A Systematic Review of IoT Systems Testing: Objectives, Approaches, Tools, and Challenges

Jean Baptiste Minani, Fatima Sabir, Naouel Moha, and Yann-Gaël Guéhéneuc

Abstract—Internet of Things (IoT) systems are becoming prevalent in various domains, from healthcare to smart homes. Testing IoT systems is critical in ensuring their reliability. Previous papers studied separately the objectives, approaches, tools, and challenges of IoT systems testing. However, despite the rapid evolution of the IoT domain, no review has been undertaken to investigate all four aspects collectively. This paper presents a systematic literature review that aggregates, synthesizes, and discusses the results of 83 primary studies (PSs) concerning IoT testing objectives, approaches, tools, and challenges. We followed the Preferred Reporting Items for Systematic Review and Meta-Analysis (PRISMA) protocol to report our findings and answer research questions (RQs). To select PSs, we applied inclusion and exclusion criteria to relevant studies published between 2012 and 2022. We extracted and analyzed the data from PSs to understand IoT systems testing. The results reveal that IoT systems testing embraces traditional software quality attributes but also introduces new ones like connectivity, energy efficiency, device lifespan, distributivity, and dynamicity. They also show that existing IoT systems testing approaches are limited to specific aspects and should be expanded for more comprehensive testing. They also show 19 testing tools and 15 testbeds for testing IoT systems with their limitations, necessitating the development or enhancement for wider coverage. The large number of heterogeneous devices generating data in different formats, along with the need for testing in real-world scenarios, poses a challenge. Thus, our study offers insights into the testing objectives, approaches, tools, and challenges associated with IoT systems. Based on the results, we also provide practical guidance for IoT practitioners by cataloging existing tools and approaches, while also identifying new research opportunities for interested researchers.

Index Terms—IoT System Testing, IoT Testing Tools, IoT Testing Approaches, IoT Testing Challenges, IoT Quality Attributes

1 Introduction

The Internet of Things (IoT) continues to mature, with more devices and systems being deployed every day in different domains. Researchers predict that more than 75 billion IoT devices will be connected by 2025 [1], while Cisco expects 500 billion IoT devices to be connected by 2030 [2]. IoT systems have become key enablers in many domains such as smart cities, healthcare, logistics, retail, manufacturing, or agriculture [3], [4], [5]. IoT systems differ from traditional software systems in that they consist of many components (known as layers) [6], while traditional software systems focus on one component (i.e., application component). These components must work together seamlessly to enable physical devices to interact with the physical world, generating and processing data in real-time. In this paper, we use the term *IoT layer* as a synonym for *IoT component* or *IoT constituent* to minimize any potential confusion or ambiguity.

The multiple layers of IoT systems introduce many challenges related to scalability, real-time responsiveness, heterogeneity, and distributivity. Therefore, comprehensive testing of IoT systems before deployment is crucial to ensure their expected functionality. A single bug at any layer can cause a complete system failure. Thus, ensuring that IoT systems are tested is of paramount importance [7], particularly in safety-critical scenarios [8].

However, testing IoT systems is difficult [9], [10], [11], [12], [13], [14], and existing approaches and tools for traditional software testing may not suffice for IoT systems testing. A combination of different tools and approaches for testing IoT systems may be needed because of the characteristics of these systems [15].

This study reviews, summarizes, and synthesizes the current state of IoT systems testing, focusing on testing objectives, approaches, tools, and challenges. By delving into these aspects, we provide valuable insights that can inform the development of effective and specialized testing solutions to ensure the quality of these systems.

Although some studies have been conducted in recent years focusing on some aspects of IoT systems testing, we are not aware of any systematic literature review (SLR) that covers the objectives, approaches, tools, and challenges of testing IoT systems. Therefore, we conduct a SLR to investigate all four aspects collectively. Our key research questions (RQs) are:

- ❶ **RQ1**: What Are Testing Objectives Considered?
- ❷ **RQ2**: What Are Testing Approaches Investigated?
- ❸ **RQ3**: What Are Testing Tools Investigated?
- ❹ **RQ4**: What Are IoT Systems Used For Evaluation?
- ❺ **RQ5**: What Are Testing Challenges Identified?
- ❻ **RQ6**: How Are Testing Challenges Addressed?

- *Jean Baptiste Minani is with the Department of Computer Science and Software Engineering, Concordia University, Canada E-mail: jeanbaptiste.minani@concordia.ca*
- *Fatima Sabir is with the Department of Computer Science and Software Engineering, Concordia University, Canada, and Department of Computer Sciences, University of the Punjab, Lahore, Pakistan E-mail: fatima.sabir@pucit.edu.pk*
- *Naouel Moha is with the Department of Computer Science, École de Technologie Supérieure (ÉTS) – Université du Québec, Canada E-mail: moha.naouel@estmtl.ca*
- *Yann-Gaël Guéhéneuc is with the Department of Computer Science and Software Engineering, Concordia University, Canada E-mail: yann-gael.gueheneuc@concordia.ca*

We followed the updated PRISMA guideline for reporting systematic reviews [16]. We screened 8,294 potentially relevant studies from eight digital libraries published between 2012 and 2022. We applied inclusion and exclusion criteria and snowballed sampling. We assessed the quality of these studies from several perspectives including study design, conduct, analysis, conclusions, and implications. We retained 83 Primary Studies (PSs). We analyzed these PSs and reported the findings on testing objectives, approaches, tools, and challenges.

Testing has various objectives such as identifying bugs and errors, ensuring quality, and meeting the requirements of users. For this study, we focused on testing objectives pertaining to the quality attributes of IoT systems.

The main contributions of this study are:

❶ Compiling *47 quality attributes*, with *5 of them* specifically related to IoT systems, helping us understand the objectives of testing IoT systems;
❷ Providing overview of testing approaches including *4 levels of testing*, *10 types of testing*, and *15 testing techniques and test practices* for IoT systems;
❸ Compiling *19 user testing tools* and *15 testbeds* for IoT systems testing, highlighting their usage at different stages of IoT systems development;
❹ Summarizing the *challenges encountered in IoT systems testing*, while highlighting *potential research directions* to effectively address the emerging and futuristic challenges associated with testing IoT systems and proposed solutions to address these challenges;
❺ Compiling *IoT systems used in evaluating testing techniques/approaches* and documenting their characteristics, which may be beneficial for *future reuse*.

The results of this study can guide both IoT practitioners and researchers through the: ① compilation of objectives of testing based on quality attributes that are being considered or evaluated; ② provision of an overview of testing approaches and tools used in IoT systems testing; ③ compilation of various IoT systems used in the evaluation of testing techniques/ approaches discussed in PSs; and, ④ summarization of the challenges of IoT systems testing and future research directions.

The rest of this paper is organized as follows: Section 2 provides background on IoT systems testing. Section 3 describes our research methodology. Section 4 presents answers to our research questions, while Section 5 discusses our answers. Section 6 presents possible threats to the validity of our study. Section 7 discusses the related works. Finally, Section 8 concludes with some future directions.

2 Background

This section provides the background on IoT systems testing. First, we introduce the concept of IoT systems. Then, we discuss IoT systems testing, including testing objectives, testing approaches, testing tools, and testing challenges.

2.1 IoT Systems

An IoT system consists of "*networked sensors and smart objects whose purpose is to measure/control/operate on an environment in such a way to make it intelligent, usable, and programmable and capable of providing useful services to humans*" [17]. The number of layers in IoT systems varies depending on the specific business application and use case [18], [19]. Cisco, IBM, and Intel proposed a reference model for IoT systems that consists of seven layers [20], as follows:

❶ *The device layer*, also known as *sensing or perception layer*, includes devices, sensors, and actuators that collect data from surrounding environments and share them between themselves [21].
❷ *The network layer* transfers data from the device layer to the edge or cloud layer and vice versa.
❸ *The edge layer* processes the data generated by the device layer in real-time before passing data on to the cloud or other systems for further processing.
❹ *The cloud layer* stores the received data that comes from the sensing layer.
❺ *The data abstraction layer* manages and processes data from multiple devices/sensors with different data formats, and protocols, and provides a unified view of the data to the application layer.
❻ *The application layer* uses the data to deliver solutions like analytics, reporting, and control to end users.
❼ *The business layer* implements data-driven solutions and generates data useful in initiating collaboration between different stakeholders.

Recently, Fahmideh et al. [6] introduced a *human layer* in their reference model consisting of five layers. Although many reference models have been proposed for IoT systems, there is no single, universally agreed-upon reference model for IoT systems. Regardless of the use case and model, many IoT systems have four key layers. Device (thing) layer, network (infrastructure) layer, cloud (platform) layer, and application layer, as illustrated in Figure 1. We opted to focus on these four layers because they are important and commonly addressed in other IoT systems testing studies [22], [23], [24], [25].

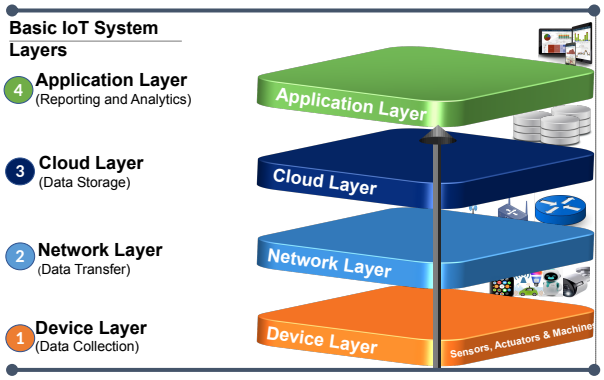


Fig. 1: IoT Key Layers

2.2 IoT Systems Testing

From a traditional software engineering perspective, testing involves the process of checking a system to identify errors [26]. Testing is critical in detecting software defects and minimizing their risks [27]. It encompasses any activity that evaluates a system's capability and ensures that it meets the required objectives [28]. Unlike conventional software, IoT systems have additional layers such as device, network, and cloud. The device layer may consist of many smart devices that communicate with the internet [17]. Therefore, testing an IoT system goes beyond assessing the software; it requires checking each layer and the system as a whole [29] with specific objectives. These objectives can vary de-

pending on the specific project, system requirements, and stakeholder expectations. To achieve the defined objectives, the testers use some approaches and tools. However, some challenges may arise when testing. In this section, we provide the background of testing objectives, approaches, tools, and challenges in the context of IoT systems.

2.3 Testing Objectives

Testing objectives can be expressed in the form of quality attributes. Quality is the ability of a product, service, system, component, or process to meet customer or user needs, expectations, or requirements [30]. A quality model is defined as *"a set of attributes, and of relationships between them, which provides a framework for specifying quality requirements and evaluating quality"* [31]. ISO/IEC/IEEE 24765:2017[1] proposed different quality attributes for traditional software systems [32]. However, there is no defined set of quality attributes for IoT systems. In this paper, we assess PSs to understand test objectives discussed in the form of quality attributes from traditional software systems, while identifying any new attributes used to evaluate the quality of IoT systems.

2.4 Testing Approaches

According to IEEE standards [33], a testing approach is a specific method employed to pick the particular test case values, and it includes testing levels, types, techniques, and practices. This may vary from very general (such as black box or white box) to very specific (such as minimum and maximum boundary values). As defined in ISO/IEC/IEEE29119-1:2022(E) [2], a testing approach is a high-level decision made during the test strategy designing activity. It includes determining the levels of testing (such as unit testing, integration testing, system testing, system integration testing, and acceptance testing), types of testing, testing techniques, testing practices, and the form of static testing to be used [34]. In this study, we aim to understand testing levels, testing types, and techniques used in IoT systems based on selected PSs.

2.5 Testing Tools

A testing tool is software that assists testers in conducting testing [29]. It can be software, such as capture-playback tools, or an artifact, such as a checklist to support manual testing. We want to analyze various testing tools used for both white box and black box testing in IoT systems based on selected PSs.

2.6 Testing Challenges

Testing challenges include any difficulty that arises when testing IoT systems due to their complex and dynamic nature, including the interconnection between different layers. We want to assess PSs to understand the challenges in testing IoT systems and suggest recommendations to overcome these challenges.

3 Research Method

We followed the updated PRISMA guidelines [35], [36] and Kitchenham et al. guidelines [37] to review and report our findings. We used three main phases: planning, conducting, and reporting the review. During the planning phase, we define the objective of SLR and review protocol. The objective of this SLR is defined in Section 1. The review protocol for conducting this SLR is defined in this section. It consists of six steps: ① defining the research questions, ② formulating the search query, ③ selecting the studies, ④ snowballing, ⑤ assessing the quality of the studies, and ⑥ extracting and analyzing the data.

3.1 Research Questions (RQs)

This study answers the following RQs:

- **RQ1: What Are Testing Objectives Considered?**
  Rationale: To ensure the quality of IoT systems, it is imperative to gain a clear understanding of the specific testing objectives, which can be achieved by assessing various quality attributes. While traditional system quality attributes can be applied, the complexity, heterogeneity, and distributed nature of IoT systems may necessitate the inclusion of additional attributes.
- **RQ2: What Are Testing Approaches Investigated?**
  Rationale: Practitioners may use several approaches to test IoT systems. We want to identify different testing approaches used to assess the quality of IoT systems. Understanding the available approaches for testing IoT systems provides valuable insights into the current state of the field and guides future research. This knowledge can help to improve the current testing practices and identify opportunities for enhancements.
- **RQ3: What Are Testing Tools Investigated?**
  Rationale: The tools for testing IoT systems hold significant importance and offer valuable assistance to both practitioners and researchers. Knowledge of these tools provides insights for researchers to explore potential enhancements, while also enabling practitioners to make informed choices that align with their specific requirements. By understanding the landscape of available testing tools, the testing process in IoT systems can be improved and tailored to better meet the needs of both researchers and practitioners.
- **RQ4: What Are IoT Systems Used For Evaluation?**
  Rationale: Developers may have different validation and verification objectives for different IoT systems. Knowing which systems are used for evaluation enables meaningful comparisons, providing valuable insights into the strengths and weaknesses of different approaches. This information is also essential for tailoring testing approaches to specific types of IoT systems.
- **RQ5: What Are Testing Challenges Identified?**
  Rationale: Testing an IoT system involves assessing multiple components, which can be tested individually or in combination with other components. The complexity inherent in IoT systems introduces unique challenges when it comes to ensuring their quality and reliability. While various approaches and tools exist for testing IoT systems, it is important to identify specific challenges that may impact the overall quality of these systems. By understanding and addressing these challenges, we can enhance testing processes, approaches, and tools to improve the overall quality and reliability of IoT systems.
- **RQ6: How Are Testing Challenges Addressed?**
  Rationale: Different PSs focused on different testing objectives, approaches, and tools as well as IoT systems. Our objective is to understand which challenges have been

1. https://www.iso.org/standard/71952.html
2. https://www.iso.org/standard/81291.html

successfully addressed and to identify those that remain unaddressed. This information is valuable for practitioners seeking solutions to testing challenges and researchers wanting to address challenges.

This study focuses on reviewing existing studies that studied testing IoT systems. The goal is to identify and discuss the findings of these studies and their insights into various aspects of IoT systems testing. To select the relevant studies, we applied the search query as illustrated in Section 3.2.

3.2 Search Query

We write our search query using the PICO (Population, Intervention, Comparison, Outcome) framework [38]:

❶ Obtaining the main terms from RQs.
❷ Identifying the possible synonyms of the main terms.
❸ Applying the Boolean OR to combine possible synonyms of the main terms.
❹ Applying the Boolean AND to combine expressions in the previous step.

As a result, we formulated the following search query:

> (IoT OR internet of thing OR IoT system OR internet of thing system OR IoT platform OR internet of thing platform OR IoT application OR internet of thing application OR IoT software OR internet of thing software) AND (test OR bug OR defect OR failure OR anomal* OR quality OR verification OR validation) AND (method OR technique OR approach OR process OR type OR level OR practice OR tool OR framework OR challenge OR concern OR problem OR layer OR component OR constituent OR attribute OR metric)

3.3 Studies Selection

1) Databases Identification: We selected 8 online digital libraries: ACM Digital Library, Compendex, IEEE Xplore, ScienceDirect, SpringerLink, Scopus, Web of Science, and Wiley. These digital libraries are the most commonly used to search for studies when conducting literature reviews in software engineering, as suggested by Dyba et al. [39]. Figure 2 shows the selected digital libraries.

We searched each of these digital libraries using our search query. Some digital libraries have limitations when performing queries. ScienceDirect accepts a maximum of eight connectors in search queries, SpringerLink does not accept the use of parentheses, ACM digital library does not accept wildcards. We customized the search query based on the specificities of each digital library. Table 1 shows the search queries executed in each digital library and the number of studies retrieved.

We applied online built-in filters to obtain 8,294 studies:

❶ Publication period: **2012-2022**
❷ Publication language: **English**
❸ Publication venues: **conferences**, **journals**, or **workshops**
❹ Excluded **security** testing studies.

2) Duplicates Removal: We exported the bibliographic entries to a spreadsheet for analysis. We used Excel's built-in filtering and sorting capabilities and manually removed 989 duplicate studies.

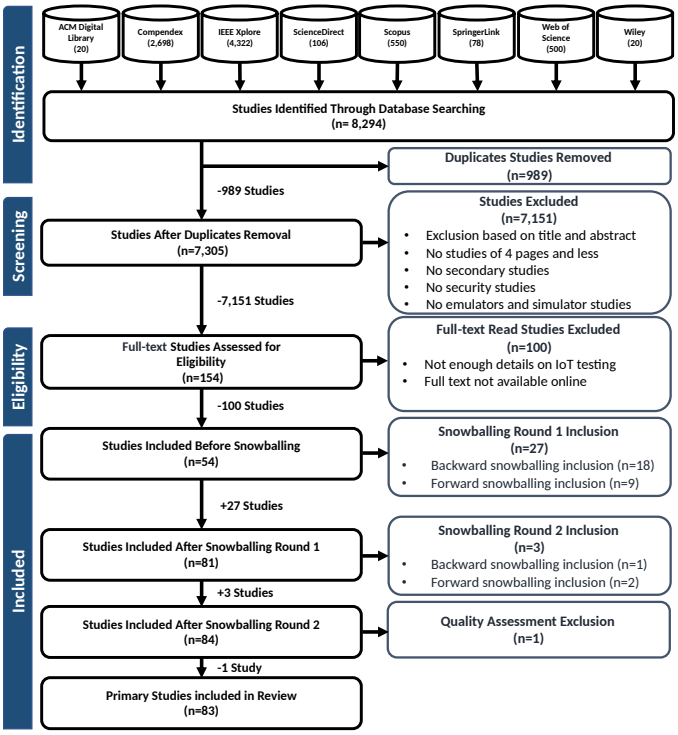


Fig. 2: PRISMA Flow for Primary Studies Selection

3) Screening: We defined a set of inclusion and exclusion criteria. We applied these criteria to select relevant primary studies and exclude irrelevant ones.

Inclusion Criteria: We considered the following inclusion criteria for PSs selection:

❶ The study is written in English.
❷ The study is published between 2012 and 2022.
❸ The study is published in journals, conferences, or workshops.
❹ The study explicitly discusses at least one aspect of IoT systems testing.
❺ The study has at least 4 pages.

