Vol. 3 No. 1 (2025)

Spectrum of Engineering Sciences

SPECTRUM OF ENGINEERING SCIENCES

Online ISSN

3007-3138

Print ISSN

3007-312X

Exploring The Role Of Exploratory Testing In Agile Software Development Environments

Majid Ali[1]
University of Management & Technology
Department of Software Engineering
raimajid2000@gmail.com

Nabeel Ali Khan[2]
University of Management & Technology
Department of Software Engineering
nabeel.ali@umt.edu.pk

Mudasira Sarfraz[3]
Superior University
Department of Computer Sciences
mudasira.sarfraz@superior.edu.pk

Sehrish Riaz[4]
University of Management & Technology
Department of Software Engineering
sehrish.riaz@umt.edu.pk

Tahir Mehmood[5]
University of Management & Technology
Department of Computer Sciences
tahir.mehmood.bahawalpur@gmail.com

Shamila Ghafoor[6]
University of Management & Technology
Department of Computer Sciences
shamila.ghafoor@umt.edu.pk

Abstract

Agile software development environments are characterized by their focus on exploratory testing as a principal methodology. The system's adaptation to requirements allows room for creativity in the treatment of areas that have so far not been taken up by formal tests only. In this way, the central question the study answers is How can I use and are these tests the right ones to implement regarding the context of agility? through a systematic review of the literature (SLR) which will be based on collecting the entire world-released research papers up to the current date. We expect to identify what is already known about these topics, different ways of their implementation, whether such activities are associated with any benefits, and what the problems are with them. To this end, various scholarly databases like IEEE Xplore, ACM digital library, SpringerLink in addition to others or even peer-reviewed sources have been explored. However, only those journal articles which meet the eligibility criteria set by the authors, i.e., they have the content capable of addressing the issues related to the topic under investigation, were considered valid materials for use in this study. These relevant papers were selected while the irrelevant ones or those that had insufficient validity were excluded. As a result, the total number of papers gathered was seventeen after a project whose primary purpose was to verify the selection standards were met 's end. However, it was noted from our analysis that lack management backing might be because few people know much about its advantages or face difficulties related with documentation being obsolete hence there is need for more empirical research if successful integration into scrum process framework stills remains a goal. This article summarizes agile software testing knowledge through a systematic review.

Information has been gathered from various sources in order to give fact based insights. These findings serve as a necessity for people doing exploratory testing as they will have to come up with new hypothesis or ideas while those that align test processes with expected project results after implementation get a chance for their wishes about working software to come true.

Keywords: Exploratory testing, Agile software development, Quality assurance, Software development methodologies

Introduction

In Agile software development everything moves quickly so testing has to keep up. Continuous integration means that there are many releases being done all the time which also impacts testing due to the fact that requirements change very often in this setup, therefore we need exploratory testing as one of our methodologies within these circumstances. Exploratory testing is different from traditional scripted testing because instead of following scripts it involves testers designing tests on-the-fly while executing them too; so they can bring in their domain knowledge, intuition or creativity when trying to discover bugs not easily captured by automated checks. However, despite its significance little research has been done on how best to employ exploratory testing under Agile environment conditions. The goal of this paper is to address this gap by conducting a systematic literature review on this subject matter. Exploratory testing is the research we aim to demonstrate about Agile environments with the help of an information collector who sincerely looks for as many things as possible. It also encompasses the open and transparent act of proving, discussing, and inventing faults (often until lately named as the testing culture and the failure of testing).

This study means to reflect deeply on the functionality of exploratory testing within Agile environments. The paper starts by mentioning that the research employs clear and well-organized methodology then it continues to compare research papers published at different times and places. Additionally, it discusses quality assessment criteria that are used to review works before addressing the next level of specific methods with which the tests can safely and effectively be done. Further, the paper highlights the advantages and verifiability of using exploratory research challenges while proposing improvements that can be made to the concept within Agile-associated methods. Finally, opportunities and challenges of exploratory testing are framed within the scope of these methods and certain professionals and academics are suggested to grab the future paper topics. By synthesizing knowledge from various sources, the SLR makes an important contribution to understanding software testing within agile contexts. It presents insights that are based on evidence and can be used for both practical activities as well as further studies hence demonstrating the importance of exploratory testing towards achieving objectives of agile software development.

Background

The table offers an introduction of 6 appropriate research documents concentrating on exploratory screening in software application growth. Each paper deals with various facets of exploratory screening, consisting of comprehending normal software program tester duties, recording exploratory screening methods, the duty of tester expertise in exploratory screening degrees of expedition in exploratory screening, as well as the harmony in between model-based plus exploratory screening. The

6th paper is an organized literary works evaluation (SLR) that discovers the function of exploratory screening in Agile software program growth settings, concentrating on its advantages as well as obstacles in placement with Agile concepts. The table consists of details on the paper title emphasis of the study magazine year, study strategy, high quality analysis research study structure, web content along with targeted electronic data sources.

Spectrum of Engineering Sciences
SPECTRUM OF ENGINEERING SCIENCES
Online ISSN
3007-3138
Print ISSN
3007-312X
SES

Sr.	References	Paper title	Focus of Survey	Publish Year	Survey Approach	Quality Assessment	Research Framework	Content	Targeted Digital Repositories
1	(Florea & Stray et al.,2023)	On the roles of software testers: An exploratory study	Understanding typical software tester roles	2023	Thematic analysis of interviews and job ads	No	No	Yes	ScienceDirect, Elsevier
2	(Bolton & M. et al.,2017)	An Exploratory Tester's Notebook	Documenting exploratory testing techniques	2017	Exploratory case study	No	No	Yes	Elsevier
3	(Itkonen &	The Role of the	How testers use	2012	Field Study	No	Yes	Yes	IEEE

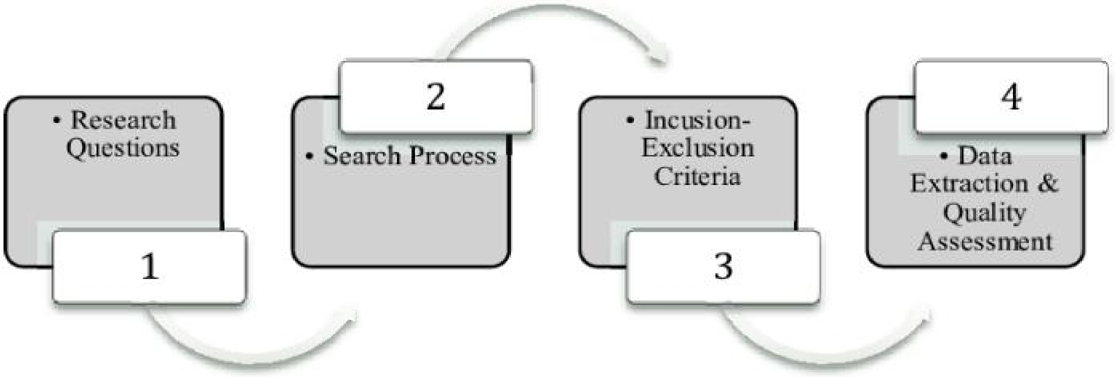
	Mäntylä et al.,2012)	Tester's Knowledge in Exploratory Software Testing	knowledge in exploratory software testing						
4	(Ghazi & Bjarnason et al.,2018)	Levels of Exploration in Exploratory Testing: From Freestyle to Fully Scripted	Degree of exploration in exploratory testing	2018	Focus Groups	Yes	No	Yes	IEEE
5	(Frajták & Bure	Model-Based	Combining model	2016	Experimental	No	Yes	Yes	IEEE

