37) Pardo C, Pino F, García F, Baldassarre MT, Piattini M. From chaos to the systematic harmonization of multiple reference models: A harmonization framework applied in two case studies. Journal of Systems and Software. 2013;86(1):125–43. https://doi.org/10.1016/j.jss.2012.07.072.

(38) Cugola G, Ghezzi C. Software processes: a retrospective and a path to the future. Software Process: Improvement and Practice. 1998;4(3):101–23.

(39) García F, Bertoa MF, Calero C, Vallecillo A, Ruíz F, Piattini M, et al. Towards a consistent terminology for software measurement. Information & Software Technology. 2006;48(8):631–44.

(40) Fuggetta A. Software process: a roadmap. In: ICSE '00: Proceedings of the Conference on The Future of Software Engineering. New York: Association for Computing Machinery; 2000. p. 25–34.

(41) Burford BJ, Welch V, Waters E, Tugwell P, Moher D, O'Neill J, et al. Testing the PRISMA-Equity 2012 Reporting Guideline: the Perspectives of Systematic Review Authors. PLoS One.

2013;8(10):e75122. https://doi.org/10.1371/journal.pone.0075122.

(42) Grosan C, Abraham A. Intelligent Systems A Modern Approach. Berlin, Heidelberg; 2011. (Intelligent Systems Reference Library, Volume 17).

(43) Lessmann S, Baesens B, Mues C, Pietsch S. Benchmarking classification models for software defect prediction: A proposed framework and novel findings. IEEE Transactions on Software Engineering. 2008;34(4):485–96. https://doi.org/10.1109/TSE.2008.35.

(44) Graham D, Veenendaal van E, Evans I, Black R. Foundations of software testing ISTQB Certification. Revised Ed. London: Cengage Learning EMEA; 2008. 258 p.

(45) International Standard Organization (ISO). ISO/IEC/IEEE DIS 29119-2 Software and systems engineering. Software testing .Part 2: Test processes [Internet]. ISO - International Organization for Standardization. 2013. 50 p. [Consulted November 20 2019] Available from: https://www.iso.org/standard/79428.html.

(46) European Telecommunications Standards Institute (ETSI). Methods for Testing & Specification MTS - Risk-based Security Assessment and Testing Methodologies. Francia; 2016.

(47) Muller A, Meucci M. The owasp Testing Guide 4.0. Maryland (USA): The OWASP Foundation; 2014. 224 p.

(48) Šolc M. Applying of Method FMEA (Failure Mode and Effects Analysis) in the Logistics Process. In: ARSA - 5th Advanced Research in Scientific Areas. EDIS - Publishing Institution of the University of Zilina; 2012. p. 1906–11.

(49) Steinmacher I, Chaves AP, Gerosa MA. Awareness Support in Global Software Development: A Systematic Review Based on the 3C Collaboration Model. Comput Support Coop Work. 2013;22:113–58.