Exclusion Criteria: We considered the following exclusion criteria:

❶ The study has less than 4 pages.
❷ The study is not a primary study.
❸ The study focuses on IoT security testing.
❹ The study focuses on emulators and simulators.
❺ The study has not been peer-reviewed.
❻ The study is a graduate thesis or project report.
❼ The study does not have its full text available online.
❽ The study does not provide enough details.

We applied our exclusion criteria in two steps based on the titles, abstracts, and full texts. We applied the first six criteria to the titles and abstracts, while we applied criteria 7 and 8 to the full texts. We chose to exclude PSs about IoT security testing despite the importance of this topic for two reasons. First, this topic deserves its own SLRs and has been already the subject of recent surveys, e.g., [40], [41], [42]. Second, security testing for IoT systems is a vast and complex topic, which would have overshadowed other objectives and increased the complexity and length of this article.

We initially excluded security studies using online, built-in

TABLE 1: Database Search Results

Database	Search Query	Results
ACM Digital Library	Title:((IoT OR ″internet of thing″) AND (test OR bug OR defect OR verification OR validation OR quality OR fail OR anomal) AND (tool OR framework OR platform OR engine OR technique OR process OR approach OR method OR level OR metric OR type OR layer OR component OR constituent OR strategy))	20
Compendex	(iot* OR ″internet of thing*″) AND (test* OR bug* OR defect* OR fail* OR ″quality assurance*″ OR verification OR validation) AND (tool OR framework OR testbed OR platform OR approach OR technique OR method OR process OR type OR level OR metric OR layer OR component OR constituent)	2,694
IEEE Xplore	(″Document Title″:iot OR ″internet of thing″ OR ″iot system*″ OR ″internet of thing system*″ OR ″iot platform*″ OR ″internet of thing platform*″ OR ″iot application″ OR ″internet of thing application″ OR ″iot software″ OR ″internet of thing software″) AND (″Document Title″:test OR bug OR defect OR failure OR anomal OR quality OR verification OR validation) AND (″Document Title″:method OR technique OR approach OR process OR type OR tool OR framework OR challenge OR concern OR problem OR artifact OR metric OR layer OR component OR constituent OR level OR attribute) AND NOT (″Document Title″:″security test″ OR ″penetration testing″ OR ″malware″ OR ″attack″)	4,342
ScienceDirect	(iot OR internet of thing) AND (test OR bug OR defect OR quality OR anomal OR fault)	106
Scopus	TITLE((iot OR ″internet of thing*″) AND (test* OR bug* OR defect* OR failure* OR quality* OR anomal* OR verification OR validation) AND (tool OR platform OR framework OR approach OR technique OR method OR process OR challenge OR concern OR problem OR metric OR artifact OR level OR type OR layer OR component OR constituent))	550
SpringerLink	″iot* test″ OR ″internet of thing* test″ OR ″iot* bug″ OR ″internet of thing* bug″ OR ″iot* defect″ OR ″internet of thing* defect″ OR ″iot* quality″ OR ″internet of thing* quality″ OR ″iot* verification″ OR ″internet of thing* verification″ OR ″iot* validation″ OR ″internet of thing* validation″ OR ″iot* anomal″ OR ″internet of thing* anomal″	78
Web of Science	(TI=((iot* OR ″internet of thing″*) AND (test OR bug OR defect OR fault OR anomal OR quality OR verification OR validation Or analysis))) AND TI=((tool OR framework OR platform OR engine OR method OR approach OR technique OR process OR strategy OR type OR level OR metric OR challenge OR problem OR concern OR issue OR layer OR component OR constituent))	500
Wiley	(IoT OR ″internet of thing″) AND (test OR bug OR defect OR fail OR anomal OR verification OR validation OR quality)	20

filters. Some studies using different, specific terms like *penetration testing* or *malware detection* remained in the retrieved primary studies. We applied the third exclusion criteria to remove those security studies manually. We excluded emulators and simulators in this SLR, as a comprehensive comparison of these tools has been previously conducted [43], [44] and we did not identify any new emulator or simulator that was not mentioned in these studies. To ensure that only relevant studies are included in our analysis, we added the last exclusion criteria to eliminate any study that discussed IoT systems testing aspects in a general sense, without providing sufficient detail on a specific objective, approach, tool, or challenge. While reading some studies, we observed that they did mention these aspects in their abstracts, introductions, or conclusions without providing enough information on the methodology or results to meet our inclusion criteria.

During the initial screening phase, we assessed the titles, abstracts, page counts, and whether qualified as a primary study or not. Two of our authors independently conducted the screening process using the defined exclusion and inclusion criteria. To ensure consistency when applying our inclusion and exclusion criteria, we compared the screening results for 30 randomly selected studies with Cohen's Kappa [45] shown in Equation 1. Table 2a shows the results of the two authors (A1, A2).

$$\kappa = \frac{P_o - P_e}{1 - P_e} \tag{1}$$

where:

P_o = relative observed agreement among authors
P_e = hypothetical probability of chance agreement

TABLE 2: Kappa Agreement

A1 \ A2	Yes	No	Total
Yes	17	0	17
No	1	12	13
Total	18	12	30

(a) First Round Screening

A1 \ A2	Yes	No	Total
Yes	16	1	17
No	0	3	3
Total	16	4	20

(b) Second Round Screening

We calculated Cohen's Kappa and obtained almost perfect agreement (k= 0.938). This value underscores the consistency of our screening approach. Throughout the screening process, the two authors met regularly to review their results, resolving disagreements through consensus and discussion. Upon completion of the first screening process, a total of 154 studies are selected for full-text review as potential candidates for PSs.

4) Eligibility Assessment: In the second round of the screening process, we used three criteria: emulators and simulators, security-related studies, and studies discussing certain aspects of IoT systems without enough details. Two authors independently applied these criteria to 154 studies by thoroughly reading them. To ensure a shared understanding of our inclusion and exclusion criteria, we compared the results of randomly selected 20 studies from the pool of 154 using Cohen's Kappa. Table 2b shows the results of two authors. We used these values to calculate the Kappa agreement, and we obtained a nearly perfect agreement (k=0.827). This value indicates the consistency between the two authors. We proceeded with confidence to complete the eligibility assessment process for the remaining studies. At the end of the process, we obtained a total of 54 PSs that met

our eligibility criteria.

3.4 Snowballing

This study has two threats to sampling adequacy.

1) Some of the relevant studies could be published in venues that are not indexed by the chosen databases, or studies excluded by the filters we used.
2) Some of the relevant studies could be published using some keywords that we did not include in our search query. For example, a study published on "IoT debugging", "IoT troubleshooting", or "IoT monitoring".

We conducted backward and forward snowballing in two rounds to find more studies.

1) Snowballing Round 1: We considered the references of all PSs to identify additional relevant studies. For each of the 54 studies identified in Section 4, we collected the cited references, leading to a total of 3,453 studies. We removed any study retrieved in our initial search and studies not focusing on IoT systems testing, and we retained 101 potentially relevant studies. We screened the 101 potentially relevant studies by removing duplicates (52 studies) and then applied the same inclusion/exclusion criteria as before, and we found 18 additional PSs.

Similarly, we used Google Scholar to identify all the studies that cited the selected 54 studies. We identified 1,645 studies. We removed any study retrieved previously and studies not focusing on IoT systems testing. We retained 152 potentially relevant studies for further screening. We screened these 152 potentially relevant studies by removing duplicates and applying the same inclusion and exclusion criteria as before. We added 9 PSs. By the end of this round 1, we found 27 additional PSs.

2) Snowballing Round 2: We considered the 27 studies found in the previous round, and we went through the snowballing process for them again. We checked all the references used in these 27 studies, and we found 5 new potential studies. We screened these 5 studies, and we obtained 1 PS based on the exclusion and inclusion criteria defined before.

We used Google Scholar to identify all studies that cited these 27 studies. We found 6 new potential studies. We applied the same exclusion and inclusion criteria and kept 2 additional PSs. After this round, we obtained 3 additional PSs. Table 3 summarizes the results of this process.

TABLE 3: Backward and Forward Snowballing

Snowballing	Round	Retrieved	Included
Backward	Round 1	101	18
Forward	Round 1	152	9
Backward	Round 2	5	1
Forward	Round 2	6	2
Total		**264**	**30**

3.5 Quality Assessment

We devised a set of guidelines, following the recommendations of Kitchenham et al. [37], to assess the quality of primary studies (PSs). Subsequently, we formulated a quality checklist consisting of 19 questions in five categories: study design, conduct, analysis, conclusion, and implication. Each question is answered with a choice of "No," "Partially," or "Yes," and scores of 0, 0.5, and 1 are assigned to these responses, respectively. Two authors applied the checklist independently on each PS. We compared the results and resolved any discrepancies through discussion. The outcome of this evaluation is the percentage of PSs answering each question in the checklist, as shown in Table 5. We calculated the quality of each study by adding the scores of all applicable questions and calculating the corresponding final percentage. The authors agreed to keep the studies with at least a 75% score. Consequently, one study, which scored less than the set threshold, was excluded. We thus obtained 83 PSs.

3.6 Data Extraction and Analysis

To obtain the data required to answer each research question (RQ1-RQ6), we studied each PS. We extracted the various data items described in Table 4, by applying Algorithm 1. The data extraction form we used is publicly available on Ptidej or on Zenodo websites.

TABLE 4: Data Extraction Elements

#	Data Item	Description	RQs
1	Code	Unique Identifier of the PS.	
2	Study Title	The title of the PS.	
3	Year	Year the PS was published.	
4	Venue	Where the PS was published.	
5	Source	The source of the PS.	
6	Author(s)	The authors of the PS.	
7	Focus	The main focus of the PS.	
8	Contribution	The key contribution of the PS.	
9	Category	Category of the study.	
10	Tools	The name of the proposed tool.	RQ3
11	Layer	IoT layer studied.	
12	Approach	Approach used.	RQ2,RQ4,RQ6
13	QA	Quality attributes discussed.	RQ1
14	Challenge	Challenge discussed in the PS.	RQ5, RQ6
15	Research or Evaluation Method	The method used in the PS.	RQ4

4 Results

In this section, we answer every research question. We use the notation of 'P' (e.g., [P21]) to distinguish primary studies from other references (e.g., [21]). The list and details of all PSs are available online[3]. RQs to be addressed are:

❶ **RQ1**: What Are Testing Objectives Considered?
❷ **RQ2**: What Are Testing Approaches Investigated?
❸ **RQ3**: What Are Testing Tools Investigated?
❹ **RQ4**: What Are IoT Systems Used For Evaluation?
❺ **RQ5**: What Are Testing Challenges Identified?
❻ **RQ6**: How Are Testing Challenges Addressed?

In the following subsections, we describe the results of each research question and the corresponding observations. Prior to presenting answers to our research questions, we start with the preliminary findings (PF) of bibliographic data. We present the main areas of research focus in IoT systems testing and commonly used research methods.

4.1 PF1: What Publication Trends Are Observed?

1) PF1.1: Publication Trends Over the Past Ten Years Figure 3 shows the trend of publications on IoT systems testing over the past 10 years. We observed a steady increase in PSs, with a peak in 2018 when 19 PSs (22.9%) were published. We observed a decrease from 2019 until the end of 2022. Most PSs are published in conferences. We did not find any PS published in 2012. The first PS was published by Reetz et al. in 2013 [P28]. From 2014 to 2016, on average, 3 PSs were published each year, and at least 1 PS was

3. https://www.ptidej.net/downloads/replications/tse23b/

TABLE 5: Quality Assessment Checklist

ID	Question	Percentage of PSs			
		Yes	**Partially**	**No**	**N/A**
Design					
QA1	Are the IoT systems testing or quality assurance activities clearly stated?	100.0%	0.0%	0.0%	0.0%
QA2	Are the studies clearly discussed either testing tools, approaches, quality attributes, or testing challenges of IoT systems?	100.0%	0.0%	0.0%	0.0%
QA3	Are the aims of the studies clearly stated?	100.0%	0.0%	0.0%	0.0%
QA4	Are the RQs relevant?	100.0%	0.0%	0.0%	0.0%
Conduct					
QA5	Are the components of IoT systems addressed in the studies clearly stated?	85.5%	2.4%	0.0%	12.0%
QA6	Are the experiments or case studies conducted?	85.5%	2.4%	0.0%	12.0%
QA7	Are the details of the system under test described?	85.5%	2.4%	0.0%	12.0%
QA8	Are the results of the studies validated?	85.5%	2.4%	0.0%	12.0%
Analysis					
QA9	Are the aims of the analysis clearly stated?	98.8%	0.0%	1.2%	0.0%
QA10	Are specific tools or algorithms used to analyze the data?	63.9%	3.6%	20.5%	12.0%
QA11	Are the data used in the analysis clearly stated in PSs?	81.9%	0.0%	3.6%	14.5%
QA12	Is the statistical analysis performed correctly?	81.9%	0.0%	3.6%	14.5%
QA13	Are the data and/or tools used available?	96.4%	0.0%	3.6%	0.0%
Conclusion					
QA14	Are validity threats discussed in PSs?	84.3%	0.0%	15.7%	0.0%
QA15	Are the results compared with state-of-the-art practices?	72.3%	0.0%	27.7%	0.0%
QA16	Do the results support the conclusions?	96.4%	0.0%	3.6%	0.0%
Implication					
QA17	Did PSs extend or improve the existing tools/approaches?	85.5%	2.4%	0.0%	12.0%
QA18	Did PSs discuss future research as an improvement or enhancement of proposed tools or approaches?	85.5%	2.4%	0.0%	12.0%
QA19	Did PSs discuss any solution for IoT systems testing challenges?	4.8%	0.0%	0.0%	95.2%

published in a journal. In 2017, no PS was published in a journal. However, 10 PSs were published in conference proceedings. The majority of PSs (87.95%) were published during the period 2017-2022. Despite a slight continuous decrease in PSs since 2019, we observed that 6 PSs (7.23%) were published in journals in 2022.

The 83 PSs were published in a total of 66 different venues, including 40 conferences, 18 journals, 5 workshops, and 3 book series, as shown in Table 6. Among the journals, *IEEE Access* and *IEEE Internet of Things Journal* stand out with 4 PSs each, accounting for 4.8% of the total. Each remaining journal has only 1 PS. In terms of conferences, the *ACM/IEEE Conference on Internet of Things Design and Implementation* has the highest number of PSs with 3 (3.6%), and 9 other conferences have 2 PSs each. Each remaining 30 conferences has only 1 PS.

Lessons from PF1.1

IoT systems testing studies showed a steady increase until 2018, followed by a slight decline. The majority of PSs were published at conferences, but there has been a recent rise in journal articles.

2) PF1.2: Top Active Authors and Affiliations The 83 PSs were written by 288 unique authors. Table 7 shows the top active authors, who are from both industry and academia. The most active contributors to our PSs are from the industry, which may suggest that industry-specific challenges are the driving force behind the research initiatives in this field.. Le Gall coauthored the highest number of PSs (7.23%) [P9], [P11], [P13], [P15], [P38], [P48] on *interoperability testing*, *conformance testing*, *testing framework*, and *model-based testing as a service*. Several authors published 4 PSs, including Leotta, Ricca, and Watteyne. There are multiple authors with 3 PSs, such as Ahmad, Ancona, Franceschini, Gutierrez-Madronal, Baqa, Kuemper, Medina-Bulo, Olianas, Ribaudo, and Toenjes. Some authors such as S. Ahmed, Ari, L. Badr, Baranwal, Bellekens, Bures, Bonnet, Clerissi, and others have 2 PSs each.

Authors are from multiple countries as shown in Figure 4, indicating a global interest in the topic. Many countries have only one PS (1.2% of total PSs). Some countries have 2 or 3 PSs (2.4% or 3.6%). Few countries, such as China, South Korea, and Taiwan have higher representation with 4 or more PSs (4.8% or more of the total PSs). France has the highest number of PSs (11 PSs), accounting for 13.3%, followed by the United Kingdom, United States, Germany, and Spain, each with 8 PSs (9.6%).

EGM in France has the highest number of PSs (7.23%). INRIA in France follows closely with 5 PSs (6.0%). University

TABLE 6: Publication Sources

Document Type	Source Title	# PSs	% PSs
Journal	IEEE Access	4	4.8%
Journal	IEEE Internet of Things Journal	4	4.8%
Conference	ACM/IEEE International Conference on Internet of Things Design and Implementation (IoTDI)	3	3.6%
Book Chapter	EAI/Springer Innovations in Communication and Computing	2	2.4%
Conference	Global Internet of Things Summit (GIoTS)	2	2.4%
Conference	IEEE/ACM International Conference on Utility and Cloud Computing Companion	2	2.4%
Conference	IEEE International Conference on Software Quality Reliability and Security	2	2.4%
Conference	IEEE International Conference on Software Testing, Verification and Validation	2	2.4%
Conference	IEEE World Forum on Internet of Things	2	2.4%
Conference	International Conference on Safety and Security in IoT, International Conference on Interoperability in IoT	2	2.4%
Conference	International Conference on Smart City Applications	2	2.4%
Conference	International Conference on Web Engineering	2	2.4%
Conference	Springer Communications in Computer and Information Science (CCIS)	2	2.4%
Workshop	ACM International Workshop on the Internet of Safe Things	1	1.2%
Conference	ACM Multimedia Systems Conference	1	1.2%
Book Chapter	Advances in Computers	1	1.2%
Journal	Arabian Journal for Science and Engineering	1	1.2%
Conference	Asia-Pacific Software Engineering Conference	1	1.2%
Book Chapter	Building the Future Internet Through FIRE: 2016 FIRE Book	1	1.2%
Journal	Computer Networks	1	1.2%
Journal	Computing	1	1.2%
Journal	Empirical Software Engineering	1	1.2%
Conference	European Safety and Reliability Conference, Probabilistic Safety Assessment and Management Conference	1	1.2%
Workshop	European Test Workshop	1	1.2%
Conference	European Wireless Conference	1	1.2%
Journal	Future Generation Computer Systems	1	1.2%
Conference	IEEE Global Conference on Internet of Things (GCIoT))	1	1.2%
Conference	IEEE/ACM International Symposium on Cluster Cloud and Internet Computing	1	1.2%
Workshop	IEEE/ACM International Workshop on Software Engineering Research and Practices for the IoT	1	1.2%
Conference	IEEE International Conference on Intelligent Computing and Information Systems	1	1.2%
Conference	IEEE International Congress on Internet of Things	1	1.2%
Journal	IEEE Micro	1	1.2%
Conference	IEEE Region 10 Symposium	1	1.2%
Journal	IEEE Sensors Journal	1	1.2%
Conference	IEEE SoutheastCon	1	1.2%
Journal	IEEE Transactions on Reliability	1	1.2%
Journal	IET Software	1	1.2%
Conference	International Conference on Ad-Hoc Networks and Wireless	1	1.2%
Conference	International Conference on Advances in System Testing and Validation Lifecycle	1	1.2%
Conference	International Conference on Computer Science and Information Technology	1	1.2%
Conference	International Conference on Current Trends towards Converging Technologies	1	1.2%
Conference	International Conference on Industrial Engineering and Applications	1	1.2%
Conference	International Conference on Internet of Things as a Service	1	1.2%
Conference	International Conference on Internet of Things Smart Innovation and Usages	1	1.2%
Conference	International Conference on Internet of Things, Big Data and Security	1	1.2%
Conference	International Conference on Inventive Communication and Computational Technologies	1	1.2%
Conference	International Conference on Nascent Technologies in Engineering	1	1.2%
Conference	International Conference on Networking Sensing and Control	1	1.2%
Conference	International Conference on Perspective Technologies and Methods in MEMS Design	1	1.2%
Conference	International Conference on Sensing and Instrumentation in IoT Era	1	1.2%
Conference	International Conference on Software Engineering	1	1.2%
Journal	International Journal of Computer Applications	1	1.2%
Conference	International Multi-Conference on Systems, Signals, and Devices	1	1.2%
Conference	International Symposium on Distributed Computing and Applications for Business, Engineering, and Science	1	1.2%
Conference	International Symposium on Leveraging Applications of Formal Methods	1	1.2%
Conference	International Symposium on Networks, Computers, and Communications	1	1.2%
Workshop	International Symposium on Software Testing, and Analysis (ISSTA/ECOOP Workshops 2018)	1	1.2%
Journal	Journal of Systems and Software	1	1.2%
Journal	Journal of the National Institute of Information and Communications Technology	1	1.2%
Journal	Mobile Networks and Applications	1	1.2%
Conference	RoEduNet Conference: Networking in Education and Research	1	1.2%
Journal	Software - Practice and Experience	1	1.2%
Journal	Software Quality Journal	1	1.2%
Journal	Wireless Communications and Mobile Computing	1	1.2%
Workshop	Workshop on Benchmarking Cyber-Physical Systems and Internet of Things	1	1.2%
Total		**83**	**100.0%**

of Cádiz in Spain, Università di Genova in Italy, and Czech Technical University in Prague, Czech Republic have 3 PSs (3.6% of the total PSs) each. Several institutions, including the University of Porto in Portugal, TU Berlin in Germany, EURECOM in France, Institute of Computer Science at the University of Osnabrueck in Germany, and University of Applied Sciences Osnabrueck in Germany, and others as shown in Table 8 have 2 PSs each.