	š et al .,2016)	Testing and Exploratory Testing: Is Synergy Possible?	-based and exploratory testing						
6	(Ali & Khan et al.,2025)	Exploring the role of Exploratory testing in Agile software development environments.	This survey explores the role of exploratory testing in Agile development, focusing on its benefits and challe	2025	Systematic	Yes	Yes	Yes	Web of science

	nges in alignment with Agile principles.	

Methodology

A step-by-step literary evaluation (SLR) was chosen as the research study technique for this research. The purpose was to thoroughly evaluate the duty and also use expedition screening in Agile software application growth settings. This consisted of exploring its placement with Agile concepts its communication with referential stability strategies and also its effect on the screening procedure and also group characteristics. The technique recommended by Ali et alia. (2024) was adhered to make sure a methodical plus honest technique to info choice and also evaluation.


• Research Questions
1
2
• Search Process
• Incusion-Exclusion Criteria
3
4
• Data Extraction & Quality Assessment

Research Questions & Objectives

The first stage of this organized literary works evaluation (SLR) entailed specifying the research study concerns along with evaluating the existing research study landscape relating to the function of pilot screening in Agile software application growth settings. This SLR intended to deal with numerous crucial research study concerns each inspired by the demand to recognize the effect, advantages together with obstacles related to pilot screening within Agile software application growth. The research study concerns together with their matching inspirations are described in the table listed below.

Research Question	Objective	Motivation
1)What are the high-quality publication channels for " Exploring the role of Exploratory testing in Agile software development environments. " and which geographical areas have been targeted by research in this field over the years?	Identify high-quality publication channels and geographical areas targeted by research on "Exploring the role of Exploratory testing in Agile software development environments" to provide insights for researchers and practitioners.	Understanding the prominent publication channels and geographical focus of research on "Exploring the role of Exploratory testing in Agile software development environments" is crucial for researchers and practitioners.
2)How are the selected research papers on	The objective of this study is to analyze the distribution of	Analyzing the distribution of selected research papers on

Exploring the role of Exploratory testing in Agile software development environments. distributed by publication year?	selected research papers on "Exploring the role of Exploratory testing in Agile software development environments" by publication year.	"Exploring the role of Exploratory testing in Agile software development environments" by publication year provides valuable insights into the evolution and trends in this research domain.
3)What quality assessment parameters are used to evaluate the role of exploratory testing in Agile software development environments?	The objective of this study is to identify the quality assessment parameters used to evaluate the role of exploratory testing in Agile software development environments.	Understanding the quality assessment parameters used to evaluate the role of exploratory testing in Agile software development environments is essential for ensuring the reliability and validity of research findings in this field.
4)How does exploratory testing contribute to the effectiveness of Agile software development?	To assess the impact of exploratory testing on the overall effectiveness of Agile software development methodologies.	Agile methodologies emphasize flexibility and responsiveness. Understanding how exploratory testing fits into Agile practices is crucial for optimizing

		software development processes.
5)What are the key benefits of integrating exploratory testing into Agile development environments?	To identify and analyze the specific advantages of using exploratory testing within Agile software development contexts.	While exploratory testing is known to improve software quality, its specific benefits within Agile environments need to be clearly understood and quantified.
6)What challenges are associated with implementing exploratory testing in Agile software development teams?	To investigate the obstacles and difficulties that Agile development teams encounter when incorporating exploratory testing.	Understanding the challenges of integrating exploratory testing into Agile practices will help in developing strategies to mitigate these challenges effectively.

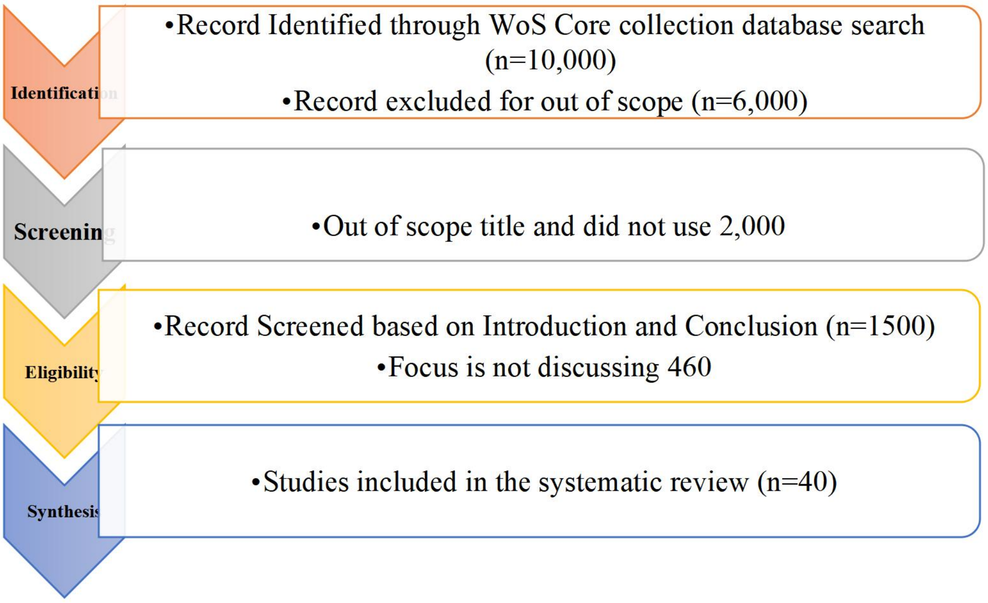
Search String

To carry out a complete look for pertinent literary works, numerous scholastic data sources were methodically inquired utilizing very carefully built search strings. Data sources such as Internet of Scientific research IEEE Xplore, ScienceDirect, and also others were consisted of in this search. The search strings were developed to catch posts together with documents especially attending to the function of exploratory screening in Agile software application growth settings. Key words such as "" exploratory screening,"" "" Agile software program advancement,"" "" software program screening methods,"" as well as associated terms were purposefully incorporated to make certain thorough insurance coverage of the literary works. Boolean drivers, term, as well as distance drivers were made use of as required to fine-tune the search results page and also make certain importance. By browsing throughout these trustworthy scholastic systems, this research study intended to consist of a varied variety of academic viewpoints plus understandings on the subject. The incorporation of numerous data sources boosts the durability along with efficiency of the literary works evaluation making it possible for a comprehensive evaluation of the existing body of understanding in this domain name.

Sources	Search String
Google Scholar, IEEE Xplore, Science Direct, MDPI, Springer Link, Web of Science ,Research gate	(Exploratory Testing OR Exploratory Tester) AND (Agile Software Development OR Agile Development)

Selection-based on Inclusion/Exclusion Criteria

Based upon the inclusion/exclusion standards detailed in Table 4 the choice procedure continued as adheres to. At first an overall of 10,000 documents were recovered from the picked databases. Out of these 6000 documents were left out as they were considered out of extent. An extra 2000 documents were left out because of titles that did not fulfill the incorporation requirements. The continuing to be 2000 documents went through testing based upon their intro and also verdict areas, causing 1,500 documents. After a detailed testimonial 460 documents were omitted as they did not concentrate on the subject of passion. Ultimately 40 research studies were consisted of in the organized evaluation based upon their importance to the research study inquiries as well as purposes.


Identificati
•Record Identified through WoS Core collection database search (n=10,000)
•Record excluded for out of scope (n=6,000)
Screenin
•Out of scope title and did not use 2,000
Eligibility
•Record Screened based on Introduction and Conclusion (n=1500)
•Focus is not discussing 460
Synthesis
•Studies included in the systematic review (n=40)

Assessment and Discussion of Research Questions

RQ#1 What are the high-quality publication channels for " Exploring the role of Exploratory testing in Agile software development environments." and which geographical areas have been targeted by research in this field over the years?

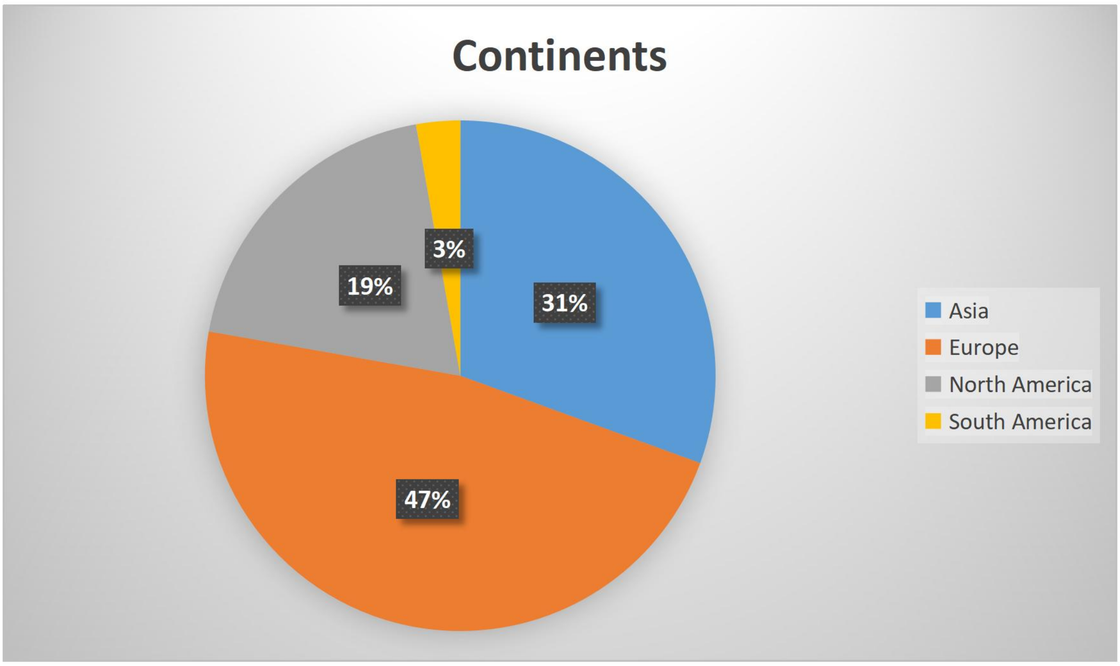
Sr No	Publication Source	No of Publications
1	Journal of Systems & Software	8
2	IEEE Transactions on Software Engineering	5
3	Springer	4
4	Institute of Computer Science, University of Tartu	3
5	International Journal of Software Engineering and Its Applications	3
6	Blekinge Institute of Technology	2

7	EasyChair	1
8	International Conference on Product-Focused Software Process Improvement	1
9	Pacific northwest software quality conference	1
10	SpringerLink	2
11	KTH Royal Institute of Technology	1
12	International Journal of Scientific and Research Publications	1
13	University of Houston – Clear Lake	1
14	Florida Institute of Technology	1
15	ieeexplore	3
16	IEEE	3
Total		40

Geographical Area

Sr no	Continent	Country	No of Publications
1	Asia	Malaysia	6
		India	1
		South Korea	1
		Indonesia	1
		China	2
2	Europe	Sweden	7
		Ireland	3
		Norway	3
		Finland	2
		Estonia	3
		Czech Republic	1

3	North America	USA	7
		Canada	2
4	South America	Brazil	1
Total			40



The geographical circulation of the documents consisted of in this organized literary works evaluation (SLR) is summed up as complies with: 11 documents were released in Asia, 17 in Europe, 7 in The United States and Canada plus 1 in South America. This circulation highlights the international rate of interest along with research study concentrate on the function of pilot screening in Agile software program growth settings throughout various continents. The fairly greater variety of magazines from Europe recommends a solid research study focus on this subject in the area, adhered to carefully by Asia. This circulation underscores the worldwide importance as well as relevance of pilot screening in the context of Agile software program growth.