Lessons from PF1.2

Notable contributors are Le Gall, Leotta, Ricca, and Watteyne. The most active institutions are EGM and INRIA. Prominent countries include France, the United Kingdom, the United States of America, Germany, and Spain. The findings reflect a global interest in IoT systems testing and highlight the active participation of industry experts.

TABLE 7: Authors With 2 or More PSs

Author Name	# PSs	Affiliation	Author Name	# PSs	Affiliation
Franck Le Gall	6	Easy Global Market (EGM), France	Christian Bonnet	2	EURECOM, France
Maurizio Leotta	4	Università di Genova, Italy	Diego Clerissi	2	Università di Genova, Italy
Filippo Ricca	4	Università di Genova, Italy	Soumya Kanti Datta	2	EURECOM, France
Thomas Watteyne	4	INRIA, France	Giorgio Delzanno	2	Università di Genova, Italy
Abbas Ahmad	3	EGM, France	Ragib Hasan	2	University of Alabama at Birmingham, USA
Davide Ancona	3	Università di Genova, Italy	Jaeyoung Hwang	2	Sejong University, South Korea
Luca Franceschini	3	Università di Genova, Italy	Koray Incki	2	Özyeğin University, Turkey
Lorena Gutierrez-Madronal	3	University of Cádiz, Spain	Jaeseung Song	2	Sejong University, South Korea
Hamza Baqa	3	EGM, France	Yasser Karim	2	University of Alabama at Birmingham, USA
Daniel Kuemper	3	University of Applied Sciences Osnabrueck, Germany	Moez Krichen	2	Al-Baha University, Saudi Arabia; University of Sfax, Tunisia
Inmaculada Medina-Bulo	3	University of Cádiz, Spain	Luis Sanchez	2	University of Cantabria, Spain
Dario Olianas	3	Università di Genova, Italy	Noha Medhat	2	Ain Shams University, Egypt
Marina Ribaudo	3	Università di Genova, Italy	Mengxuan Zhao	2	EGM, France
Ralf Toenjes	3	University of Applied Sciences Osnabrueck, Germany	Sherin Moussa	2	Ain Shams University, Egypt
Bestoun S. Ahmed	2	Czech Technical University, Czech Republic	Elke Pulvermueller	2	Institute of Computer Science, University of Osnabrueck, Germany
Ismail Ari	2	Özyeğin University, Istanbul, Turkey	Manisha Singh	2	Banaras Hindu University, India
Nagwa L. Badr	2	Ain Shams University, Egypt	Federico Sismondi	2	IRISA, France
Gaurav Baranwal	2	Banaras Hindu University, India	Mohammed Tolba	2	Ain Shams University, Egypt
Xavier Bellekens	2	Czech Technical University, Czech Republic	Cesar Viho	2	IRISA, France
Miroslav Bures	2	Czech Technical University, Czech Republic	Sebastien Ziegler	2	Mandat International, Switzerland

✵ **INRIA**: Institut National de Recherche en Informatique et en Automatique.

TABLE 8: Affiliations of authors with at least 2 PSs

Affiliation Name	# PSs	Affiliation Name	# PSs
EGM, France	6	Université de Franche-Comté, France	2
INRIA, France	5	University of Porto, Portugal	2
University of Cádiz, Spain	3	TU Berlin, Germany	2
Università di Genova, Italy	3	EURECOM, France	2
Czech Technical University in Prague, Czech Republic	2	University of Osnabrueck, Germany	2
Banaras Hindu University, India	2	University of Applied Sciences Osnabrueck, Germany	2
University of Alabama at Birmingham, USA	2	Özyeğin University, Turkey	2
Ain Shams University, Egypt	2	IRISA, France	2
Al-Baha University, Saudi Arabia	2	Mandat International, Switzerland	2
University of Sfax, Tunisia	2	University of Surrey, United Kingdom	2
Sejong University, South Korea	2	University of Cantabria, Spain	2
Korea Electronics Technology Institute, South Korea	2		

✵ **PSs**:Primary Studies.

3) PF1.3: Collaboration in PSs **Co-authorship by country**: We considered the countries with at least two PSs. Authors of PSs in France collaborated with other authors from 7 different countries on 11 PSs. The authors of PSs in the USA collaborated with authors from three different countries. Our observation suggests that international collaboration in IoT systems testing is critical, with France serving as a prominent hub for collaboration with authors from multiple countries.

Co-authorship by affiliation: We identified 132 different institutions in 83 PSs. We considered institutions with at least 2 PSs. We assigned a unique number to each institution. Table 8 shows the assigned number (code) we used for the top 23 institutions. We provided the entire list of all institutions and their assigned code in the replication package. We used VOSViewer [4] package to visualize the collaboration between different institutions. EGM has 17 links. Institutions like INRIA in France, Sejong University and Korea Electronics Technology Institute in South Korea, EURECOM in France, and the University of Cantabria in Spain collaborated with EGM. INRIA has 22 links. Seventeen institutions collaborated with INRIA on 1 PS, while five institutions collaborated with INRIA on at least 2 PSs. The institutions with more collaborations are industry-based.

Co-authorship by Authors: We assessed the collaborative efforts among authors by examining those who participated in at least two primary studies (PSs). Our analysis revealed that several authors collaborated on multiple PSs. One such author is Le Gall of EGM, who collaborated with eight other authors on at least two PSs. Maurizio and Filippo, both from Università di Genova in Italy, collaborated in all four PSs [P8], [P47], [P77], [P79] they coauthored. They collaborated with six other authors from the same institution. Thomas of INRIA in France also had four PSs and collaborated with authors from IRISA (Federico and Cesar) and Sebastien (Mandat International). The analysis suggests

4. https://www.vosviewer.com/

Algorithm 1 Data Extraction Algorithm

```
1: function DataExtraction(Studies studies)
2:     Input
3:         S    Set of studies
4:         N    Set of categories
5:     Output
6:         K    Studies in each category
7:     Let S = studies, N={QA, TA, TT, TC}, K[] = ∅
8:     Let i = 1
9:     Let j = 1
10:    while S ≠ ∅ do
11:        Select study P_i from S
12:        Read P_i to identify its primary category N_j
13:        if N_j is found in N then
14:            Put P_i in K[j]
15:            Read P_i to find another category N_l; l = ++j
16:            if K_l is found in N then
17:                Put P_i in K[l]
18:                Go to Line 15
19:            else
20:                Go to Line 26
21:            end if
22:        else
23:            Flag P_i for authors' discussion
24:        end if
25:        Extract data items and fill data extraction form
26:        Remove P_i from S
27:        i++
28:        Go to Line 10
29:    end while
30:    return K
31: end function
```

that collaboration among authors is common, with some authors collaborating on multiple PSs.

Lessons from PF1.3

Several researchers have extensive collaborations with authors from other countries, which may evidence the global reach of their work. Additionally, the involvement of authors from the industry in these collaborations may underscore the practical and real-world applicability of their research efforts, which may contribute to more comprehensive and impactful approaches to IoT systems testing.

4.2 PF 2: What Are Aspects of Testing Studied?

We analyzed the 83 PSs to identify the focus of each PS. Some of PSs ([P12], [P14], [P15], [P17], [P19], [P31], [P41], [P42], [P47], [P65], [P67], [P70], [P72], [P73], [P79], [P83]) focused on more than one aspect, therefore, we agreed to focus on the primary and secondary aspects for each PS and considered PSs falling under these aspects to address the RQs. Once each author completed the analysis, we calculated Cohen's Kappa and obtained almost perfect agreement (k= 0.9). We solved any disagreement through discussion among the team members. Table 9 shows the focus of each PS. We noticed that Some PSs discussed more than one testing aspect. For example, PSs such as [P12], [P79] primarily focus on testing approaches but also mention some testing tools. The primary category for each PS refers to the testing aspect that received more emphasis in the study, while the secondary category relates to other aspects mentioned in the PS with less emphasis.

TABLE 9: Areas of Focus for IoT Systems Testing in PSs

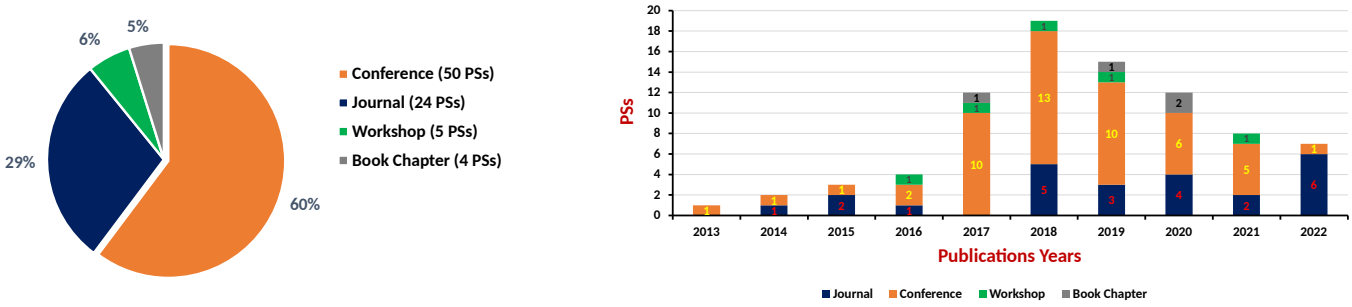
Category	PSs	% PSs
Quality Attributes		
Primary Category	[P1], [P16], [P52], [P53], [P54], [P62], [P63], [P67]	9.6%
Secondary Category	[P4], [P9], [P11], [P22], [P26], [P38], [P41], [P43], [P44], [P48], [P50], [P51], [P60], [P65],[P66], [P69], [P71], [P75],[P78], [P80], [P83]	25.3%
Testing Approaches		
Primary Category	[P2], [P3], [P5], [P6], [P7], [P8], [P9], [P11], [P12], [P14], [P15], [P22], [P23], [P24], [P26], [P27], [P28], [P30], [P32], [P37], [P38], [P39], [P40], [P41], [P42], [P48], [P49], [P50], [P51],[P55], [P56], [P57], [P58], [P59], [P60], [P64], [P65], [P68], [P70], [P71], [P72], [P73], [P74], [P75], [P76], [P77], [P78], [P79], [P81], [P83]	60.2%
Secondary Category	[P17], [P18], [P19], [P31], [P47], [P67]	7.23%
Testing Tools		
Primary Category	[P4], [P10] ,[P13], [P19], [P20], [P21], [P25], [P29], [P31], [P35], [P36], [P43], [P44], [P45], [P46], [P47], [P61], [P66], [P69], [P80], [P82]	25.3%
Secondary Category	[P12],[P15], [P42],[P65], [P79]	4.8%
Testing Challenges		
Primary Category	[P17], [P18], [P33], [P34]	4.8%
Secondary Category	[P14], [P70], [P72], [P73]	4.8%

Out of the 83 PSs studied, the majority (56 or 67.4%) relate to approaches for testing IoT systems. Of those 56 PSs, 50 PSs (60.2%) focused primarily on testing approaches, while 6 PSs (7.23%) discussed testing approaches together with other aspects. Another aspect discussed in PSs is testing tools with 25 PSs (30.1%). Of those 25 PSs, 21 PSs primarily focused on different tools used for testing IoT systems. Additionally, 4 PSs discussed testing tools alongside other aspects. 29 PSs (34.9%) discuss test objectives based on quality attributes. Of those 29 PSs, 8 PSs (9.6%) primarily focused on test objectives, while 21 PSs (25.3%) discussed test objectives alongside other aspects. Testing challenges are the least studied aspect, with only 8 PSs (9.64%) covering this topic. Of those 8 PSs, only 4 PSs (4.8%) primarily focused on testing challenges, while the other 4 PSs discussed testing challenges alongside other aspects.

We analyzed PSs to understand which layers of IoT systems are discussed. Table 10 shows the IoT system layers identified in PSs.

The most discussed layer is the *device layer* with 63 PSs, accounting for 75.90%. Fifty PSs (60.24%) discuss testing the *network layer*. Forty PSs (48.19%) discuss the *application*

This article has been accepted for publication in IEEE Transactions on Software Engineering. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSE.2024.3363611



(a) Distribution of PSs by Study Type

(b) Distribution of PSs by Year

Fig. 3: Distribution of PSs

layer testing. Eighteen PSs (21.69%) mention the testing of the *cloud layer*. We observed that the authors of PSs used different terms to refer to describe approaches. Some authors used the term *framework* [P6], [P14], [P23], [P27], [P38], [P42], [P47], [P50], [P55], [P76], [P83], while others used the same term to denote a tool [P19], [P80]. Similarly, some authors referred to it as a *platform* [P17], while others used the same term to describe a tool [P15], [P31]. Other authors used the term *methodology* [P59], [P83]. Others used the term *algorithm* [P2], [P3], [P5], [P58], [P59], [P60], while others used the term *test strategy* [P70]. 55 PSs (66.2%) discuss testing approaches in the form of testing practices, techniques, types, and levels. 15 PSs (18.0%) provide details on testbeds, while 19 PSs (22.8%) mention end-user testing tools. We provide more details on testbeds, testing techniques, types, and levels in RQ2 and RQ3.

Lessons from PF2

The device layer was the most discussed layer for IoT system testing. Cloud layer is the least considered in the selected PSs. Testing approaches include testing frameworks and platforms. End-user testing tools and testbeds are commonly discussed in PSs.

4.3 PF 3: What Research/Evaluation Methods Are Used?

Table 11 provides an overview of the research and evaluation methods used in PSs. *Experiments* emerge as the most popular choice among PSs, with forty-three of the total PSs opting for this method. *Case studies* stand out as the second most preferred method, with thirteen PSs. We observed that the findings from these studies may have limitations on the generalizability of the results. *Surveys* rank as the third preferred method, with eleven PSs. Surveys can provide valuable insights into the perceptions and opinions of IoT practitioners. However, they have potential limitations and criticisms, including the risk of bias, low response rate, and dependence on self-reported data. *Combining case studies with experiments* are also common research methods used in PSs, with eleven PSs. We observed that combining case studies and experiments can enhance the validity of the findings. *Grounded theory* is the least used research method, with five PSs: four PSs focused on testing approaches [P70], [P71], [P76], [P81], and one PS addressed quality issues [P62].

Lessons from PF3

Experiments are the most commonly used evaluation method, with case studies and surveys also being used. Some studies combined case studies and experiments, allowing for an in-depth exploration of specific situations while also providing evidence of the generalizability of the proposed solutions. In contrast to studies relying solely on either case studies or experiments, those combining both methods yield comprehensive results.

4.4 RQ1: What Are Testing Objectives Considered?

To understand and evaluate the quality of IoT systems, researchers considered various testing objectives. These objectives are expressed in terms of quality attributes. We collected the quality attributes investigated in IoT systems. A quality attribute (QA) refers to a measurable or testable property of a system, that measures the degree to which the system meets this attribute. While traditional software systems have well-known quality attributes [46], no universally recommended quality attributes for IoT systems. Our objective is to identify the quality attributes of IoT systems in PSs. To understand why IoT systems should be tested, these quality attributes serve as test objectives.

We conducted a thorough analysis of quality attributes discussed in PSs to identify various objectives considered for testing IoT systems. Table 12 reports all quality attributes identified in PSs. In addition to PSs discussing quality attributes as primary or secondary categories as shown in Table 9, We also included any quality attributes from other PSs that discuss them in general.

We identified 47 attributes that drive test objectives. *Interoperability* and *performance* emerged as the two most frequent quality attributes (in seventeen PSs or 20.4%) of IoT systems. *Scalability* (discussed in thirteen PSs or 15.6%) is the third most discussed quality attribute in PSs. *Usability* (in ten PSs or 12.0%) is the next most discussed in PSs. *Conformance* (discussed in nine PSs or 10.8%) is also among the top discussed quality attributes in PSs. *Reliability* is discussed in seven PSs (8.4%). *Resource utilization* or *energy efficiency* is mentioned in six PSs (7.2%). In addition to the attributes listed in Table 12, [P83] introduces *artificiality*, and *concor-*

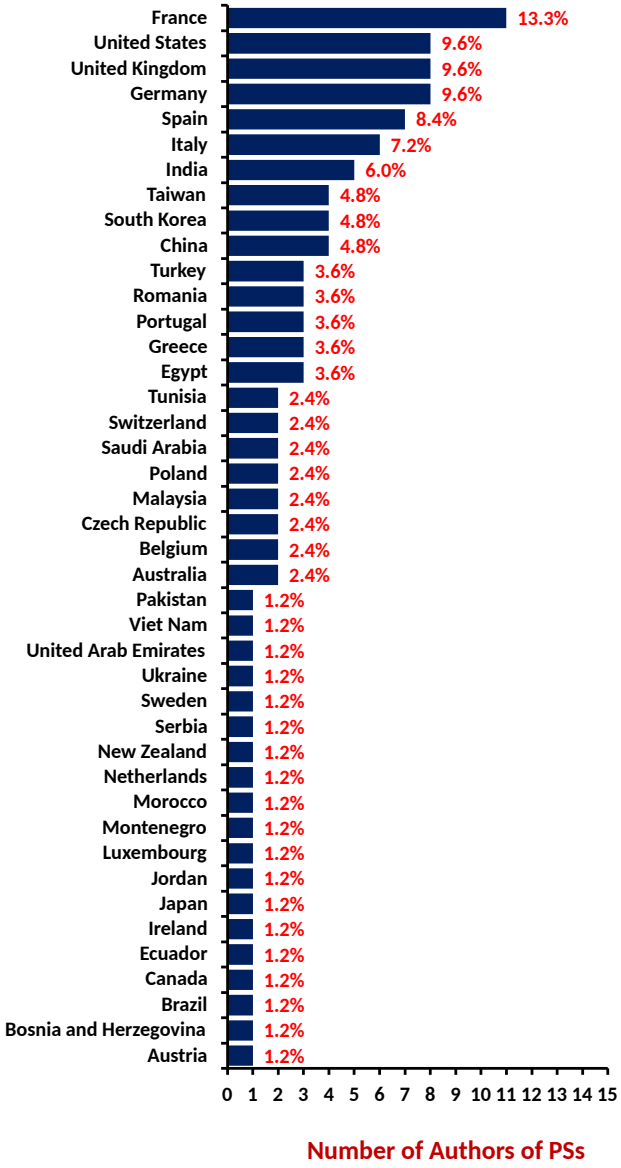


Fig. 4: Authors' Countries

dance as quality attributes for IoT systems. Consequently, we observed that testing interoperability and performance are the top test objectives, followed by testing scalability, usability, conformance, reliability, and energy efficiency.

We observed that quality attributes such as *connectivity*, *device lifespan expectancy*, *distributivity*, *dynamicity*, and *energy efficiency* are required for IoT systems. Traditional software systems may not require these attributes. Attributes such as availability, compatibility, correctness, fault-tolerance, functional suitability, installability, interoperability, maintainability, performance, portability, reliability, resource utilization, reusability, and usability are already defined in ISO/IEC 25010 [46]. In this section, we propose the definitions for attributes not defined in ISO.

- **Accessibility**. The degree to which users can access and use the IoT system effectively.
- **Accuracy**. The accuracy of sensor data is defined as the precision in data measured in terms of the standard deviation of a data value relative to its mean [P16].
- **Artificiality**. Artificiality in IoT systems refers to the extent to which sensor data is processed or created. This can include data sourced from a single sensor, aggregated values from multiple sources, or artificially interpolated values generated using algorithms [P83].

TABLE 10: IoT System Layers Identified in PSs

PS	Layer			
	Device	Network	Cloud	Application
[P2]	✓	-	-	✓
[P3]	✓	✓	-	✓
[P4]	✓	-	✓	-
[P5]	-	-	-	✓
[P6]	✓	✓	-	✓
[P7]	✓	-	-	-
[P8]	✓	✓	✓	✓
[P9]	✓	✓	-	-
[P10]	✓	✓	✓	✓
[P11]	✓	-	-	-
[P12]	-	-	-	✓
[P13]	✓	✓	✓	✓
[P14]	✓	✓	-	✓
[P15]	✓	✓	-	✓
[P16]	✓	-	-	-
[P17]	✓	✓	✓	✓
[P18]	✓	✓	✓	✓
[P19]	-	✓	✓	-
[P20]	✓	✓	-	-
[P21]	✓	-	-	✓
[P22]	-	✓	-	-
[P23]	✓	✓	✓	✓
[P24]	✓	✓	-	-
[P25]	✓	✓	-	-
[P26]	✓	✓	-	✓
[P27]	✓	✓	-	-
[P28]	✓	✓	-	✓
[P29]	✓	✓	-	✓
[P30]	✓	✓	-	✓
[P31]	-	✓	-	-
[P32]	-	✓	-	-
[P35]	✓	-	-	-
[P36]	✓	✓	-	-
[P37]	✓	-	-	-
[P38]	✓	-	-	-
[P39]	-	-	-	✓
[P40]	-	-	-	✓
[P41]	-	✓	-	-
[P42]	✓	✓	-	✓
[P43]	✓	✓	✓	-
[P44]	✓	-	-	-
[P45]	✓	✓	-	-
[P46]	✓	✓	-	-
[P47]	✓	✓	-	✓
[P48]	✓	-	-	✓
[P49]	-	✓	-	-
[P50]	✓	✓	-	✓
[P51]	✓	✓	✓	✓
[P55]	✓	✓	✓	✓
[P56]	✓	✓	✓	✓
[P57]	✓	✓	✓	✓
[P58]	✓	-	-	-
[P59]	✓	-	-	-
[P60]	✓	-	-	-
[P61]	✓	-	-	-
[P62]	✓	✓	✓	✓
[P64]	✓	✓	✓	✓
[P65]	✓	✓	✓	✓
[P66]	✓	-	-	✓
[P67]	✓	✓	✓	-
[P68]	✓	✓	-	✓
[P69]	-	✓	-	-
[P70]	✓	✓	-	✓
[P71]	✓	-	-	-
[P72]	✓	✓	-	-
[P73]	✓	✓	-	✓
[P74]	-	✓	-	-
[P75]	✓	✓	-	✓
[P76]	-	✓	-	-
[P77]	✓	-	-	✓
[P78]	✓	-	-	✓
[P79]	✓	✓	✓	✓
[P80]	✓	✓	-	-
[P81]	✓	-	-	-
[P82]	✓	-	-	-
[P83]	✓	-	-	✓
Total	**63**	**50**	**18**	**40**
Percentage	**75.90%**	**60.24%**	**21.69%**	**48.19%**

✺✓: Discussed; - : Not Discussed.

TABLE 11: Research/Evaluation Methods

Research Method	PSs	#	%
Experiments	[P2], [P3], [P6], [P7], [P10], [P11], [P13], [P15], [P18], [P20], [P21], [P22], [P23], [P25], [P26], [P29], [P31], [P32], [P35], [P36], [P37], [P41], [P43], [P44], [P45], [P46], [P48], [P51], [P55], [P56], [P57], [P58], [P59], [P60], [P61], [P66], [P67], [P69], [P72], [P74], [P80], [P82], [P83]	43	51.8%
Case Studies	[P9], [P14], [P19], [P24], [P28], [P30], [P38], [P39], [P40], [P49], [P50], [P65], [P73]	13	15.66%
Surveys	[P1], [P12], [P33], [P34], [P52], [P53], [P54], [P63], [P64], [P68], [P75]	11	13.25%
Case Studies with Experiments	[P4], [P5], [P8], [P16], [P17], [P27], [P42], [P47], [P77], [P78], [P79]	11	13.25%
Ground Theory	[P62], [P70], [P71], [P76], [P81]	5	6.02%
Total		**83**	**100.0%**

TABLE 12: Quality Attributes Reported in PSs

PS	A	B	C	D	E	F	G	H	J	K	L	M	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT	AU
[P1]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P4]	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	✓	-
[P9]	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P11]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P15]	-	-	-	-	-	✓	-	✓	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P16]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	✓	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P18]	-	-	-	✓	-	✓	-	✓	-	-	-	✓	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P19]	-	-	-	-	-	✓	-	✓	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P22]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P26]	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓
[P38]	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P41]	-	-	-	-	-	✓	-	✓	-	-	-	-	-	-	-	✓	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P43]	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P44]	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	-	-
[P48]	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P50]	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P51]	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P52]	-	✓	-	-	-	✓	-	✓	✓	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	✓	-	✓	✓	✓	✓	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P53]	-	-	-	✓	-	-	-	✓	✓	✓	✓	-	✓	✓	-	✓	-	-	-	-	✓	-	✓	-	✓	-	-	-	-	✓	✓	✓	✓	✓	✓	✓	-	-	-	-	-	-	-	-	-	-
[P54]	-	✓	-	-	-	✓	-	✓	✓	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	✓	-	✓	✓	✓	✓	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P60]	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P62]	-	-	-	-	-	✓	-	✓	✓	-	-	✓	-	-	✓	-	-	-	-	-	-	✓	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P63]	-	-	-	✓	-	-	-	✓	✓	✓	-	-	-	-	✓	-	-	-	-	-	✓	-	-	-	✓	-	-	-	-	✓	✓	-	✓	✓	✓	✓	✓	✓	-	-	-	-	-	-	-
[P65]	-	-	-	-	-	-	-	✓	✓	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	✓	✓	-	-	-	-	-	-	-	-	-	-	✓	-	-	✓	-	-	-	-	-	-	-
[P66]	-	-	-	-	-	✓	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-
[P67]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	✓	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P69]	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P71]	-	-	-	-	-	✓	-	✓	-	-	-	✓	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P74]	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[P75]	-	-	-	✓	-	-	-	✓	-	-	-	✓	-	-	✓	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-
[P78]	-	-	-	-	-	✓	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	✓	✓	-	-	-	-	-	-
[P80]	-	-	-	-	-	✓	-	✓	-	-	-	✓	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-
[P83]	-	-	-	-	-	-	-	-	-	-	-	-	✓	-	✓	-	✓	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Total	1	4	1	6	1	17	3	17	7	1	1	13	3	1	10	9	2	1	2	3	1	4	4	4	2	2	2	2	2	2	2	1	2	3	3	2	1	3	1	1	1	1	2	1	1

✺ **A**: Accessibility; **B**: Availability; **C** :Device lifespan Expectancy; **D**: Energy Efficiency; **E**: Installability; **F**: Interoperability; **G**: Maintainability; **H**: Performance; **J**: Reliability; **K**: Responsiveness; **L**: Satisfaction; **M**: Scalability; **O**:Timeliness; **P**: Testability; **Q**: Usability; **R**: Conformance; **S**: Completeness; **T**: Suitability; **U**: Accuracy; **V**: Correctness; **W**: Plausibility; **X**: Robustness; **Y**: Connectivity; **Z**: Compatibility; **AA**:Sensitivity; **AB**: Relevance; **AC**: Stability; **AD**: Mobility; **AE**: Flexibility; **AF**: Extendibility; **AG**: Effectiveness; **AH**: Cost Efficient; **AI**: Portability; **AJ**: Functional Suitability; **AK**: Verifiability; **AL**: Data Integrity; **AM**: Reusability; **AN**:Validity; **AO**: Regulatory Compliance; **AP**:Safety; **AQ**: Integration; **AR**: Resiliency; **AS**: Distributivity; **AT**: Fault-Tolerance; **AU**:Dynamicity

✺ ✓: Discussed; -: Not discussed.

- **Completeness**. The sensor data completeness refers to the degree to which sensor data values are not missing for a given time window [P16].
- **Concordance**. Concordance refers to a measure used to describe the level of agreement or alignment between the information provided by a specific data source and the information obtained from other independent data sources that report related or correlated effects [P83].
- **Conformance**. The degree to which a system adheres to and meets the explicit requirements, specifications, or standards set forth by relevant authorities, industry norms, or regulatory bodies.
- **Connectivity**. As suggested by authors in[47], we can define connectivity in IoT as the system's ability to establish and maintain reliable and efficient connections between its various components and devices.
- **Cost Efficient**. The ability of the system to achieve optimal functionality while minimizing resource usage.
- **Data Integrity**. The degree to which data remains accurate, consistent, and unaltered throughout its lifecycle, safeguarding against errors or corruption.
- **Device Lifespan Expectancy**. As suggested by the authors in [48], we can define IoT device lifespan expectancy as the period during which the device is expected to remain operational and perform its intended functions reliably before it may need replacement or significant maintenance.
- **Distributivity**. Distributivity refers to the characteristic of an IoT system in which its components, located across networked devices, possess the capability to effectively and efficiently communicate, coordinate, and exchange data through message passing. This allows the system to appear to its users as a single, coherent system [49], [50].
- **Dynamicity**. Dynamicity refers to the property of the system's architecture that allows components and connectors to be created, interconnected, or removed during the system's operation. It signifies the system's ability to adapt, evolve, and reconfigure itself in real-time, often in response to changing conditions or requirements, without the need for significant disruptions or downtime [51].
- **Effectiveness**. Effectiveness is a measure of how well the system achieves its outcomes.
- **Energy efficiency**. Energy efficiency is the ability of IoT system to achieve the desired results or functionality using less energy resources [52].
- **Extendibility**. The degree to which a system can be expanded or enhanced to accommodate new features, functionalities, or changes.

- **Flexibility**. The system's capability for devices to undergo changes in both hardware and software configurations, as well as adaptations in their waveform [P54].
- **Mobility**. Capability of devices or nodes within the system to move or change location while maintaining seamless connectivity and functionality.
- **Plausibility**. Plausibility refers to a measure that determines whether the information received from a data source aligns with the probabilistic knowledge of what it is supposed to measure, evaluating whether the received data makes sense about its expected purpose [P83].
- **Regulatory Compliance**. The extent to which a system adheres to and satisfies the established laws and regulations relevant to its operations.
- **Relevance**. The degree to which the data collected and processed by the system aligns with the intended goals and requirements.
- **Resilience**. The capability of IoT systems to withstand disruptions and crises, recover from emergencies and near-catastrophes, and adapt effectively to a dynamic and ever-changing environment.
- **Robustness**. The degree to which a system or component can function correctly in the presence of invalid inputs or stressful environmental conditions.
- **Satisfaction**. The extent to which user needs are satisfied when the system is used in a particular operation [P1].
- **Responsiveness**. The system's ability to handle a request within a specified or required time interval [P1].
- **Safety**. The extent to which a system minimizes the risk of harm to users during its operation or in the occurrence of unexpected events.
- **Scalability**. System's ability to handle and accommodate an increasing number of devices efficiently, without compromising performance, reliability, or responsiveness.
- **Sensitivity**. The degree to which the output changes with respect to change in input parameters [P52].
- **Stability**. Stability in an IoT system refers to the consistent and reliable output of sensors over time [P52].
- **Suitability**. The extent to which a sensor is appropriate or well-suited to fulfill the specific needs and demands of a particular application. The sensor suitability defines whether the data value expected from the application lies within the measuring interval of the sensor [P16].
- **Testability**. The extent to how easy is to design and conduct tests for the system [P1].
- **Timeliness**. The ability of the system to deliver and process data at the right time.
- **Validity**. The accuracy and correctness of the data collected by sensors.
- **Verifiability**. System's capability to provide evidence or confirmation that it behaves and performs as specified.

Lessons from RQ1

The top test objectives are interoperability, performance, scalability, conformance, usability, reliability, and energy efficiency. Specific objectives for IoT system testing include connectivity, energy efficiency, device lifespan expectancy, distributivity, and dynamicity.

4.5 RQ2: What Are Testing Approaches Investigated?

In [6], the authors proposed a framework for developing IoT systems. The framework consists of five layers. The device layer and the application layer require five phases: analysis, design, coding, testing, and installation. The platform layer (cloud layer) encompasses four phases: design, coding, testing, and installation. The infrastructure layer (network) involves the analysis phase. In our study's second and third research questions (RQ2 and RQ3), we used these phases to elucidate the specific phases where the approaches and tools are used.

According to the International Organization for Standardization (ISO), a testing approach *"is a high-level implementation choice of a testing strategy"* [34]. It defines how testing will be carried out and includes various elements, such as level of testing, type of testing, technique of testing, and testing practices. ISO also defines the testing level as *"one of a sequence of test stages, each of which is typically associated with the achievement of particular objectives and used to treat particular risks"* [34]. The following are common levels of testing: unit/component testing, integration testing, system testing, system integration testing, and acceptance testing. ISO further defines testing type as a specific category of testing focusing on specific quality attributes, such as functional testing, usability testing, and performance testing. Testers can carry out testing activity at a specific level or across multiple levels. For example, conducting performance testing at both the unit test level and the system test level. The testing technique is defined as *"a procedure used to create or select a test model, identify test coverage items, and derive corresponding test cases"*[34]. For example, software testing uses specific test techniques such as equivalence partitioning, boundary value analysis, decision table testing, and branch testing. In this section, we discuss testing approaches (testing levels, testing types, and testing techniques and practices) investigated in PSs.

We identified several testing levels in PSs. Table 13 provides an overview of these testing levels, which are further explored in this section. Unit testing plays a role in testing IoT systems by isolating individual components, such as sensors, smart objects, communication layer, and application, testers can trace and identify the root causes of system failures [P65]. Unlike integration testing in traditional software systems, which focuses on testing the interaction between different modules, integration testing in IoT systems focuses on testing the interfaces between IoT components or layers, such as smart objects (IoT devices) and gateways [P64], [P75].

System testing is used to *"test the system with its actual operational environment with different scales and scenarios"* [P64]. It ensures that the system's functionality and performance meet the desired requirements. System testing includes testing the system as a whole rather than individual components or layers. If the real-world environment is not available, simulation tools can be used to replicate the testing environment.

Acceptance testing, which is user-centric, focuses on evaluating IoT systems from the end-user perspective. It tests the services provided with user involvement [P64]. The primary objective is to verify that the system delivers the expected

TABLE 13: Testing Levels

Category 3: Testing Levels			
Name	**Focus**	**Phase**	**Contribution**
Unit Testing [P64], [P75]	Individual components	C, T	Identification of part of the system causing the failure.
Integration Testing [P64], [P75]	Component integrations	T	Testing the interface between various IoT components.
System Testing [P64]	Entire system	T	Use of simulation tools to simulate the environment.
Acceptance Testing [P8], [P47], [P79]	User-system interaction	T	Use of existing testing tools: Appium, Sikuli, etc.

✺ **C**: Coding; **T**:Testing.

services in the user environment. Analysis of PSs revealed that some approaches involve testing multiple levels [P64], [P65], [P75], while others focus on one level, such as integration testing, which examines the interface between various layers to ensure the proper functioning of IoT systems or acceptance testing [P8], [P79] for validation.

We also identified several testing types in PSs. Table 14 provides the summary of identified testing types.

The analysis revealed few testing types in the PSs. Maintainability testing and portability testing are absent in the PSs. Certain testing types specific to IoT systems, such as connectivity testing and distributivity testing, are also missing. Among the discussed testing types, interoperability testing was the most prominent (six PSs). Conformance testing ranked second (three PSs). Functional testing, performance testing, and scalability testing received attention in two PSs each. Other testing types like compliance testing[5], data integrity testing, compatibility testing, reliability testing, and usability testing are addressed in one PS each.

Finally, we identified many testing techniques considered for IoT systems. Table 15 shows the summary of identified testing techniques. The authors of [53] proposed a list of software testing techniques that can be leveraged for testing the application layer of IoT systems. The author of [29] presented different testing techniques specifically tailored for IoT systems. Automation testing is the most researched technique (6 PSs), but full automation of IoT systems testing remains elusive and debatable. IoT system test automation currently focuses on semi-automation, where some layers are tested manually. Three PSs discussed fault testing for anomaly detection. Three other PSs discussed fuzzy testing, which involves generating large volumes of random test data. Two PSs covered pattern-based testing, two PSs addressed standard-based testing, two PSs explored verification and validation, two PSs examined simulation-based testing, and two PSs investigated coverage-based testing. We identified one PS for each of the following testing categories: formal verification, continuous testing, requirement-based testing, data validation testing, log analysis, and knowledge-based testing. We considered *fuzzy testing, automated testing, and model-based testing* as testing practices [29], which are also part of testing approaches.