Spectrum of Engineering Sciences

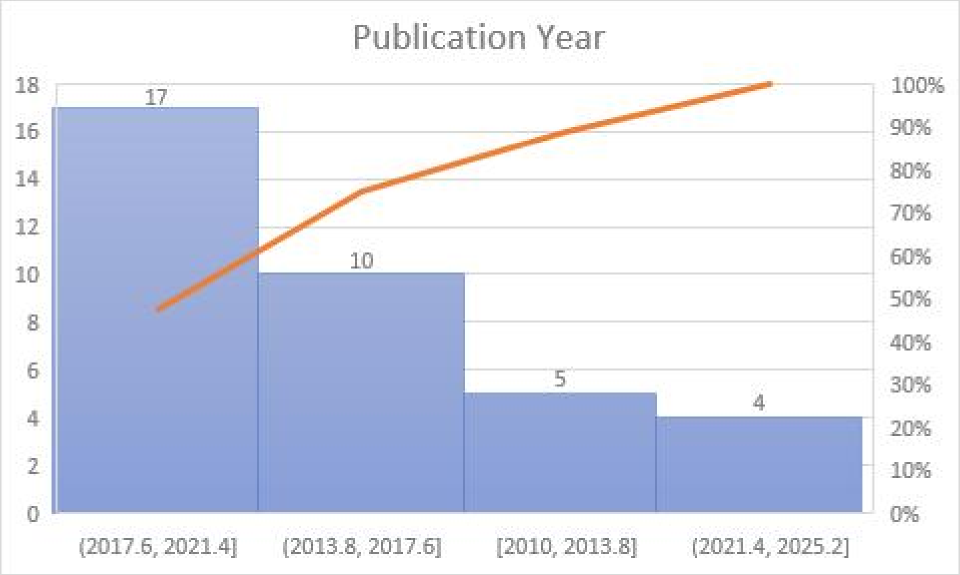
SPECTRUM OF ENGINEERING SCIENCES

Online ISSN 3007-3138

Print ISSN 3007-312X

RQ#2 How are the selected research papers on Exploring the role of Exploratory testing in Agile software development environments. distributed by publication year?

The circulation of documents released each year consisted of in this organized literary works testimonial (SLR) is as complies with: 5 documents were released in 2010 10 documents in 2014 17 documents in 2018 together with 8 documents in between 2022 plus 2024. This circulation mirrors the advancing passion plus research study concentrate on the duty of exploratory screening in Agile software program advancement atmospheres throughout the years with a remarkable rise in magazines observed in the last few years.



RQ#3 What quality assessment parameters are used to evaluate the role of exploratory testing in Agile software development environments?

The table provides a comprehensive summary of different studies and publications that have examined the quality assessment criteria used to assess the role of exploratory testing in Agile software development environments. It includes various types of investigations such as case studies, literature reviews, empirical researches and systematic literature reviews.

The methods applied vary from qualitative methods like thematic analysis, grounded theory and action research to mixed methods involving both quantitative techniques and qualitative approaches. These papers investigates the efficiency of the exploratory testing in the agile contexts. these papers also discuss the usability , challenges faced during implementation as well as its contribution towards the software construction process. In general this table compiles different scholarly works aimed at fostering insight into the significance of exploratory testing in agile through rigorous empirical and theoretical analysis.

Ref	P. Channel	Publication Year	Research Type	Empirical Type	Methodology
(Miroslav & Ahmed et al., 2018)	Journal of Systems & Software	2018	Case Study	Qualitative	Experimental
(Michael & Bolton et al., 2017)	IEEE Transactions on Software Engineering	2017	Exploratory	Qualitative	Not mentioned
(Florea & Stray et al., 2023)	Journal of Systems & Software	2023	Exploratory	Qualitative	Thematic Analysis
(Rekhi & Kalyan	Springer	2020	SLR	Qualitative	Systematic

et al., 2020)					Literature Review
(Joey & Cho et al., 2010)	IEEE	2010	Field Study	Mixed Method	Empirical Field Study
(Basri & Dominic et al., 2019)	IRICT 2018, AISC 843	2019	Case Study	Qualitative	Action Research
(Ali & Babar et al., 2019)	Proceedings of the joint working IEEE/IFIP Conference on software architecture; pp. 81-90	2019	Empirical	Qualitative	Experiment
(Beni & Suranto et al., 2015)	International Conference on Computer, Communications, and Control Technology (I4CT)	2015	Empirical	Mixed Method	Case Study
(Barraood & Mohd et	Knowledge Management	2021	Case Study	Qualitative	Grounded Theory

Spectrum of Engineering Sciences
SPECTRUM OF ENGINEERING SCIENCES
Online ISSN 3007-3138
Print ISSN 3007-312X

al.,2021)	International Conference (KMICe) 2021				
(Nawaz & Malik et al., 2018)	Department of Computer Science, School of Engineering, Blekinge Institute of Technology	2018	Literature Review	Qualitative	Thematic Analysis
(Pfahl & Huishi Yin et al., 2014)	ESEM'14	2014	Empirical	Qualitative	Survey
(Begel & Nagappan et al., 2017)	Journal of Systems & Software	2017	Empirical	Qualitative	Mixed Methods
(Kasowaki & Akara et al., 2023)	EasyChair	2023	SLR	Qualitative	Systematic Mapping Study
(Mårtensson & Bosch et al., 2021)	Journal of Systems and Software	2021	Literature Review	Qualitative	Narrative Review
(Martini	International	2021	Case	Qualitati	Case

& Bosch et al., 2021)	Conference on Product-Focused Software Process Improvement		Study	ve	Study Research
(James & Bach et al., 2013)	Springer	2013	Empirical	Mixed Method	Ethnography
(Tinkham & Kaner et al., 2013)	Proceedings of the Pacific northwest software quality conference	2013	Empirical	Qualitative	Action Research
(Mårtensson & Bosch et al., 2021)	SpringerLink	2021	Empirical	Qualitative	Design Science Research
(Pfahl & Münch et al., 2014)	Institute of Computer Science, University of Tartu	2014	Empirical	Qualitative	Survey
(Bhatti & Ghazi et	Blekinge Institute of	2015	Case Study	Mixed Method	Case Study

al., 2015)	Technology				Research
(Fredrik & Asplund et al., 2018)	KTH Royal Institute of Technology	2018	Literature Review	Quantitive	Systematic Review
(Hashimi et al., 2023)	AISC 843	2023	Case Study	Mixed Method	Grounded Theory
(Basri & Almomani et al., 2019)	IRICT 2018	2019	Empirical	Qualitative	Experiment
(Pfahl & Münch et al., 2014)	Institute of Computer Science, University of Tartu	2014	Empirical	Mixed Method	Survey
(Itkonen & Lasseniu s et al., 2012)	IEEE Transactions on Software Engineering	2012	Empirical	Qualitative	Mixed Methods
(Mohamed & Deraman et al., 2014)	International Journal of Software Engineering and Its Applications	2014	Literature Review	Qualitative	Narrative Review