5. *Conformance* and *compliance* are often used interchangeably, but they are slightly different. **Conformance** refers to adhering to a specific standard, specification, or set of requirements. **Compliance** goes beyond conformance and involves meeting legal, regulatory, or industry-specific obligations.

Lessons from RQ2

Approaches for interoperability testing are prominent, followed by conformance and functional testing. Automated testing is well-researched, but there are no fully automated approaches.

4.6 RQ3: What Are Testing Tools Investigated?

Testing tools in software engineering are software applications that automate or facilitate the testing process. The purpose of these tools is to help the testers verify various aspects of IoT systems, such as the functional correctness and performance of IoT systems. They can simulate various scenarios and environments to validate the behavior of IoT systems under different conditions. Testing tools help to ensure the reliability and quality of IoT systems. We analyzed PSs to identify the tools proposed for testing IoT systems. We identified two categories of tools: *test execution tools* and *test environment tools (or testbeds)*. Test execution tools help to execute test cases, while test environment tools provide the environment for testing.

Table 16 reports the summary of test execution tools in PSs. We faced challenges in collecting detailed information about many tools. In addition to the data from PSs, we consulted the official websites (if available) of the tools and reached out to the authors of PSs who discussed the tools in order to obtain more information. However, many of them did not respond.

Various tools are proposed, including traditional software testing tools and tools specifically designed for IoT systems. Commonly used tools for traditional software testing, such as Selenium, Robot Framework, Apache JMeter, and Stryker Mutator, are also used in testing IoT systems.

We also identified tools specifically designed for IoT systems testing, including Héctor [P26], PatrIoT [P50], F-Interop [P18], ICAT [P29], IoT-CIRTF [P3], Smartesting Certifyit [P48], SemTest [P15], Tsung [P51], and Eclipse IoT Testware [P67]. These tools target various qualities of IoT systems testing, such as performance, semantic verification, interoperability, and compatibility. However, some of these tools developed by the authors of PSs, such as MATTER and Izinto, are not publicly available to end-users. Testing as a service (IoT-TaaS [P38]) has emerged as a novel approach for IoT systems testing, allowing organizations to outsource testing activities and access testing resources on demand. This approach alleviates the need for costly in-house testing infrastructure and offers a more cost-effective solution.

Although we identified several tools, their application to IoT systems testing is limited to a few qualities, such as

TABLE 14: Testing Types

Category 2: Testing Types			
Name	**Focus**	**Phase**	**Contribution**
Performance Testing [P65], [P75]	Different application domains	Analysis	Testing in health and medical, smart cities, and agriculture domains. [P75]
Functional Testing[P23], [P29]	Device testing	Testing	FSM-based test cases [P23] Compatibility testing tool [P29]
Conformance Testing [P11], [P49]	Client environment dependencies	Analysis	Testing interface between components [P11] ML-based detection for reactive system failures [P49]
Scalability Testing [P65], [P38]	Adapter for protocols	Design	Validation and verification based on test report
Compliance Testing [P31]	OneM2M library	Design	Inter-component communications
Interoperability Testing [P18], [P19], [P50], [P69], [P80], [P83]	Integration	Analysis Design	Testing interface between components Compatibility among packets and input used
Data Integrity Testing [P65]	Definitions	NP	Data integrity in IoT systems
Compatibility Testing[P65]	Definitions	NP	Compatibility in IoT systems
Reliability Testing[P65]	Definitions	NP	Reliability in IoT test environment
Usability Testing[P65]	Definitions	NP	Usability of IoT systems for different users

✷ **NP**: Not Provided.

scalability and connectivity. Our findings emphasize the need for more comprehensive testing tools that encompass a wider range of qualities of IoT systems. Table 17 reports our findings concerning infrastructure testing tools (testbeds), including their domains, programming languages, software implementation, and target layers.

Several testbeds support multiple programming languages ([P4], [P10], [P13], [P20], [P43], [P45], [P61], [P66]), with Java being the most popular (used in 8 testbeds), followed by C, C++, and Python (used in 3 testbeds). *Testbeds As A Service* (TaaS) ([P82], [P43], [P35], [P21]) offer cost-effective testing services. It enables users to request testbeds and necessary resources, submit testing details, and download testing reports upon completion, leading to shortened test environment setup time. While most testbeds focus on device and network layers ([P82], [P44], [P43], [P35], [P26]), some testbeds support three layers, including device, cloud, and application layers ([P4], [P10], [P21]). ContikiOS is commonly used in testbeds ([P35], [P44], [P66]), followed by Kafka [P4], [P26]. One PS reported the use of a proprietary graphical operating system for users to access and configure IoT devices through a web-based interface [P10]. We did not find hardware and software specifications of certain testbeds, particularly for cloud-based testbeds [P13], [P21] or for on-premise installation testbeds [P36], [P45], [P46].

For improved clarity and ease of replication, Table 17 reports the names of the testbeds, clickable repository links, and hardware specifications if available. Our analysis revealed that existing testbeds mainly focus on specific layers of IoT systems, particularly network (13 PSs) and device layers (11 PSs). Most discussed testbeds are cloud-based (10 PSs), providing scalability, flexibility, and cost efficiency. There is a significant emphasis on IoT testbeds for smart cities, with five testbeds identified [P4], [P13], [P45], [P46], [P61]. Seven testbeds are generic and applicable to any application domain.

The lack of hardware and software specifications for certain testbeds, particularly cloud-based and on-premises installations, may hinder transparency and replication of experiments. Existing testbeds mainly focus on specific IoT layers, such as network and device layers, potentially leaving other layers underrepresented in testing. Cloud-based testbeds dominate the landscape, favored for their scalability, flexibility, and cost efficiency. Moreover, the emphasis on IoT testbeds for smart cities may highlight the significance of urban applications. However, the relatively small number of generic testbeds may indicate a need for more versatile solutions applicable across various IoT domains. Addressing these implications could enhance the effectiveness and inclusivity of IoT systems testing efforts.

Lessons from RQ3

Several tools for traditional software testing, such as Selenium, Robot Framework, Apache JMeter, and Stryker Mutator, are used in IoT systems testing. Specific tools for testing IoT systems including PatrIoT, ICAT, IoT-CIRT, and Héctor, are found in PSs.

4.7 RQ4: What Are IoT Systems Used For Evaluation?

We compiled a catalog of diverse IoT systems used to evaluate various approaches proposed in PSs. Each IoT system has a distinct focus and may include different components, covering areas like Ambient Assisted Living (AAL), smart street systems, smart parking, smart manufacturing, smart agriculture, and more. Some are mobile apps for Android and iOS, while others provide data through web interfaces. The source code for the evaluated systems is not publicly available, except for *DiaMH*. Table 18 shows several systems identified from PSs.

We noticed that some authors did not specify the name of the system they used in their experiment or did not make their source code publicly available. They simply mentioned that they conducted experiments, without further details. The absence of such details may impact result reproducibility and potentially hinder the progress of other researchers. We believe that sharing this information can be beneficial to other researchers in the field of IoT systems testing.

TABLE 15: Testing Techniques and Best Practices

Category 1: Testing Techniques and Practices					
Name	**Focus**	**Phase**	**Strengths of Existing Works**	**Weaknesses of Existing Works**	**Contribution**
Fuzzy Testing [P24], [P27], [P55]	Network protocol	Design, Coding	Unlocking the potential of AI algorithms in testing	Not suitable for E2E testing	AI guided fuzzing for testing IoT protocols and applications
Data Validation Testing [P60]	Data validity testing	Testing	It supports abstract-level testing	Not suitable for complex scenario testing	Error detection in IoT data
Log Analysis [P81]	Bug detection	Design	It supports log mining	Not suitable for cloud-based applications	Log analysis for different events
Knowledge-Based Testing [P59]	Knowledge Mining	Design	Leveraging ML algorithms to improve IoT systems testing	Needs a lot of processing resources	Abnormal sensor detection
Pattern Based Testing [P42], [P58]	Event pattern identification	Design	Generic and validated IoT test patterns	Not suitable for fuzzy logical systems	Generic patterns-based testing framework ML-Based sensor abnormality detection
Coverage Based Testing [P14], [P18]	Smart contracts	Design	It offers Testing as a Service	Limited to commercial apps only	Path selection for device testing Challenges for online testing
Simulation-Based Testing [P70], [P73]	Devices	Coding	No requirements for real devices	Requires real-time user interaction	User-interaction simulation-based testing
Fault Testing [P37], [P48]	Anomaly detection	Testing	Using ML algorithms for complex cases	Needs a lot of processing resources	Track faulty devices IoT testing as service
Automated Testing [P11], [P23], [P28], [P31], [P38], [P50]	Network and devices	Design	It uses an intermediary model to validate IoT services	Not suitable for E2E testing	Interface testing between components Test cases generation Interoperability testing
Verification and Validation[P19], [P30], [P78]	Scenario analysis	Analysis	It supports textual notation and UML modelling	Requires clear criteria for test cases	Validation criteria for interoperability testing Textual description of test case
Requirement Based Testing [P32]	Network	Analysis	Experimental results	Not suitable if requirements are not clear	Testing specific parts of the network based on specific application protocols
Standard Based Testing [P67], [P76]	Network Protocols	Analysis	Implementation-Independent	Not suitable if requirements are not clear	Use of EclipseIoT for testing protocols Test execution based on TTCN3 standard
Continuous Testing [P12]	DevOps	All	It proposed best practices based on open sources tools	Suitable for simple scenarios only	Development of IoT applications
Formal Verification and Model-Based Testing [P30]	Formal verification and MBT	Design	It supports abstract level testing	Supports only partial user scenario testing due to resource constraints	Verification and testing for smart cities
Model-Based Testing[P30], [P48], [P49]	Testing as a service	Design	It enables testing IoT data and platforms	Limited use case scenarios	MBTAAS for IoT Platforms Detecting failures in reactive systems

- **MBTAAS**: Model-Based Testing As A Service; **MBT**: Model-Based Testing; **E2E**: End-to-End.
- **Level of Automation**: None of the proposed techniques or approaches are fully automated. Most of the techniques and approaches suggested are *semi-automated*, requiring some degree of human intervention. However, *requirement-based testing, formal verification, and standard-based testing* approaches are conducted *manually*. *Continuous testing* suggests the use of various tools to automate the testing process; however, no empirical study has been provided.
- **Targeted Application Domain**: Most of the approaches and techniques proposed in PSs are generic and can be applied to various IoT systems. However, *formal verification and Model-Based* testing is discussed specifically in the context of Smart Cities, with a case study involving a *Temperature Measuring System* composed of one collector and four sensors. The remaining techniques and approaches discussed are generic.

Lessons from RQ4

Several IoT systems have been used to evaluate the proposed approaches. Most of these systems are not publicly available, thus affecting the reproducibility of the results reported in PSs.

4.8 RQ5: What Are Testing Challenges Identified?

We identified the following testing challenges in the PSs.

❶ *Large number of heterogeneous devices* [P14], [P65]. If IoT systems have hundreds, or even thousands, of heterogeneous devices, a challenge is identifying which devices should be tested to detect faults. Another challenge is testing *different communication protocols*. Some devices face power issues because the *availability of energy and network* cannot be guaranteed [P34].

❷ IoT devices generate *data in different formats*, making it difficult to create a universal method for collecting and analyzing data from devices [P65]. The *lack of APIs for some IoT devices* can also pose a challenge for collecting and analyzing data in IoT systems.

❸ Access to *real devices* to test IoT systems, reproducing IoT bugs, fault localization, and testing *diverse technologies* are also challenges [P33], [P65].

❹ *Limitations of existing tools, and approaches*, in testing some aspects of IoT systems [P70].

❺ The *tight coupling between hardware and software* creates a testing challenge, as defects in either component can impact the overall functionality of IoT systems [P65].

❻ Testing *non-standard compliant devices* is challenging be-

TABLE 16: IoT Systems Testing Tools Identified in PSs

Name	Focus	Language	IoT Layer	Testing Method	API	Testing Level	Technologies	SDLC Phase
Héctor[P26]	Automated testing	Python	Devices Network	White-box	REST	Integration System	Apache Flink Kafka	Testing
Node-RED [P47]	Devices Cloud	Node.js	Application	Black-box	REST	Acceptance System	MQTT, CoAP	Design
Apache JMeter [P47]	Load and performance test	Java	Application	-	SOAP, REST	Unit	-	Coding
Eclipse IoT-Testware [P67]	Standardized Abstract Test Suite (ATS)	Java	Multiple layers	White-box Black-box	-	-	CoAP, MQTT	Design
PatrIoT [P50]	Distributed IoT applications	Java	Device Network	-	REST	Integration	-	Testing
Tsung [P51]	Stress testing	Erlang	-	-	SOAP	-	CoAP, MQTT, WebSocket	Testing
SemTest [P15]	Semantic compliance and interoperability testing	-	Application	Black-box	-	-	CoAP, AMQP	Testing
MATTER [P47]	Executable test cases	Python Java	Application	Black-box	-	-	MQTT, TCP	Design
F-Interop [P18]	Interoperability performance	-	Device	Black-box	REST, Rspec (XML based)	System Acceptance	AMQP, CoAP, MQTT, 6TiSCH	Testing
IoT-TaaS [P38]	Testing-As-A-Service	-	Device	Black-box	-	Integration	CoAP, HTTP, MQTT	Testing
Izinto [P42]	Pattern-based test automation framework	Java Node.js	-	-	-	-	MQTT, REST API	Testing
ICAT [P29]	Compatibility testing	-	Device	Black-box	-	-	Bluetooth, WiFi	Testing
Robot Framework [P29]	Test automation and robotic process automation	Python	Application	Black-box	REST API	Acceptance System	-	Testing
Selenium [P12]	Test automation	Java, JavaScript, Python, Ruby, C#	Application	Black-box		Acceptance System	-	Testing
IoT-TEG [P39]	Test event generation	EPL, Java 8	Device	Black-box	REST API	Unit	Websockets	Testing
Smartesting CertifyIt [P48]	Model-based testing for enterprise IT	UML, OCL, BPMN	Application	Black-box	-	-	-	Design
Stryker Mutator [P77]	Mutation generation	JavaScript TypeScript C# , Scala	Application	Black-box	-	-	-	Design
IoT-CIRTF [P3]	Prioritizing and selecting test cases	Java, Python, XML	Device	White-box Black-box	-	Integration	HTTP, MQTT, WiFi, Zigbee	Coding
VectorCAST [P65]	Embedded systems	C++, C	Device Application	White-box Black-box	-	Unit Integration	-	Coding

✺ -: Information not found.
✺ **SDLC**:System Development Life Cycle.

cause they do not adhere to standard protocols and specifications [P67], and cannot be tested in standardized scenarios.

❼ *Testing in real-world scenarios* with the deployment of real devices is costly [P70], [P72].

❽ *Lack of testing methodologies* introduces a challenge to compare different IoT devices or to develop testing frameworks that apply to all IoT devices [P72].

❾ Testing IoT is challenging when the devices behave differently in *non-repeatable scenarios* because of environmental changes or unpredictable device behavior, leading to unpredictable results and difficulty in identifying issues [P72].

❿ *Lack of uniform communication interface and synchronization* in a hybrid environment leads to another challenge. The system under test cannot distinguish between simulated and real-life behavior. Synchronization of testing involves combining data generated by simulated entities and data generated by real entities [P73].

TABLE 17: Comparison of IoT Testbeds in PSs

Name	Features	ES	IAL	API	PL	IS	Software	TD	Specifications
Two-Tier Fog Testbed [P4]	Handling streaming data efficiently. High availability and fault tolerance	LS	Device Cloud	NP	Java Scala Python R	CB	Kafka Hadoop Spark	SC	1 GB memory for Raspberry Pi 3B and 4 GB for Pi 4B, with peripheral memory up to 32 GB, high-speed ethernet, and deployment configurations for streaming and analytics systems. WiFi and LAN support.
AssIUT IoT [P10]	Enable user access and configuration of IoT nodes through GUI	SS	Device Network Cloud	NP	C# VB.NET F#	CB	WS IIS ASP.NET MS SQL Server	ED	1.DHT-11 sensor Arduino mega 2560 Rev3, WiFi (ESP8266-12E module) and Zigbee
FIESTA-IoT [P13]	Fault-tolerant, federated experiments	LS	Network	Yes	C Java Python	CB	NA	SC	NA
FIT IoT-Lab [P20]	Large-scale open access IoT testbed, Free-of-charge access to thousands of wireless IoT devices, online bug reporting	LS	Network	Yes	Java	CB	Embedded OS	SF	WSN430 (16-bit MCU, TI CC2420/CC1101 radios, ambient sensors), M3 (ARM Cortex M3, atmospheric sensors, accelerometers), and A8 (ARM Cortex M3).
FogTestBed [P21]	Request resources and submit experiment online	SS	Device Cloud	Yes	Java	CB	NA	GEN	NA
HATBED [P25]	Affordable with remote debugging and flexible, non-invasive software profiling.	LS	Device Network	NA	NA	OP	OpenOCD Sigrok	GEN	RPi 3B+ includes a 1.4GHz quad-core processor, dual-band wireless, and USB ports, along with additional components: FT2232HL for USB-to-UART, UM220-III GPS module, and CY7C68013A microcontroller.
IoT Bed [P35]	Cost effective for users	LS	Device Network	Yes	Java	CB	ContikiOS Cooja	GEN	NA
IoTier [P36]	Easy integration with container orchestrators	SS	Device Network	NA	NA	OP	NA	GEN	NA
JOSE [P43]	Setup can be completed in a very short time	LS	Network Device	Yes	Java, C JavaScript	OP	NA	GEN	50 Storage and 1200 Computation Servers
LinkLab [P44]	scalable, multi-site, and multi-user support, remote development	LS	Device Network	NA	Python	CB	Alibaba Cloud	ED	The setup includes 40 Arduino Mega 2560 with various Grove sensors, 10 Arduino Uno R3 with Dragino LoRa Shield, 50 ESP32 with temperature/humidity sensors, 1 RPi 3B+, 5 RPi 3B+ with Grove sensors, 1 RPi 3B+ with LoRa Gateway Shield, and 50 TelosB with Zigbee and ContikiOS.
LocURa4IoT [P45]	Flexible indoor localization for experiments	SS	Network	Yes	C/C++	OP	NA	SC	NA
EAWN Testbed [P46]	Enable energy-aware protocol testing in real devices	SS	Network	NA	C++	OP	NA	SC	NA
Smart Santander [P61]	Guaranted dependability	LS	Application Network	Yes	Java JavaScript	CB	NA	SC	PCs, 2 Xbee-Pro radio modules, WiFi, 3G, Bluetooth, and Ethernet interfaces; Sensor nodes based on the ATmega1281 microcontroller with 8 KB SRAM, 4KB EEPROM, 128KB FLASH and 2GB SD extra storage; 7 analog and 8 digital interfaces for I/O sensor connection, plus 1 PWM, 2UART, 1 I2C, and 1 USB interfaces
SD IoT Testbed [P66]	Simulation environment for real-world application testing	SS	Application Device	Yes	Python C++, Java Node.js	CB	ContikiOS TinyOS FreeRTOS	EM	RPi, Intel Galileo board, BME280 sensor, Arduino UNO
UiTIoT [P82]	Reliability, availability, effectiveness	LS	Device Network	Yes	XML	CB	OpenStack DeltaQ QOMET	GEN	RPi board with a 900MHz quad-core ARM Cortex-A7 CPU and 1GB LPDDR2 SDRAM; TP-LINK portable plug-and-play wireless adapter

✺ **EC**:Evaluated Scale, **IOL**: IoT Architecture Layer, **API**: Application Programming Interface, **PL**: Programming Language, **IS**:Installation Setup, **TD**: Target Domain; **WS**: Window Server.
✺ **SS**: Small Scale; **LS**: Large Scale; **SC**: Smart City; **ED**: Education; **SF**: Smart Factory; **NA**: Not Available; **GEN**: Generic; **EM**: Environment Monitoring; **CB**: Cloud-Based; **OP**: On-Premise.