(Shravan & Pargaonkar et al., 2023)	International Journal of Scientific and Research Publications	2023	Case Study	Mixed Method	Case Study Research
(Santos & Silva et al., 2011)	IEEE	2011	Empirical	Quantitative	Action Research
(Ghazi & Runeson et al., 2018)	IEEE Transactions on Software Engineering	2018	Empirical	Mixed Method	Design Science Research
(Karel Frajtak & ınekz et al., 2018)	IEEE Transactions on Software Engineering	2018	Literature Review	Mixed Method	Systematic Review
(Bhatti & Ghazi et al., 2021)	Elsevier	2021	Empirical	Qualitative	Survey
(Whiting & Lake et al., 2019)	University of Houston – Clear Lake	2019	Empirical	Qualitative	Mixed Methods
(Kaner & J.D et al., 2008)	Florida Institute of Technology	2008	SLR	Qualitative	Systematic Mapping Study

(Raappana & Saukkoriipi et al., 2016)	ieeexplore	2016	Literature Review	Qualitative	Thematic Analysis
(Hellmann & Maurer et al., 2011)	ieeexplore	2011	Case Study	Qualitative	Case Study Research
(Ghazi & Runeson et al., 2018)	IEEE Transactions on Software Engineering	2018	Empirical	Mixed Method	Ethnography
(Florea & Stray et al., 2023)	Journal of Systems & Software	2023	Exploratory	Qualitative	Thematic Analysis
(Rekhi & Kalyan et al., 2020)	Springer	2020	SLR	Qualitative	Systematic Literature Review
(Ali & Babar et al., 2019)	Proceedings of the joint working IEEE/IFIP Conference on software architecture;	2019	Empirical	Qualitative	Experiment

	pp. 81-90				
(Beni & Suranto et al., 2015)	International Conference on Computer, Communications, and Control Technology (I4CT)	2015	Empirical	Mixed Method	Case Study

RQ#4 How does exploratory testing contribute to the effectiveness of Agile software development?

The directive table here details the ad of value agile software development would experience when exploratory testing is utilized. This would allow testing methods to be responsive to changing requirements and priorities while giving immediate feedback for adjustments and enhancements. It equips testers with the ability to find defects that their creativity and knowledge in the domain being tested might not have identified in the past; this promotes collaboration among team members through communication sharing ideas and information. Moreover, with the help of exploratory work, the upgradation of projects that are participating in the digital transformation has become one of the key paths. The first way that comes to my mind is by writing about continuous improvement and through there one can get the necessary information to understand what exactly it is exploratory testing and if they are going to use it they should stop what they are doing and get on board, but, if not, they can continue to do what they are doing.

Flexibility: Flexibility is a part of the agile software development process, and this document gives its definition. This enables developers to create a test, the tests can be made on the fly while a new system is looked at in exploratory testing. This is a crucial thing because it helps with the processes that formal methods could not handle, since they might have omitted certain areas or aspects of the process (Miroslav & Ahmed et al., 2018).

Real-time feedback: Quick bug detection and subsequent fixing are the main ways of keeping running the short iterative cycles smooth. For this reason, we always have to keep an eye on the system performance at development due time because of pressure on the release frequency nicely as others (Michael & Bolton et al., 2017).

Creativity: Creativity requires the application of domain knowledge, intuition, and innovative thinking to elucidate the unseen and the boundaries of our knowledge (Florea & Stray et al., 2023).

Collaboration: Collaboration is the concept of a close association or dialogue amongst the team members in the process of testing to reach the goal of shared quality (Rekhi & Kalyan et al., 2020).

Continuous Improvement: Improving the continuous process is done through reflections of the past with a 'desire to make them better' created by the lessons learned at the beginning of the semester in the classroom (Joey & Cho et al., 2010).

Risk Mitigation: Vulnerability examination begins by early identification and subsequent mitigation of any issues that may emerge so that they don't get to a point where they become more serious problems or even threats (Basri & Dominic et al., 2019).

Customer Satisfaction: The satisfaction of the customer makes sure that the software delivered matches with the functionality, usability, and performance required by the user (Ali & Babar et al., 2019).

References	Aspect	Contribution of Exploratory Testing to Agile Development Effectiveness	Description	Example
(Miroslav & Ahmed et al., 2018)	Flexibility	Enables adaptability in testing approaches to accommodate changing requirements and priorities during iterative development cycles.	In Agile environments, where requirements evolve rapidly, exploratory testing allows testers to quickly adjust test strategies to address new features or changes.	Testers can modify test scenarios on-the-fly to explore newly implemented functionality during a sprint review.

(Michael & Bolton et al., 2017)	Real-time Feedback	Provides immediate feedback on software functionality, allowing for timely adjustments and improvements.	Unlike scripted testing, exploratory testing offers real-time insights into the software's behavior, helping developers identify and address issues promptly.	Testers can provide immediate feedback on user interface responsiveness during exploratory sessions, leading to usability improvements.
(Florea & Stray et al., 2023)	Creativity	Empowers testers to use domain knowledge, intuition, and creativity to uncover unforeseen issues and edge cases that may not be captured	Testers can explore the software from various user perspectives, leveraging their expertise to discover critical defects and	During exploratory testing, a tester may simulate real-world scenarios to uncover unusual behavior, such as entering

		by scripted tests.	usability issues.	unexpected data inputs.
(Rekhi & Kalyan et al., 2020)	Collaboration	Facilitates collaboration within Agile teams by promoting communication and knowledge sharing among team members during testing activities.	Exploratory testing encourages testers, developers, and stakeholders to collaborate closely, fostering a shared understanding of software quality goals and priorities.	Testers may collaborate with developers to investigate and resolve complex defects discovered during exploratory testing sessions.
(Joey & Cho et al., 2010)	Continuous Improvement	Supports continuous improvement by encouraging reflection on testing strategies and fostering	Through regular retrospectives and feedback loops, Agile teams can identify areas for	Testers may experiment with new exploratory testing techniques and share their findings

		a culture of learning and experimentation within the development process.	improvement in their testing practices and adapt their approaches accordingly.	with the team to enhance overall testing effectiveness.
(Basri & Dominic et al., 2019)	Risk Mitigation	Helps identify and mitigate risks early in the development lifecycle by uncovering defects and vulnerabilities before they escalate into larger issues.	After comprehensive study of software testers can uncover the potential risks and vulnerabilities that may not be seen through traditional approaches and allowing teams to address them effectively.	Testers may conduct exploratory testing sessions focused on security vulnerabilities to identify potential exploits and strengthen the software's defenses.
(Ali &	Customer	Enhances	Exploratory	By

Babar et al., 2019)	Satisfaction	customer satisfaction by ensuring that the delivered software meets or exceeds user expectations through thorough and adaptable testing.	testing teams can uncover issues that affect usability, such as speed bottlenecks and problems with quality that affect the user experience; thus this leads to increased client satisfaction levels.	incorporating user feedback into exploratory testing sessions, teams can prioritize improvements that directly align with user needs and preferences.

RQ#5 What are the key benefits of integrating exploratory testing into Agile development environments?

The table describes the principal advantages of exploratory testing in an agile environment. It stands out as a "shape shifter" that means the testers are at their disposal to transform the plan any time they realize the project requires new features or has different priorities. The method of Agile testing type is prompt feedback in the sense that bugs can be resolved instantly as the development process progresses. In addition, it promotes a higher test coverage by detecting defects and scenarios that couldn't be included in the Initial level of tests focused on certain use case only. Furthermore, this kind of trial implies the interaction of the team members, who happen more to work in collaboration with each other by this creating a common understanding of the group, thus it also has the desired effect of enhancing the team’s proper functioning. Equally, it carries the advantage that it can serve as an ongoing learning exercise through the sharing of experience so that what has been derived from the prior stage should be the guide for the next iterations. Moreover, the timely identification of the potential threats such as early recognition of risks would be a positive way to manage the life cycle of software products. Also, the customer satisfaction test should have real-world practices to make sure that software satisfies user expectations.

Adaptability: The most important skill of a great tester requires the ability to shift testing strategies and approaches as the project grows (Hashimi et al., 2023).

Real-time feedback: It is the ability to get immediate understanding of software operation that speedily reveals defects which permits them to get fixed (Basri & Almomani et al., 2019).

Enhanced test coverage: Such an engineering approach has been proved to bring about intensified testing of certain pieces of code and as a result some mistakes that were hard to be found by only predefined test cases will become visible shortly(Pfahl & Münch et al., 2014).

Collaboration: The primary focus is to actively perform the test in such a way that it will enable all the members to share the responsibilities alongside achieving the common objectives of quality thence (Itkonen & Lassenius et al., 2012).

Continuous improvement: This is about course correction that is done in the status quo in the hope that the future would look more positive than it did in the past (Mohamed & Deraman et al., 2014).

Early risk identification: Identifying possible threats or limitations when they are just beginning and before they turn into more severe problems which will, in turn, allow them to be settled proactively (Shravan & Pargaonkar et al., 2023).

Increased customer satisfaction: Making sure that the programs fulfilled all of the users' expectations in terms of functionality, ease of operation, and also speed (Santos & Silva et al., 2011).

References	Key Benefit	Description	Example
(Hashimi et al., 2023)	Adaptability	Allows testers to adjust their testing strategies in real-time to accommodate changes in requirements, priorities, and project direction during Agile cycles.	When new features are added mid-sprint, testers can quickly create and execute new test cases to validate them without waiting for updated test scripts.
(Basri & Almomani et al., 2019)	Real-time Feedback	Provides immediate insights into	Testers can report bugs as they are found

		software functionality, enabling rapid detection and resolution of defects.	during exploratory testing sessions, allowing developers to fix issues before the end of the sprint.
(Pfahl & Münch et al., 2014)	Enhanced Test Coverage	Uncovers defects and scenarios not covered by predefined test cases, ensuring a more comprehensive assessment of the software.	By exploring the application from various user perspectives, testers might discover usability issues or edge cases that scripted tests miss.
(Itkonen & Lassenius et al., 2012)	Collaboration	Promotes close cooperation and communication among team members, enhancing overall team dynamics and project understanding.	During the exploratory testing sessions, testers, developers, and product owners are able to join forces so that every aspect is checked and

			the most important problems solved.
(Mohamed & Deraman et al., 2014)	Continuous Improvement	Encourages ongoing refinement of testing practices based on lessons learned from exploratory testing.	Teams can hold regular retrospectives to discuss findings from exploratory testing and integrate improvements into their testing process.
(Shravan & Pargaonkar et al., 2023)	Early Risk Identification	Identifies potential risks and vulnerabilities early in the development process, allowing for proactive mitigation.	Exploratory testers might discover security vulnerabilities or performance bottlenecks early, enabling the team to address these issues before they affect production.
(Santos &	Increased	Ensures that the	Testers can

Silva et al., 2011)	Customer Satisfaction	software meets or exceeds user expectations by addressing real-world usage scenarios.	simulate realistic user behaviors and workflows to ensure the application is intuitive and performs well under various conditions, leading to a better user experience.

RQ#6 What challenges are associated with implementing exploratory testing in Agile software development teams?

The chart identifies a number of challenges faced when performing exploratory tests within an Agile environment. These include but are not limited to the fact that there is often not enough detailed documentation available such as making it difficult for test scenarios and results to be replicated; limited management support can lead resources being allocated towards automated rather than exploratory testing thus hampering its effectiveness; because they are unscripted in nature it is hard ensure consistent and comprehensive coverage with them while this method heavily relies on tester skills and experience if they have to be productive at all. Besides, due to time frames imposed by such processes, testers may end up hurrying through their work thereby missing out some bugs which would have been caught had more attention been paid. Compared to scripted ones measuring coverage as well as efficiency for example balancing it out with automation additionally knowledge sharing becomes an issue especially where team members do not share insights gained during these tests among themselves.

Documentation: Detailed records of test cases and their results were not available all the time. Therefore test scenarios could not be replicated and testing processes understood (Ghazi & Runeson et al., 2018).

Management's Limited Involvement: Sometimes top level managers did not fully support or comprehend what exploratory testing is about; this could obstruct its effective merging with other activities as well as allocation resources for it (Karel Frajtak & ınekz et al., 2018).

Test Coverage Inconsistency: Because there is no script involved in this kind of a test, each session might have different scope from another thus leaving some areas untested (Bhatti & Ghazi et al., 2021).

Skilled Based Activity: The success or failure of such an activity is pegged on the expertise, experience and familiarity with given subject matter possessed by individual who performs it (Whiting & Lake et al., 2019).

Time Bound: In Agile set ups where there are very strict timelines failure to do things right can lead one not doing them at all but just for the sake therefore thorough exploration becomes impossible due limited hours within which quality work has to be delivered before deadline catches up with scrum team members once again (Kaner & J.D et al., 2008).

Coverage Measurement Challenge: Although there are no specific rules for how much should be done during exploratory testing, some teams may end up doing too little while others do too much and still not find any bugs if these unexplored areas remain untouched since they are not automated (Raappana & Saukkoriipi; 2016).

Blending Automation Into Explorations: Attempting to balance manual work with automation can be challenging particularly in highly scripted environments (Hellmann & Maurer; 2011).

Knowledge Transfer: Knowledge gained from hands-on activities must not only be stored mentally but also shared among different team members either partially or collectively; failure to do this leads to repeating errors unnecessarily delaying potential areas of improvement from being identified besides failing to learn from previous mistakes (Ghazi & Runeson; 2018).