Lessons from RQ5

Testing IoT systems has several challenges, including device heterogeneity, difficulties in collecting and analyzing diverse data, tight coupling between hardware and software, non-standard compliant IoT devices, costly real-world and non-repeatable scenarios, and limitations of existing approaches and tools.

4.9 RQ6: How Are Testing Challenges Addressed?

We reviewed the PSs to understand how they addressed IoT testing challenges. Some of the challenges identified can be addressed by some solutions proposed in the PSs. There is no solution proposed to tackle challenges such as testing in real-world scenarios, non-repeatable scenarios, unavailability of devices due to lack of energy or network, lack of APIs for some IoT devices, testing diverse technologies, and tight coupling between hardware and software. We highlighted the addressed test objectives as shown in Table 19 putting

TABLE 18: Systems Used For Evaluation

IoT System Used	Descriptions	Focus	Source Code	Repository
DiaMH [P8], [P47], [P77], [P79]	DiaMH is a Diabetes Mobile Health IoT system that: (i) DiaMH, is a Diabetes Mobile Health IoT system that monitors glucose levels in patients, sends alerts to both patients and doctors when glucose levels go beyond a specified range and regulates insulin dosing. It comprises wearable devices, such as a glucose sensor and insulin pump, connected to smartphones that act as intermediaries to a cloud-based healthcare system. This cloud-based system processes data and provides valuable information, including alerts and insulin dosage recommendations.	E2E	Available	Repo
Care Receiver Monitor in AAL [P42]	In this testing scenario, a care receiver is observed in a controlled environment, with a caregiver responsible for monitoring. Various sensors collect and transmit health and environmental data to a central server, which maintains a health record. The server triggers predefined actions and alerts, notifying the caregiver as needed. This setup includes body sensors (e.g., blood pressure monitor, weight scale, fitness bracelet), ambient sensors (e.g., temperature, humidity, air quality, luminosity), and actuators (e.g., lightbulb holder, smart socket for the air conditioner) for performing actions and generating alerts.	E2E	N.A	N.A
4 Android apps, 4 iOS apps [P29]	The authors created a scenario comprising one cloud server, one Wi-Fi access point, eight app versions (two versions of Android apps on two different Android OS versions, and two versions of iOS apps on two different iOS versions), four gateway versions, and six sensor versions.	E2E	N.A	N.A
Smart Street System, AMQ Online product [P50]	A smart street system based on an active messaging queue (AMQ Online) for improved urban living, featuring smart lighting, traffic monitoring, and environmental sensors for safer and more efficient city streets	E2E	No	No
GSM [P3]	GSM facilitates efficient connectivity for IoT devices and offers two datasets: one with 80 requirements and 100 TCs for IoT device connection efficiency, and another for Mobile IoT (MIoT) with 51 requirements and 41 TCs.	E2E	N.A	N.A
Sensing App [P6]	Temperature and Light sensors which represent IoT devices that generate very small amounts of data.	E2E	N.A	N.A
Smart Parking IoT System, Smart Manufacturing System [P14]	A smart parking application uses 1840 devices to efficiently manage parking spaces, helping users find available spots and reduce congestion. A smart manufacturing system comprises 161 devices, including sensors and cameras, collaborating to monitor the manufacturing environment. Data from machines, cameras, and sensors is sent to servers. Authorized users access a web server via cellphones or computers to query, update, and remotely control the devices in the factories.	E2E	No	No
Smart Agriculture Applications [P16]	The smart agriculture app utilizes an IoT setup with an LM35 temperature sensor, Arduino UNO, and an internet-connected laptop. The LM35 sensor records temperature data, transmitted to ThingSpeak via Arduino and Python for real-time storage and analysis using MATLAB.	E2E	No	No
Temperature Monitoring System [P30]	The system comprises four sensors that measure ambient temperature and transmit the data to a collector, which stores the values in a database for later use.	E2E	No	No
Smart Home [P37]	*Simulated Smart Home* implemented 11 apps that automate 17 IoT devices in a simulated smart home: ✺ **Motion-Activated-Lights** to turn *on* lights when motion is and *off* when motion is not active. ✺ **Smoke-Alarm** to sound the alarm and unlock the doors when smoke is detected. ✺ **Temperature-Control** to keep the temperature between 70-80 degrees by turning the heater and air conditioner on and off. ✺ **Water-Leak-Detector** to sound the alarm and close the water valve when a leak is detected. ✺ **Welcome-Home** to unlock the doors and turn on the coffee machine when the user arrives home. ✺ **Secure-Patio** to send a text message to the user when a user is not present and contact is detected. ✺ **Energy-Saver** to close the window if the window is open and either the heater or air conditioner is on. ✺ **Secure-Home** to lock doors and close windows when the user is not present at home. ✺ **Intruder-Detector** to send a text message to the user when a user is not present at home and motion is detected. ✺ **Alarm-Safety** to turn on lights when an alarm is activated. ✺ **Morning-Air** to open and close windows at specific times.	E2E	No	No
SeRGIo [P73]	SeRGIo is mobile sensing system that focuses on developing geospatial IoT applications through participatory sensing, involving academic and industrial partners. It aims to gather qualitative data from citizens and workers through distributed sensing tasks.	E2E	No	No
Smart Mobility [P78]	The smart mobility IoT system assesses pollution levels on urban cycling routes using smart bicycles, a central data server, and smart poles.	E2E	No	No
ODAA [P83]	Open Data Aarhus (ODAA) is an open data portal for Aarhus, providing access to datasets, including traffic data from 449 sensors reporting vehicle count and speed in the city	E2E	No	No
Smart Parking Application [P4]	A smart parking application uses technology to efficiently manage parking spaces, helping users find available spots and reduce congestion	E2E	No	No

✺ **GSM**:Global System for Mobile Communications; **AAL**: Ambient Assisted Living.
✺ **E2E**: End-to-End; **N.A**: Not Available.

the most addressed challenges at the top, and the ones that remain mostly unaddressed at the bottom.

Lessons from RQ6

No specific PS focused on a particular challenge for testing IoT systems. However, we found that solutions proposed in certain PSs can partially or fully address some of the identified challenges.

5 Discussions

5.1 Results Overview

This section discusses the results and explores the implications of the lessons learned.

✺ Our findings indicate that researchers have devoted less attention to specific objectives of testing IoT systems such as testing connectivity, resource usage, device life expectancy, correctness, and timeliness. Objectives such as testing interoperability, compatibility, performance, reliability, scalability, verification, data integrity, data validity, usability, regulatory compliance, conformance, integration, and fault tolerance, can be associated with testing approaches. Although some PSs discussed other objectives, they did not mention any specific approach to test them. We observed that functional suitability testing was the only objective assessed through various levels of testing (unit testing, integration testing, system testing, and acceptance testing); however, PSs did not discuss the testing levels for other objectives. We observed that only 14 out of the 47 quality attributes are associated with the testing approaches and techniques discussed in the PSs. We could not find a reason why other quality attributes

TABLE 19: Proposed Approaches for Addressing Some Challenges

Challenge	Potential Solution in PSs	Addressed Objective
Large number of heterogeneous devices [P14], [P65].	Authors in [P79] proposed the acceptance testing approach using a UI at the system level based on the concept of test scenario. They assumed that if the system provides the desired functionality at the system level, the devices work as expected too. in [P14], the authors recommended the use of a combinatorial testing path selection framework for IoT systems, called CT-IoT, that systematically identifies and recommends testing paths in IoT systems for effective testing. In [P39], the authors proposed a solution based on a test event generator to test multiple connected IoT devices and make decisions according to the complex and real data.	Correctness, connectivity, integration, validation, availability, functional suitability, and verification.
Different communication protocols	Authors in [P50], proposed an open architecture of the testing framework to allow the connection of various devices and infrastructural parts by adding proper communication adapters between the test cases and the devices. Additionally, they suggested the addition of new types of simulated devices, which are configurable from predefined building blocks to minimize the concern of different communication protocols.	Connectivity
Limitation of existing tools, approaches, or lack of testing methodologies [P70], [P72]	To overcome this challenge, some authors proposed either online solutions in the form of testing as a service while other authors proposed some specific tools or frameworks for testing IoT systems: [P80], authors discussed how F-Interop can provide an extensive experimental platform for IoT systems to allow online testing. The platform can provide online interoperability, conformance, and performance testing. Authors in [P11], proposed a solution for automated and scalable online conformance testing for IoT applications. Besides this, other authors proposed some tools [P8], [P47], [P77], [P79] and framework [P42], [P50].	Interoperability, conformance, performance, and scalability.
Non-standard compliant devices, different data format [P67], [P65]	Authors in [P67] suggested the use of Eclipse IoT-Testware (standards-based open-source solution) to ensure protocol conformance, robustness, and secure implementation. Authors in [P11], [P31], [P38], suggested the use of oneM2M to test the conformance of IoT devices. They recommended the use of standardized conformance testing mechanisms with automated and scalable conformance testing for IoT applications by introducing a test triggering mechanism based on a standardized test interface. oneM2M is used to verify that IoT devices are implemented correctly with respect to the protocol message format, and its exchange procedures are defined by oneM2M specifications.	Conformance, interoperability, robustness, verification
Lack of uniform communication interface and synchronization in a hybrid environment [P73]	Authors proposed an AI-powered proxy-based solution that could function as a synchronization mechanism [P73]. The proposed synchronization can be implemented over multiple dimensions, but the study discussed synchronization over the time and space dimension. Within the space dimension, authors focused on consistency between data sensed at a specific location, while within the time dimension, authors focused on maintaining consistency between data sensed at a specific time. Thus, they proposed (a) proxy-based synchronization and (b) reducing synchronization requirements by space isolation. They suggested the use of a proxy synchronization method to intercept and forward messages between the real and virtual environments. To reduce synchronization needs, the authors proposed to isolate specific geographic zones for either real or simulated LEs. This eliminates the necessity for an extra synchronization mechanism, but may still result in data variability mismatches at zone boundaries. Further, they proposed a hybrid simulation-based testing approach that can effectively facilitate interactions of local entities.	Data integrity, connectivity, distributivity, scalability, interoperability, compatibility.
Availability of real devices, and testing in real-world scenarios [P33], [P65], [P70], [P72]	No specific solution has been discussed in PSs to address real-world scenario-related challenges. However, for real devices, authors of [P35] proposed a solution called IoTbed that provides IoT devices on-demand, just-in-time, and allows users to only pay for the time devices are used basis. The proposed solution presents an economic model that offers monetary incentives to the device owners if their devices are used in the testbed experiment. This can attract other testbed providers to rent out their smart devices through IoTbed.	-
Non-repeatable scenarios [P72]	No specific solution was identified in PSs to address non-repeatable scenarios resulting from the diverse dynamics of IoT systems. However, some studies [P48], [P49] proposed the use of model-based testing while others [P56] proposed a runtime verification of IoT systems using complex event processing.	Verification
Availability due to Energy and network [P34]	No PS has discussed a solution for the energy challenge causing device unavailability. For Connectivity, authors in [P50] suggested a solution to test the connectivity of various IoT devices.	-
Lack of APIs for IoT devices	No PS has discussed a solution to the lack of APIs for IoT devices.	-
Testing diverse technologies [P33], [P65]	No PS has discussed a solution for testing diverse technologies	-
Tight coupling between H/W and S/W [P65]	No PS has discussed a solution to the tight coupling challenge between hardware and software in the IoT context.	-

✺ **LE**:Local Entity; **S/W**:Software; **H/W**: Hardware.

are not associated with testing approaches. However, we believe that there could be three possible reasons. One possible reason could be the rapid growth of the IoT domain. Keeping up with this rapid growth can make it difficult for researchers to focus on all testing objectives simultaneously. It is possible to expect other studies to focus on techniques and approaches for other objectives. Another reason could be the researchers' priorities. If specific objectives are considered more critical by researchers, testing studies may be biased toward these objectives. A last possible reason is that some of these objectives pertained to traditional software systems and can still be applied in the same way to IoT systems. Our findings highlighted research opportunities to explore and develop testing approaches for the overlooked objectives.

✺ Testing approaches encompass testing levels, testing types, testing techniques, and industry practices. We identified that all the levels of testing found in traditional software systems apply to IoT systems, although the context may exhibit slight variations. In IoT systems, unit testing may also involve the testing of each IoT layer in isolation. This shows the distinct nature of this approach in the context of IoT. Comparing the recommended testing types, techniques, and practices for IoT systems [29], we can suggest that further exploration is necessary to investigate how to apply these types, techniques, and approaches to IoT systems.

✺ While observing various testing tools, we found that some are tailored to different levels of testing. Nevertheless, when considering testing types, techniques, and industry practices, the availability of such tools is relatively scarce. Among the observed testing tools, Selenium, Robot Framework, and Apache JMeter, widely used in traditional software systems have been adopted to test the application layer of IoT systems. In terms of testbeds, many of them are used for testing the device layer and network layer. Conversely, test-environment tools for the application layer are scarce.

✺ Many challenges identified are specifically related to testing the device layer. The lack of standards could be one of the causes of these challenges. If IoT devices adhere to the same standards, it would address various issues such

as different data formats, device APIs, communication interfaces, etc.

5.2 Implications for IoT Researchers

The lessons learned from this study indicate that testing IoT systems is a new and growing field. The number of PSs has been growing since 2013, with most of them being published in 2018. However, the number of PSs has been decreasing since then. This decrease could be attributed to the effects of COVID-19, making it difficult for researchers to conduct some experiments, especially in fields such as agriculture or transport. Most active researchers in PSs are industry-based rather than academia. The industry has more resources available for research activities related to IoT systems. To ensure that their IoT systems are reliable, secure, and meet customer expectations, the industry can be motivated by commercial pressures.

PSs predominantly emphasized device testing over other layers of IoT systems. Experiments are the primary evaluation method used in IoT systems testing studies. PSs that used experiments provided understandable results supported by factual evidence. PSs highlight various quality attributes in IoT systems that are also used in traditional software systems. PSs also report new quality attributes for IoT systems such as connectivity, energy efficiency, device lifespan expectancy, distributivity, and dynamicity. Traditional software testing approaches are used to test some quality attributes of IoT systems, with interoperability, conformance, and functional testing being prominent. Additionally, testing tools for testing traditional software systems, are used in IoT systems testing. However, testing IoT systems still poses certain challenges, offering new avenues for exploration.

5.3 Implications for IoT Practitioners

PSs reported that IoT systems testing focuses more on interoperability, performance, scalability, conformance, usability, reliability, and resource utilization. Moreover, on top of traditional quality attributes [46], PSs also reported new quality attributes like connectivity, energy efficiency, device lifespan expectancy, distributivity, and dynamicity. Practitioners can use traditional software testing approaches and tools to test those attributes common to traditional software systems. However, future research may develop specialized approaches and tools tailored to IoT systems testing, addressing the unique attributes of these systems.

5.4 Limitations of this Study

There are several limitations to our study. We focused on the testing objectives, approaches, tools, and challenges discussed in the published studies between 2012 and 2022. We acknowledge that IoT systems testing is extensive and evolving. Many other testing approaches (testing types, techniques, and practices) may exist beyond the ones discussed in PSs. We did not address security testing approaches and tools in our research, as we decided to study security testing as a separate topic of its own SLR. Additionally, the tools we identified do not include emulators and simulators, as recent studies [43], [44] have focused on them.

5.5 Discovering Connections and Key Observations

Figure 5 connects our results. The key observations are:

- Out of the 47 possible objectives identified in PSs, only 14 objectives are addressed by the testing approaches found in PSs. Among the remaining 33 objectives, there is no discussion in PSs on how to address them for IoT systems. We found no information explaining how the proposed testing approaches and tools achieved the remaining test objectives. Although many of them can be handled similarly to traditional systems, specific aspects like connectivity, distributivity, dynamicity, and device life expectancy need attention, as they are unique to IoT systems and not be addressed by traditional approaches.
- Connectivity is crucial for any IoT system. Testing for connectivity and distributivity should be regarded as essential testing types for IoT systems; however, they are not present in PSs. Moreover, for certain testing types like scalability, no tools are found in PSs, despite their importance, given the potential expansion of IoT systems with the addition of new devices.
- IoT-specific tools are proposed either on-premises or offered as a service in the cloud. Additionally, tools commonly used in traditional software systems, such as Selenium, Robot Framework, and Apache JMeter, are of equal importance for testing IoT systems. We believe that IoT system testers can also leverage other tools like WebdriveIO or Appium for automated testing of web or mobile user interfaces.
- All the challenges summarized in Table 19 are depicted in Figure 5. In Figure 5, each challenge is presented within its box, except for the limitation of existing tools and approaches, which are grouped in one box to reduce the size of Figure 5, resulting in a total of 14 boxes for the challenges. Similarly, in Table 19, the limitations of existing tools, approaches, and lack of testing methodologies are grouped in one entry. In the same table, access to real devices and testing in real-world scenarios are also combined as one entry. Additionally, non-standard compliant devices and different data formats are grouped as one entry. Although Table 19 has 11 entries, these entries encompass all the challenges presented in Figure 5. Individual studies may not have fully addressed specific challenges. However, various PSs discussed approaches that could potentially resolve these challenges. We summarized the proposed approaches found in the PSs that could be used to address some of these challenges in Table 19.
- Many IoT systems have been proposed to evaluate various approaches, but most of these systems are neither available nor accessible online.
- Some authors did not disclose the name of the system they used in their evaluation. For example, in [P29], the authors provided descriptions for various *mobile apps* without specifying their names. We highlighted this in Figure 5 because it is important, even though no specific name was provided.

6 Threats to Validity

There are threats to the construct validity, internal validity, external validity, and conclusion of this study.