References	Challenge	Description	Example
(Ghazi & Runeson et al., 2018)	Lack of Documentation	Exploratory testing often lacks detailed documentation, making it difficult to reproduce test scenarios and results.	Testers might identify critical bugs, but without proper documentation, other team members may struggle to understand the test steps or recreate the issues.
(Karel Frajtak & ınekz et al.,	Limited Management Support	Insufficient understanding or buy-in from	Management may prioritize automated

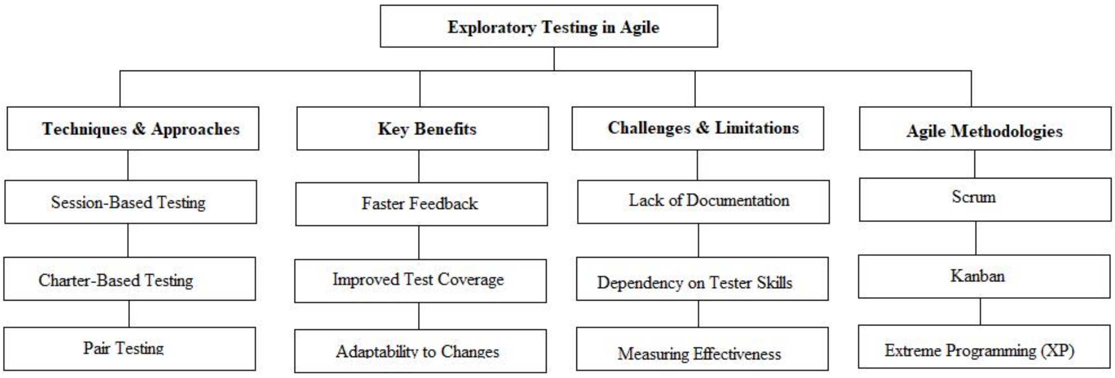
Spectrum of Engineering Sciences
Online ISSN 3007-3138
Print ISSN 3007-312X
SPECTRUM OF ENGINEERING SCIENCES

2018)		management can hinder the effective integration of exploratory testing into Agile processes.	testing and scripted tests over exploratory testing, limiting resources and time allocated for exploratory efforts.
(Bhatti & Ghazi et al., 2021)	Inconsistent Test Coverage	Since exploratory testing is unscripted, ensuring consistent and comprehensive coverage across different testing sessions can be challenging.	Different testers may focus on different areas of the application, leading to potential gaps in test coverage if not properly coordinated.
(Whiting & Lake et al., 2019)	Skill Dependency	The effectiveness of exploratory testing heavily relies on the tester's skills, experience, and domain knowledge.	Inexperienced testers might miss critical issues or not explore the application thoroughly, reducing the effectiveness of exploratory

			testing.
(Kaner & J.D et al., 2008)	Time Constraints	Agile environments often operate under tight schedules, leaving limited time for thorough exploratory testing.	Testers may have to rush their exploratory testing sessions to meet sprint deadlines, potentially overlooking important defects.
(Raappana & Saukkoriipi et al., 2016)	Difficulty in Measuring Coverage	Quantifying the coverage and effectiveness of exploratory testing can be more challenging compared to scripted testing.	Unlike scripted tests, there are no predefined metrics or pass/fail criteria to measure the extent of exploratory testing coverage.
(Hellmann & Maurer et al., 2011)	Integration with Automation	Balancing exploratory testing with automated testing can be difficult, especially in	Testers may struggle to find enough time for exploratory testing if most tests are automated and

		environments heavily reliant on automation.	must be maintained and run regularly.
(Ghazi & Runeson et al., 2018)	Knowledge Sharing	Capturing and sharing insights gained from exploratory testing can be problematic without proper mechanisms in place.	Information that is found in these types of tests may never reach anyone else on the team which means they will end up doing the same work twice or miss chances to improve it.

Taxonomy Diagram:



Conclusion and Future Directions

Conclusion

This comprehensive literature review has researched the part and consequences of exploratory testing in Agile software development settings. According to our research, exploratory testing is a very flexible and efficient method that conforms to Agile core values such as immediate feedback, increased test coverage and team member collaboration. The test aids early identification of risks while also greatly contributing to overall adaptive nature as well as quick response capability among different teams involved in this process. Nonetheless some challenges like little managerial help, absence of documentation coupled with heavy reliance on testers' skills point out specific weak areas which need more efforts directed towards them. Therefore we need to handle these challenges so that we can integrate it more effectively within our agile methodologies.

Future Work

The next step in the research should involve setting up parameters for recording and archiving the data so that it can be reused and to support open knowledge transfer in future exploratory experiments. Moreover, top management must know what role these kind of testing methods play in relation to the project goals of the organizations that apply agile development. There also should be more quantitative a studies which leads to a better understanding of the impact of conducting such tests on various projects' outcomes if it is done through this kind of approach. This will help us to recognize whether they are supposed to be executed simultaneously or separately. Also, there ought to be studies on this subject on automation tools that might contribute to the tasks besides the normal baby steps in exploration testing. We can actually pinpoint specific spots where humans will have to take a few steps back so robots can do the job faster due to the fact they operate smarter than they used to quite possibly. However, essential education to improve the knowledge and skills of the testers in the exploratory testing assure that the majority of the people will perform their tests more efficiently than the used one and consequently the customers will get the products they want on time.

References

1. Bures, M., Frajtak, K., & Ahmed, B. S. (2018). Tapir: Automation support of exploratory testing using model reconstruction of the system under test. *IEEE Transactions on Reliability*, *67*(2), 557-580.

2. Bolton, M. (2007, October). An Exploratory Tester's Notebook. In *SOFTWARE QUALITY CONFERENCE* (p. 117).

3. Florea, R., Stray, V., & Sjøberg, D. I. (2023). On the roles of software testers: An exploratory study. *Journal of Systems and Software*, *204*, 111742.

4. Chakrabarti, S., Nath, R., Banerji, P. K., Datta, S., Poddar, S., & Gangopadhyaya, M. (Eds.). (2022). *Interdisciplinary Research in Technology and Management*. CRC Press.

5. Cho, J. J. (2010). An exploratory study on issues and challenges of agile software development with scrum. *All Graduate theses and dissertations*, 599.

6. Basri, S., Dominic, D. D., Murugan, T., & Almomani, M. A. (2019). A proposed framework using exploratory testing to improve software quality in sme's. In *Recent Trends in Data Science and Soft Computing: Proceedings of the 3rd International*

Conference of Reliable Information and Communication Technology (IRICT 2018) (pp. 1113-1122). Springer International Publishing.

7. Babar, M. A. (2009, September). An exploratory study of architectural practices and challenges in using agile software development approaches. In *2009 Joint Working IEEE/IFIP Conference on Software Architecture & European Conference on Software Architecture* (pp. 81-90). IEEE.