- **Construct Validity**. Construct validity relates to sources investigated and data collection. It pertains to the selection of PSs and how we extracted data from these PSs in

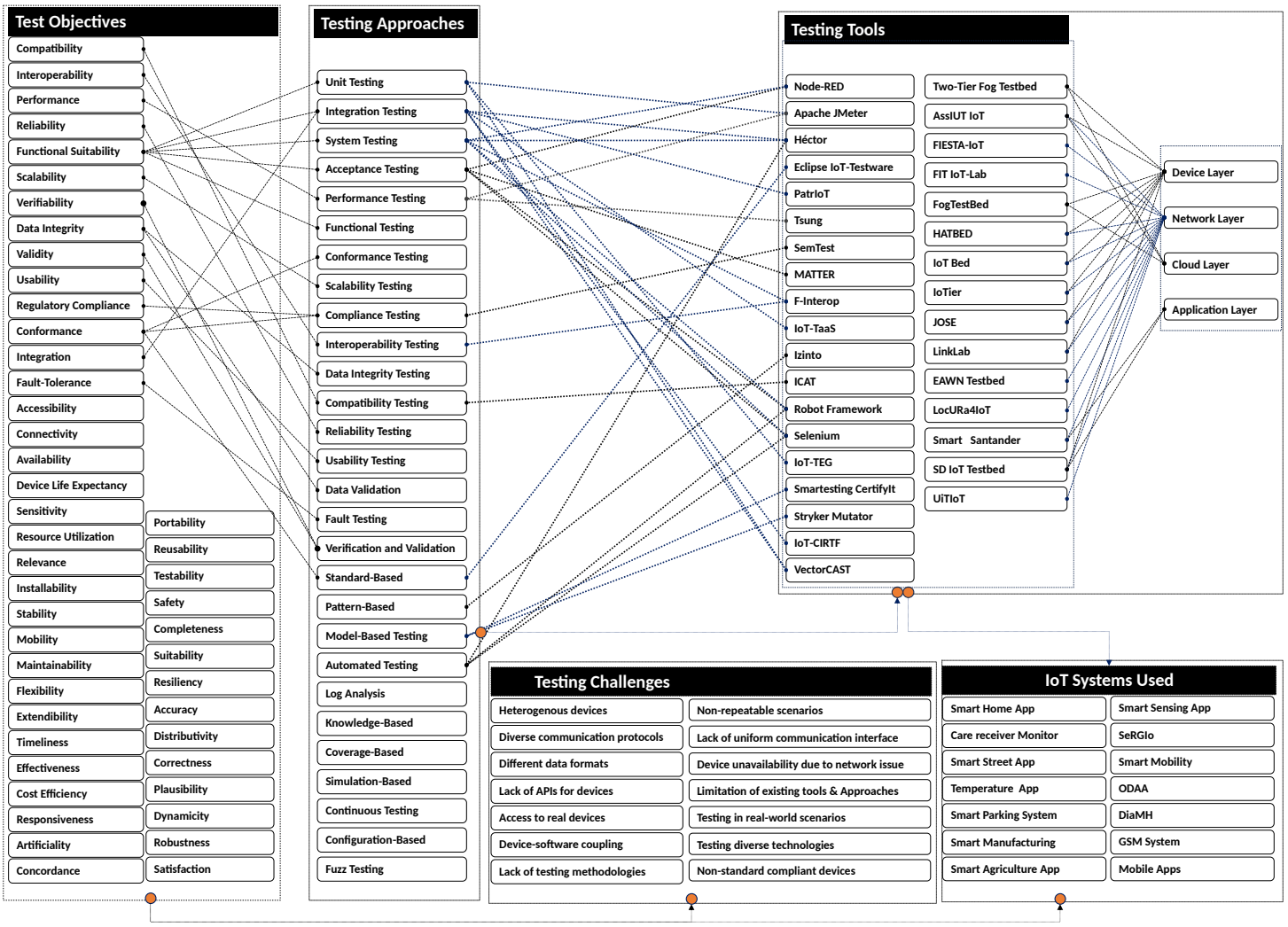


Fig. 5: Results Overview

relation to the RQs.

❶ *Missing relevant studies*. Incompleteness is a threat. Our search relied on titles and keywords, which may have resulted in the omission of relevant studies. The accuracy of our search depends on how well digital libraries organize and categorize papers. To mitigate this threat, we used seven databases, which are reputable and most comprehensive digital libraries for literature reviews [39]. We used two rounds of snowballing to find more potentially relevant studies.

❷ *PSs selection bias*. During the selection of PSs, we may have excluded relevant studies. The subjective nature of manually conducting the selection process could contribute to this biased selection. To mitigate this selection bias, we defined the purpose of the study and the research questions in advance, following the PRISMA guideline, and established clear inclusion and exclusion criteria.

❸ *Data extraction bias*. The data extraction process involved extensive manual efforts, which may be subject to personal bias. To minimize this bias, we defined the data collection form. Two authors followed the agreed form and independently conducted the data extraction. We used interrater agreement and several discussion sessions involving all authors to reach a common agreement. Another threat arises from the inconsistent terminologies in PSs. We discussed each of these inconsistencies and agreed on a unified vocabulary.

✺ **Internal Validity**. Internal validity concerns the methods used in the study and related conclusions.

❶ *Scope of the review*. The research questions do not cover all aspects of IoT systems testing. We may have missed some test objectives that are not based on quality attributes. We missed any test objective, test approach, and tool related to the security testing of IoT systems. We recommended other studies that focused on other aspects of IoT systems testing not covered in this study.

❷ *Completeness of the review*. Our study focuses on IoT systems testing objectives, approaches, tools, and challenges. Our PSs may not provide sufficient details on all these aspects. To minimize this threat, we included only relevant PSs that are specific to each research question.

❸ *Method of the review*. The methodology to conduct research may introduce various threats, including the potential for bias, which can undermine the reliability and validity of the findings. To mitigate this threat,

we followed the updated PRISMA guidelines and carefully selected relevant studies while removing irrelevant ones. All authors defined and reviewed the inclusion/exclusion criteria.

- **External validity**. External validity refers to the generalizability of our findings to all IoT systems testing. This review focuses exclusively on academic research. Industry practices may not be included if they were not reported in academic publications. We reviewed studies published within a specific timeframe, which may limit the extent to which the findings can be generalized. However, we consider this review to be valuable for both academia and industry practitioners, and we will conduct an industrial study to complement it.
- **Conclusion validity** concerns the degree to which the conclusions drawn from the data extracted are reasonable and valid. We ensured the validity of our conclusions by carefully analyzing the data extracted from PSs. To enhance validity, we conducted multiple discussion sessions wherein we collectively drew and cross-verified our conclusions against the extracted data. Therefore, our conclusions solely rely on the findings obtained from PSs.

7 Related Work

We now review the works related to IoT systems testing, categorized into four key areas: Testing Objectives, Testing Approaches, Testing Tools, and Testing Challenges. Table 20 compares our work with related studies.

7.1 Quality Attributes for IoT Systems

Few studies addressed the quality aspects of IoT systems. In [64], the authors conducted a mapping study of quality attributes of IoT systems. Their study aimed to identify quality attributes discussed in the existing literature based on the quality model proposed by ISO/IEC 25000 [46]. They found that researchers discussed performance, suitability, compatibility, usability, security, and maintainability in the context of IoT systems. In [67], the authors provided a detailed classification of IoT quality attributes, which only covered performance, security, and privacy. They suggested the need for a comprehensive study on all quality aspects of IoT systems. In [7], the authors discussed the end-to-end quality of services in IoT systems, focusing on performance metrics of IoT systems. They recommended some quality factors for assessing the quality of IoT systems.

In summary, prior research has addressed various quality attributes in IoT systems, with differing levels of depth and scope. However, a comprehensive review encompassing all quality aspects is still needed.

7.2 Testing Approaches for IoT Systems

Several studies discussed various testing approaches for IoT systems. A recent study [55] summarized the latest testing methods. The study identified cloud-based and machine-learning-based approaches as the two most popular testing methods. The study discussed testing approaches based on the scope of the test and the objective. It also discussed testing methods such as white-box, black-box, and gray-box testing. In [63], the authors provided an overview of methods and approaches available for integration testing of IoT systems. The study concluded that there is a need for more effective testing methods that focus specifically on IoT systems. In [66], the authors analyzed the model-based testing (MBT) approach to ensure the quality of IoT systems. They provided several useful analyses of MBT in the test case generation process. In [56], the authors described several types of IoT testing and discussed some testing challenges. In [70], the authors provided an overview of techniques or approaches that use machine learning (ML) algorithms in the IoT application testing process and automatic generation of test cases from textual language. They identified the use of supervised and unsupervised algorithms for security attack identification, fault prediction or anomaly detection, and test case generation. However, their study only focused on the application layer. In [57], the authors provided an overview of testing approaches for IoT solutions from an industry perspective. They used mined data to propose a framework to enhance the availability of IoT-based healthcare systems. However, this study was limited to the availability of IoT devices in the context of healthcare devices.

7.3 Testing Tools for IoT Systems

Few studies have explored testing tools for IoT systems. One study [55] surveyed the development of testbeds and performance analysis of existing testbeds. This study identified some technical issues, such as concurrency and large-scale, that require further investigation. Authors suggested that intelligent testing and multiple technology integration should be the future direction of IoT testbeds. Other studies [68], [44] provided an overview of the available tools for testing IoT systems, focusing on testbeds, emulators, and simulators. These studies recommended further studies to explore more testing tools and automation procedures for testing and continuous integration for IoT systems. A study [60] provided a comparison of test environment tools for IoT systems. The study selected a test environment that focuses on testing software systems on geo-distributed, heterogeneous computing infrastructures such as IoT and edge/fog architectures. It excluded hardware-only testbeds. It recommended further studies to provide comprehensive test environments, including the simulation and modeling of IoT system characteristics.

7.4 Testing Challenges for IoT Systems

Few studies have attempted to identify the challenges of testing IoT systems. In [56], the authors discussed several limitations and challenges faced by testing teams for IoT systems, identifying five main challenges: lack of standardization, heterogeneity in IoT devices and platforms, device interoperability, security testing, and testing environments and tools. In [65], the authors provided an industry perspective overview of testing challenges for IoT systems, highlighting platform diversity, software-hardware interconnection, real-time data velocity, security, and scalability as the main challenges.

We did not find any study that specifically focused on test objectives, approaches, tools, and challenges. Therefore, our study focuses on these four aspects. We attempted to broaden the scope of our review, and we believe that similar reviews will be required regularly to uncover emerging trends in IoT systems testing.

TABLE 20: Comparison with closely related works

Study	Year	Aim of Study	Research Method	Selected PSs	Focus on			
					TO	TA	TT	TC
[54]	2023	Overview of IoT testing from industry perspective	Survey and Interviews	N.A	◐	◐	◐	●
[40]	2023	Security testing in IoT	Review	$\leq$ 2022	◐	◐	◐	◐
[55]	2022	Testing methods and testbeds in IoT	Survey	N.A	○	◐	◐	○
[56]	2022	Testing types and challenges in IoT	Survey	N.A	○	◐	○	●
[57]	2022	Using ML to test IoT applications	Systematic Mapping	2016-2022	○	◐	○	◐
[58]	2022	Privacy and security, and blockchain in IoT	Review	N.M	○	◐	◐	○
[59]	2021	IoT device's availability testing	Survey	N.A	○	◐	◐	○
[60]	2021	IoT test environments	Survey	N.A	○	○	◐	○
[61]	2021	Fuzzing techniques on IoT devices	Review	N.M	○	◐	◐	○
[62]	2021	Security testing in IoT	Review	2010-2019	○	◐	◐	○
[63]	2020	Methods and approaches for integration testing in IoT	Systematic Mapping	2009-2019	○	●	○	○
[64]	2020	MBT for IoT	Systematic Mapping	2009-2019	◐	◐	◐	○
[65]	2020	IoT testing challenge from industry perspective	Survey	N.A	○	○	○	●
[66]	2019	Software testing techniques in IoT	Systematic Mapping	2008-2018	○	◐	○	○
[43]	2019	Simulators, Emulators, and Testbeds for IoT	Review	N.M	○	○	◐	○
[67]	2018	Methods for quality assurance in IoT	Systematic Mapping	2009-2017	◐	◐	○	○
[68]	2018	Testing types, tools (emulators and simulators), and challenges in IoT	Review	N.M	○	◐	◐	◐
[44]	2018	Testbeds, Emulators, Simulators in IoT	Review	2012-2017	○	○	◐	○
[69]	2018	Testing tools and techniques for IoT	Survey	N.A	○	◐	◐	○
[7]	2017	QoS approaches in IoT architecture	Systematic Mapping	2000-2016	◐	○	○	○
This study		Testing challenges, objectives, approaches, and tools	Review	2012-2022	●	●	●	●

✺ **N.A**:Not Applicable; **N.M**:Not Mentioned.
✺ **TO**:Testing Objective; **TA**: Testing Approach; **TT**: Testing Tools; **TC**:Testing Challenge.
✺ ●:Covered; ◐: Partially Covered; ○: Not Covered.

8 Conclusion And Future Work

This study aimed to understand how IoT systems are tested in terms of objectives, approaches, tools, and challenges. We conducted a detailed review of 83 PSs, compiling 47 quality attributes, with 5 of them specifically related to IoT systems, helping us understand the objectives of testing IoT systems. We provided an overview of testing approaches including 4 testing levels, 15 testing types, 15 testing techniques and practices for IoT systems. We presented the compilation of 19 user testing tools and 15 testbeds for IoT systems testing, highlighting their usage at different stages of IoT systems development. We summarized the challenges encountered in IoT systems testing and highlighted potential research directions to effectively address the emerging and futuristic challenges associated with testing IoT systems.

Our study has the following implications for researchers and practitioners:

- It highlights the objectives (based on quality attributes), approaches, tools, and challenges of IoT systems testing. Furthermore, it identifies avenues for further research due to the limitations of existing approaches and tools, challenges faced by testers, and untested quality attributes.
- It serves as a guide for practitioners to understand different testing approaches and tools.

Future work involves developing IoT systems testing taxonomy using the testing terms mined from our PSs. We believe that once developed, it will help IoT systems practitioners to better understand the why, what, and how of IoT systems testing. We will leverage existing tools and approaches to develop an IoT system testing approach based on this taxonomy. We will also stay updated with emerging approaches and tools, conducting a systematic review with broader coverage and more in-depth analysis.

Acknowledgment

The authors thank the researchers who replied to our request for information on their IoT testing tools. The Canada Research Chair program partly funded this work.

References

[1] T. Alam, "A reliable communication framework and its use in internet of things (IoT)," *CSEIT1835111— Received*, vol. 10, pp. 450–456, 2018.

[2] J. P. Dias, H. S. Ferreira, and T. B. Sousa, "Testing and Deployment Patterns For The Internet-of-Things," in *Proceedings of the 24th European Conference on Pattern Languages of Programs*, 2019, pp. 1–8.

[3] B. Alhafidh and W. Allen, "Design and simulation of a smart home managed by an intelligent self-adaptive system," *International Journal of Engineering Research and Applications*, vol. 6, pp. 2248–2264, 08 2016.

[4] R. Petrolo, V. Loscri, and N. Mitton, "Towards a smart city based on cloud of things, a survey on the smart city vision and paradigms," *Transactions on Emerging Telecommunications Technologies*, vol. 28, 02 2015.

[5] M. Fahmideh and D. Zowghi, "An Exploration of IoT Platform Development," *Information Systems*, vol. 87, p. 101409, 2020.

[6] M. Fahmideh, A. Ahmad, A. Behnaz, J. Grundy, and W. Susilo, "Software Engineering For Internet of Things: The Practitioners' Perspective," *IEEE Transactions on Software Engineering*, vol. 48, no. 8, pp. 2857–2878, 2021.

[7] G. White, V. Nallur, and S. Clarke, "Quality of Service Approaches in IoT: A Systematic Mapping," *Journal of Systems and Software*, vol. 132, pp. 186–203, 2017.

[8] M. Leotta, F. Ricca, D. Clerissi, D. Ancona, G. Delzanno, M. Ribaudo, and L. Franceschini, "Towards an Acceptance Testing Approach for Internet of Things Systems," in *ICWE Workshops*. Springer International Publishing, 2017, pp. 125–138.

[9] M. Bettayeb, O. Abu Waraga, M. Abu Talib, Q. Nasir, and O. Einea, "IoT Testbed Security: Smart Socket and Smart Thermostat," in *2019 IEEE Conference On Application, Information And Network Security(AINS)*, 2019, pp. 18–23.

[10] N. Varghese and R. Sinha, "Can Commercial Testing Automation Tools Work For IoT? A Case Study Of Selenium And Node-RED," in *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2020, pp. 4519–4524.

[11] S. Bosmans, S. Mercelis, J. Denil, and P. Hellinckx, "Testing IoT Systems Using A Hybrid Simulation Based Testing Approach," *Computing*, vol. 101, no. 7, SI, pp. 857–872, JUL 2019.

[12] A. Savidis, Y. Valsamakis, and D. Linaritis, "Simulated IoT Runtime With Virtual Smart Devices: Debugging and Testing End-user Automations," in *Proceedings Of The 17th International Conference On Web Information Systems And Technologies (WEBIST)*, F. Mayo, M. Marchiori, and J. Filipe, Eds., 2021, pp. 145–155.

[13] P. M. Pontes, B. Lima, and J. P. Faria, "Test Patterns For IoT," in *Proceedings of the 9th ACM SIGSOFT international workshop on automating TEST case design, selection, and evaluation*, 2018, pp. 63–66.

[14] H. Kim, A. Ahmad, J. Hwang, H. Baqa, F. Le Gall, M. A. R. Ortega, and J. Song, "IoT-TaaS: Towards a Prospective IoT Testing Framework," *IEEE ACCESS*, vol. 6, pp. 15 480–15 493, 2018.

[15] N. Medhat, S. Moussa, N. Badr, and M. F. Tolba, "Testing Techniques in IoT-Based Systems," in *2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS)*, 2019, pp. 394–401.

[16] M. J. Page, J. E. McKenzie, P. M. Bossuyt, I. Boutron, T. C. Hoffmann, C. D. Mulrow, L. Shamseer, J. M. Tetzlaff, E. A. Akl, S. E. Brennan, *et al.*, "The PRISMA 2020 Statement: An Updated Guideline For Reporting Systematic Reviews," *International journal of surgery*, vol. 88, p. 105906, 2021.

[17] X. Larrucea, A. Combelles, J. Favaro, and K. Taneja, "Software Engineering For The Internet Of Things," *IEEE Software*, vol. 34, no. 1, pp. 24–28, 2017.

[18] P. Sethi and S. R. Sarangi, "Internet of Things: Architectures, Protocols, and Applications," *Journal of Electrical and Computer Engineering*, vol. 2017, 2017.

[19] P. M. Pontes, B. Lima, and J. P. Faria, "Izinto: A Pattern-Based IoT Testing Framework," in *Companion Proceedings for the ISSTA/ECOOP 2018 Workshops*, 2018, pp. 125–131.

[20] H. Kaur and R. Kumar, "A survey on internet of things (iot): Layer-specific, domain-specific and industry-defined architectures," in *Advances in Computational Intelligence and Communication Technology*. Singapore: Springer Singapore, 2021, pp. 265–275.