8. Suranto, B. (2015, April). Exploratory software testing in agile project. In *2015 International Conference on Computer, Communications, and Control Technology (I4CT)* (pp. 280-283). IEEE.

9. Barraood, S. O., Mohd, H., & Baharom, F. (2021). A comparison study of software testing activities in agile methods. In *Knowledge Management International Conference (KMICe)* (Vol. 2021). Kedah, Malaysia: University Utara Malaysia.

10. Nawaz, A., & Malik, K. M. (2010). *Software testing process in agile development*. LAP Lambert Academic Publishing.

11. Pfahl, D., Yin, H., Mäntylä, M. V., & Münch, J. (2014, September). How is exploratory testing used? a state-of-the-practice survey. In *Proceedings of the 8th ACM/IEEE international symposium on empirical software engineering and measurement* (pp. 1-10).

12. Begel, A., & Nagappan, N. (2007, September). Usage and perceptions of agile software development in an industrial context: An exploratory study. In *First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007)* (pp. 255-264). IEEE.

13. Kasowaki, L., & Akara, N. (2023). *Exploratory Testing Strategies for Software Quality Assurance* (No. 11360). EasyChair.

14. Mårtensson, T., Ståhl, D., Martini, A., & Bosch, J. (2021). Efficient and effective exploratory testing of large-scale software systems. *Journal of Systems and Software*, *174*, 110890.

15. Mårtensson, T., Martini, A., Ståhl, D., & Bosch, J. (2019, November). Excellence in exploratory testing: Success factors in large-scale industry projects. In *International Conference on Product-Focused Software Process Improvement* (pp. 299-314). Cham: Springer International Publishing.

16. Bach, J. (2003). Exploratory testing explained. *Online: http://www. satisfice. com/articles/et-article. pdf*, 1-10.

17. Tinkham, A., & Kaner, C. (2003). Learning styles and exploratory testing. In *Proceedings of the Pacific northwest software quality conference*.

18. Mårtensson, T., Ståhl, D., & Bosch, J. (2017, October). Exploratory testing of large-scale systems–testing in the continuous integration and delivery pipeline. In *International Conference on Product-Focused Software Process Improvement* (pp. 368-384). Cham: Springer International Publishing.

19. Pfahl, D., Yin, H., Mäntylä, M. V., & Münch, J. (2014, September). How is exploratory testing used? a state-of-the-practice survey. In *Proceedings of the 8th ACM/IEEE international symposium on empirical software engineering and measurement* (pp. 1-10).

20. Afzal, W., Ghazi, A. N., Itkonen, J., Torkar, R., Andrews, A., & Bhatti, K. (2015). An experiment on the effectiveness and efficiency of exploratory testing. *Empirical Software Engineering*, *20*, 844-878.

21. Asplund, F. (2019). Exploratory testing: Do contextual factors influence software fault identification?. *Information and Software Technology*, *107*, 101-111.

22. Hashimi, H. A. A. (2023). *Usage and perceptions of spikes in agile software development: An exploratory study in an industrial context* (Doctoral dissertation, University of Southampton).

23. Basri, S., Dominic, D. D., Murugan, T., & Almomani, M. A. (2019). A proposed framework using exploratory testing to improve software quality in sme's. In *Recent Trends in Data Science and Soft Computing: Proceedings of the 3rd International Conference of Reliable Information and Communication Technology (IRICT 2018)* (pp. 1113-1122). Springer International Publishing.

24. Pfahl, D., Yin, H., Mäntylä, M. V., & Münch, J. (2014, September). How is exploratory testing used? a state-of-the-practice survey. In *Proceedings of the 8th ACM/IEEE international symposium on empirical software engineering and measurement* (pp. 1-10).

25. Itkonen, J., Mäntylä, M. V., & Lassenius, C. (2012). The role of the tester's knowledge in exploratory software testing. *IEEE Transactions on Software Engineering*, *39*(5), 707-724.

26. Almomani, M. A. T., Basri, S., Mahmood, A. K. B., & Bajeh, A. O. (2015, August). Software development practices and problems in Malaysian small and medium software enterprises: a pilot study. In *2015 5th international conference on IT convergence and security (ICITCS)* (pp. 1-5). IEEE.

27. Pargaonkar, S. (2023). A Study on the Benefits and Limitations of Software Testing Principles and Techniques: Software Quality Engineering.

28. dos Santos, A. M., Karlsson, B. F., Cavalcante, A. M., Correia, I. B., & Silva, E. (2011, March). Testing in an agile product development environment: An industry experience report. In *2011 12th Latin American Test Workshop (LATW)* (pp. 1-6). IEEE.

29. Ghazi, A. N., Petersen, K., Bjarnason, E., & Runeson, P. (2018). Levels of exploration in exploratory testing: From freestyle to fully scripted. *IEEE Access*, *6*, 26416-26423.

30. Frajtak, K., Bures, M., & Jelinek, I. (2016, September). Model-Based Testing and Exploratory Testing: Is Synergy Possible?. In *2016 6th International Conference on IT Convergence and Security (ICITCS)* (pp. 1-6). IEEE.

31. Bhatti, K., & Ghazi, A. N. (2010). Effectiveness of Exploratory Testing, An empirical scrutiny of the challenges and factors affecting the defect detection efficiency.

32. Whiting, E., & Datta, S. Performance Testing and Agile Software Development: A Systematic Review.

33. Kaner, C. (2008). A tutorial in exploratory testing. *QUEST*, *6*.

34. Raappana, P., Saukkoriipi, S., Tervonen, I., & Mäntylä, M. V. (2016, April). The Effect of Team Exploratory Testing--Experience Report from F-Secure. In *2016 IEEE Ninth International Conference on Software Testing, Verification and Validation Workshops (ICSTW)* (pp. 295-304). IEEE.

35. Hellmann, T. D., & Maurer, F. (2011, August). Rule-based exploratory testing of graphical user interfaces. In *2011 Agile Conference* (pp. 107-116). IEEE.

36. Ghazi, A. N., Petersen, K., Bjarnason, E., & Runeson, P. (2018). Levels of exploration in exploratory testing: From freestyle to fully scripted. *IEEE Access*, *6*, 26416-26423.

37. Mårtensson, T., Ståhl, D., Martini, A., & Bosch, J. (2021). Efficient and effective exploratory testing of large-scale software systems. *Journal of Systems and Software*, *174*, 110890.

38. Kasowaki, L., & Akara, N. (2023). *Exploratory Testing Strategies for Software Quality Assurance* (No. 11360). EasyChair.

39. Itkonen, J. (2011). Empirical studies on exploratory software testing.

40. Shoaib, M., Fatima, E., Sabahat, N., & Qamar, N. (2024). Exploring Agile Testing Methodologies: A Perspective from the Software Industry.