[21] Otoum, Yazan and Liu, Dandan and Nayak, Amiya, "DL-IDS: a deep learning–based intrusion detection framework for securing IoT," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 3, p. e3803, 2022, e3803 ett.3803. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/ett.3803

[22] Burhan, Muhammad and Rehman, Rana Asif and Khan, Bilal and Kim, Byung-Seo, "IoT Elements, Layered Architectures and Security Issues: A Comprehensive Survey," *Sensors*, vol. 18, no. 9, 2018. [Online]. Available: https://www.mdpi.com/1424-8220/18/9/2796

[23] A. Abdullah, H. Kaur, and R. Biswas, "Universal layers of IoT architecture and its security analysis," in *New Paradigm in Decision Science and Management: Proceedings of ICDSM 2018*. Springer, 2020, pp. 293–302.

[24] T. A. Rao and E. Haq, "Security challenges facing IoT layers and its protective measures," *International Journal of Computer Applications*, vol. 179, no. 27, pp. 31–35, 2018.

[25] H. Touqeer, S. Zaman, R. Amin, M. Hussain, F. Al-Turjman, and M. Bilal, "Smart home security: challenges, issues and solutions at different IoT layers," *The Journal of Supercomputing*, vol. 77, no. 12, pp. 14 053–14 089, 2021.

[26] M. Mayeda and A. Andrews, "Evaluating Software Testing Techniques: A Systematic Mapping Study," *Elsevier*, vol. 123, 2021.

[27] A. Adekanmi, "Research On Software Testing And Effectiveness Of Automation Testing," *DOF10*, vol. 13140, 2019.

[28] B. Hetzel, *The Complete Guide To Software Testing*, 2nd ed. USA: QED Information Sciences, Inc., 1988.

[29] J. D. Hagar, *IoT System Testing*. Springer, 2022.

[30] B. Edvardsson and J. Olsson, "Key concepts for new service development," *Service Industries Journal*, vol. 16, no. 2, pp. 140–164, 1996.

[31] X. Franch and J. Carvallo, "Using quality models in software package selection," *IEEE Software*, vol. 20, no. 1, pp. 34–41, 2003.

[32] I. . 2017, "ISO/IEC/IEEE International Standard - Systems and software engineering–Vocabulary," pp. 1–541, 2017.

[33] ISO, "IEEE/ISO/IEC International Standard for Software and systems engineering–Software testing–Part 3:Test documentation," *ISO/IEC/IEEE 29119-3:2021(E)*, pp. 1–98, 2021.

[34] S. Reid, "Software and Systems Engineering Software Testing Part 1: Concepts and Definitions," ISO/IEC/IEEE 29119-1, Tech. Rep., 2013.

[35] M. J. Page, J. E. McKenzie, P. M. Bossuyt, I. Boutron, T. C. Hoffmann, C. D. Mulrow, L. Shamseer, J. M. Tetzlaff, E. A. Akl, S. E. Brennan, *et al.*, "The PRISMA 2020 Statement: An Updated Guideline For Reporting Systematic Reviews," *International journal of surgery*, vol. 88, p. 105906, 2021.

[36] M. J. Page, D. Moher, P. M. Bossuyt, I. Boutron, T. C. Hoffmann, C. D. Mulrow, L. Shamseer, J. M. Tetzlaff, E. A. Akl, S. E. Brennan, *et al.*, "PRISMA 2020 Explanation And Elaboration: Updated Guidance And Exemplars For Reporting Systematic Reviews," *bmj*, vol. 372, 2021.

[37] B. A. Kitchenham, L. Madeyski, and D. Budgen, "SEGRESS: Software Engineering Guidelines For Reporting Secondary Studies," *IEEE Transactions on Software Engineering*, 2022.

[38] A. Cooke, D. Smith, and A. Booth, "Beyond PICO: the SPIDER Tool For Qualitative Evidence Synthesis," *Qualitative health research*, vol. 22, no. 10, pp. 1435–1443, 2012.

[39] T. Dyba, T. Dingsoyr, and G. K. Hanssen, "Applying Systematic Reviews To Diverse Study Types: An Experience Report," in *First international symposium on empirical software engineering and measurement (ESEM 2007)*. IEEE, 2007, pp. 225–234.

[40] F. Lonetti, A. Bertolino, and F. Di Giandomenico, "Model-Based Security Testing in IoT Systems: A Rapid Review," *Information and Software Technology*, vol. 164, p. 107326, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950584923001817

[41] W. Fei, H. Ohno, and S. Sampalli, "A Systematic Review of IoT Security: Research Potential, Challenges and Future Directions," *ACM Computing Surveys*, 2023.

[42] C. Greco, G. Fortino, B. Crispo, and K.-K. R. Choo, "AI-enabled IoT penetration testing: state-of-the-art and research challenges," *Enterprise Information Systems*, vol. 17, no. 9, 2023.

[43] N. D. Patel, B. M. Mehtre, and R. Wankar, "Simulators, Emulators, and Test-beds for Internet of Things: A Comparison," in *2019 Third International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, 2019, pp. 139–145.

[44] M. Chernyshev, Z. Baig, O. Bello, and S. Zeadally, "Internet of Things (IoT): Research, Simulators, and Testbeds," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1637–1647, 2018.

[45] J. Cohen, "Weighted Kappa: Nominal Scale Agreement Provision For Scaled Disagreement Or Partial Credit," *Psychological bulletin*, vol. 70, no. 4, p. 213, 1968.

[46] Y. I. Irawan and E. S. Negara, "Evaluation of Software Quality Assurance Silampari Smart City Of Lubuklinggau Based On ISO/IEC 25010:2011 Analysis Model," in *2022 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS)*, 2022, pp. 154–160.

[47] A. Taivalsaari and T. Mikkonen, "On the development of iot systems," in *2018 Third International Conference on Fog and Mobile Edge Computing (FMEC)*, 2018, pp. 13–19.

[48] S. Lee, J. Park, H. Choi, and H. Oh, "Energy-efficient ap selection using intelligent access point system to increase the lifespan of iot devices," *Sensors*, vol. 23, no. 11, 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/11/5197

[49] K. Nadiminti, M. D. De Assunçao, and R. Buyya, "Distributed systems and recent innovations: Challenges and benefits," *InfoNet Magazine*, vol. 16, no. 3, pp. 1–5, 2006.

[50] A. Tanenbaum and M. Steen, "Introduction to distributed systems," *Distributed Systems: Principles and Paradigms, Prentice Hall (Jan. 15, 2002)*, pp. 1–33, 2015.

[51] E. Cavalcante, T. Batista, and F. Oquendo, "Supporting dynamic software architectures: From architectural description to implementation," in *2015 12th Working IEEE/IFIP Conference on Software Architecture*. IEEE, 2015, pp. 31–40.

[52] M. G. Patterson, "What is energy efficiency?: Concepts, indicators and methodological issues," *Energy policy*, vol. 24, no. 5, pp. 377–390, 1996.

[53] D. Firesmith, "A Taxonomy of Testing," Carnegie Mellon University, Software Engineering Institute's Insights (blog), Aug 2015, accessed: 2023-May-2. [Online]. Available: http://insights.sei.cmu.edu/blog/a-taxonomy-of-testing/

[54] J. B. Minani, F. Sabir, N. Moha, and Y.-G. Guéhéneuc, "A Multimethod Study of Internet of Things Systems Testing in Industry," *IEEE Internet of Things Journal*, vol. 11, no. 1, pp. 1662–1684, 2024.

[55] S. Zhu, S. Yang, X. Gou, Y. Xu, T. Zhang, and Y. Wan, "Survey Of Testing Methods And Testbed Development Concerning Internet Of Things," *Wireless Personal Communications*, vol. 123, no. 1, pp. 165–194, 2022.

[56] J. A. Fadhil and Q. I. Sarhan, "A Survey on Internet of Things (IoT) Testing," in *2022 International Conference on Computer Science and Software Engineering (CSASE)*. IEEE, 2022, pp. 77–83.

[57] L. Freitas and V. Lelli, "Using Machine Learning on Testing IoT Applications: A Systematic Mapping," in *Proceedings of the Brazilian Symposium on Multimedia and the Web*, 2022, pp. 348–358.

[58] A. Ayub Khan, A. A. Laghari, Z. A. Shaikh, Z. Dacko-Pikiewicz, and S. Kot, "Internet of Things (IoT) Security With Blockchain Technology: A State-of-the-Art Review," *IEEE Access*, vol. 10, pp. 122 679–122 695, 2022.

[59] S. Abbas, M. Naz, Z. Anwaar, M. U. Farooq, and F. U. Khan, "Availability Testing of IoT-Based Health Care Devices: A Survey," in *2021 International Conference on Innovative Computing (ICIC)*, 2021, pp. 1–6.

[60] J. Beilharz, P. Wiesner, A. Boockmeyer, L. Pirl, D. Friedenberger, F. Brokhausen, I. Behnke, A. Polze, and L. Thamsen, "Continuously Testing Distributed IoT Systems: An Overview Of The State Of The Art," in *Service-Oriented Computing–ICSOC 2021 Workshops: AIOps, STRAPS, AI-PA and Satellite Events, Dubai, United Arab Emirates, November 22–25, 2021, Proceedings*. Springer, 2022, pp. 336–350.

[61] M. Eceiza, J. L. Flores, and M. Iturbe, "Fuzzing the Internet of Things: A Review on the Techniques and Challenges for Efficient Vulnerability Discovery in Embedded Systems," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10 390–10 411, 2021.

[62] O. Tauqeer, S. Jan, A. Khadidos, A. Khadidos, F. Khan, and S. Khattak, "Analysis of Security Testing Techniques," *Intelligent Automation and Soft Computing*, vol. 29, pp. 291–306, 05 2021.

[63] M. Bures, M. Klima, V. Rechtberger, X. Bellekens, C. Tachtatzis, R. Atkinson, and B. S. Ahmed, "Interoperability and Integration Testing Methods for IoT Systems: A Systematic Mapping Study," in *International conference on software engineering and formal methods*. Springer, 2020, pp. 93–112.

[64] T. I. Kh. and I. I. Hamarash, "Model-Based Quality Assessment of Internet of Things Software Applications: A Systematic Mapping Study," *Int. J. Interact. Mob. Technol.*, vol. 14, pp. 128–152, 2020.

[65] www.thinxtream.com, "IoT Testing Challenges and Approaches," https://www.thinxtream.com/iot-testing-solutions-services.html, 2020, [White Paper.Accessed 10-Feb-2023].

[66] M. Cortés, R. Saraiva, M. Souza, P. Mello, and P. Soares, "Adoption of Software Testing in Internet of Things: A Systematic Literature Mapping," in *Proceedings of the IV Brazilian Symposium on Systematic and Automated Software Testing*, 2019, pp. 3–11.

[67] B. S. Ahmed, M. Bures, K. Frajtak, and T. Cerny, "Aspects of Quality in Internet of Things (IoT) Solutions: A Systematic Mapping Study," *IEEE Access*, vol. 7, pp. 13 758–13 780, 2019.

[68] J. P. Dias, F. Couto, A. C. Paiva, and H. S. Ferreira, "A Brief Overview of Existing Tools for Testing the Internet-of-Things," in *2018 IEEE international conference on software testing, verification and validation workshops (ICSTW)*. IEEE, 2018, pp. 104–109.

[69] G. Murad, A. Badarneh, A. Qusef, and F. Almasalha, "Software Testing Techniques in IoT," in *2018 8th International Conference on Computer Science and Information Technology (CSIT)*. IEEE, 2018, pp. 17–21.

[70] S. Abbas, M. Naz, Z. Anwaar, M. U. Farooq, and F. U. Khan, "Availability Testing of IoT-Based Health Care Devices: A Survey," in *2021 International Conference on Innovative Computing (ICIC)*. IEEE, 2021, pp. 1–6.

SELECTED PRIMARY STUDIES (PSS)

[P1] M. Bures, X. Bellekens, K. Frajtak, and B. S. Ahmed, "A Comprehensive View On Quality Characteristics Of The IoT Solutions," in *3rd EAI International Conference on IoT in Urban Space*. Springer, 2020, pp. 59–69.

[P2] W. Zhang, J. Wang, G. Han, S. Huang, Y. Feng, and L. Shu, "A Data Set Accuracy Weighted Random Forest Algorithm For IoT Fault Detection Based On Edge Computing And Blockchain," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2354–2363, 2020.

[P3] N. Medhat, S. M. Moussa, N. L. Badr, and M. F. Tolba, "A Framework For Continuous Regression And Integration Testing In IoT Systems Based On Deep Learning And Search-Based Techniques," *IEEE Access*, vol. 8, pp. 215 716–215 726, 2020.

[P4] S. Nguyen, Z. Salcic, X. Zhang, and A. Bisht, "A Low-Cost Two-Tier Fog Computing Testbed For Streaming IoT-Based Applications," *IEEE Internet of Things Journal*, vol. 8, no. 8, pp. 6928–6939, 2020.

[P5] K. İnçki and I. Ari, "A Novel Runtime Verification Solution For IoT Systems," *IEEE Access*, vol. 6, pp. 13 501–13 512, 2018.

[P6] B. Ramprasad, J. Mukherjee, and M. Litoiu, "A Smart Testing Framework For IoT Applications," in *2018 IEEE/ACM International Conference on Utility and Cloud Computing Companion (UCC Companion)*. IEEE, 2018, pp. 252–257.

[P7] M. Padmanabhan, "A Study On Transaction Specification Based Software Testing For Internet Of Things," in *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)*. IEEE, 2018, pp. 1–6.

[P8] M. Leotta, D. Clerissi, D. Olianas, F. Ricca, D. Ancona, G. Delzanno, L. Franceschini, and M. Ribaudo, "An Acceptance Testing Approach For Internet Of Things Systems," *IET Software*, vol. 12, no. 5, pp. 430–436, 2018.

[P9] S. K. Datta, C. Bonnet, H. Baqa, M. Zhao, and F. Le-Gall, "Approach For Semantic Interoperability Testing In Internet Of Things," in *2018 Global Internet of Things summit (GIoTS)*. IEEE, 2018, pp. 1–6.

[P10] M. AbdelHafeez and M. AbdelRaheem, "Assiut IoT: A Remotely Accessible Testbed For Internet Of Things," in *2018 IEEE Global Conference on Internet of Things (GCIoT)*. IEEE, 2018, pp. 1–6.

[P11] J. Hwang, A. Aziz, N. Sung, A. Ahmad, F. Le Gall, and J. Song, "AUTOCON-IoT: Automated And Scalable Online Conformance Testing For IoT Applications," *IEEE Access*, vol. 8, pp. 43 111–43 121, 2020.

[P12] L. G. Guşeilă, D.-V. Bratu, and S.-A. Moraru, "Continuous Testing In The Development Of IoT Applications," in *2019 International Conference on Sensing and Instrumentation in IoT Era (ISSI)*. IEEE, 2019, pp. 1–6.

[P13] M. Serrano, A. Gyrard, M. Boniface, P. Grace, N. Georgantas, R. Agarwal, P. Barnagui, F. Carrez, B. Almeida, T. Teixeira, *et al.*, "Cross-Domain Interoperability Using Federated Interoperable Semantic IoT/Cloud Testbeds And Applications: The FIESTA-IoT Approach," in *Building the Future Internet through FIRE*. River Publishers, 2022, pp. 287–321.

[P14] L. Hu, W. E. Wong, D. R. Kuhn, R. N. Kacker, and S. Li, "CT-IoT: A Combinatorial Testing-Based Path Selection Framework For Effective Iot Testing," *Empirical Software Engineering*, vol. 27, pp. 1–38, 2022.

[P15] S. K. Datta, C. Bonnet, H. Baqa, M. Zhao, and F. Le-Gall, "Developing And Integrating A Semantic Interoperability Testing Tool In F-Interop Platform," in *2018 IEEE Region Ten Symposium (Tensymp)*. IEEE, 2018, pp. 112–117.

[P16] K. Fizza, P. Jayaraman, A. Banerjee, D. Georgakopoulos, and R. Ranjan, "Evaluating Sensor Data Quality in Internet of Things Smart Agriculture Applications," *IEEE Micro*, vol. 42, no. 1, pp. 51–60, 2022.

[P17] L. Gutiérrez-Madroñal, A. García-Domínguez, and I. Medina-Bulo, "Evolutionary Mutation Testing For IoT With Recorded And Generated Events," *Software: Practice and Experience*, vol. 49, no. 4, pp. 640–672, 2019.

[P18] S. Ziegler, S. Fdida, C. Viho, and T. Watteyne, "F-Interop: Online Platform Of Interoperability And Performance Tests For The Internet Of Things," *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*, vol. 190, pp. 49–55, 2017.

[P19] M. Palattella, F. Sismondi, T. Chang, L. Baron, M. Vučinić, P. Modernell, X. Vilajosana, and T. Watteyne, "F-Interop Platform And Tools: Validating IoT Implementations Faster," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11104 LNCS, pp. 332–343, 2018.

[P20] C. Adjih, E. Baccelli, E. Fleury, G. Harter, N. Mitton, T. Noel, R. Pissard-Gibollet, F. Saint-Marcel, G. Schreiner, J. Vandaele, *et al.*, "FIT IoT-LAB: A Large Scale Open Experimental IoT Testbed," in *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*. IEEE, 2015, pp. 459–464.

[P21] Y. Karim and R. Hasan, "FogTestBed: A Generic Architecture For Testbed For Fog-Based Systems," in *2020 SoutheastCon*. IEEE, 2020, pp. 1–7.

[P22] V. Geetha Lekshmy and J. Kannimoola, "Formal Verification Of IoT Protocol: In Design-Time And Run-Time Perspective," *Lecture Notes in Networks and Systems*, vol. 145, pp. 873–884, 2021.

[P23] D. Kümper, E. Reetz, M. Fischer, E. Pulvermueller, and R. Tönjes, "From Semantic IoT-Service Descriptions To Executable Test Cases-Information Flow Of An Implemented Test Framework," in *VALID 2014-6th International Conference on Advances in System Testing and Validation Lifecycle*. International Academy, Research and Industry Association, IARIA, 2014.

[P24] V. Sharma, J. Kim, S. Kwon, I. You, and H.-C. Chen, "Fuzzy-Based Protocol For Secure Remote Diagnosis Of IoT Devices In 5G Networks," *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*, vol. 246, pp. 54–63, 2018.

[P25] L. Yi, J. Ma, and T. Zhang, "HATBED: A Distributed Hardware Assisted Testbed For Non-Invasive Profiling Of IoT Devices," in *Proceedings of the 2nd Workshop on Benchmarking Cyber-Physical Systems and Internet of Things*, 2019, pp. 13–17.

[P26] I. Behnke, L. Thamsen, and O. Kao, "Héctor: A Framework For Testing IoT Applications Across Heterogeneous Edge And Cloud Testbeds," in *Proceedings of the 12th IEEE/ACM international conference on utility and cloud computing companion*, 2019, pp. 15–20.

[P27] X. Liu, B. Cui, J. Fu, and J. Ma, "HFuzz: Towards Automatic Fuzzing Testing Of NB-IoT Core Network Protocols Implementations," *Future Generation Computer Systems*, vol. 108, pp. 390–400, 2020.

[P28] E. S. Reetz, D. Kuemper, K. Moessner, and R. Tönjes, "How To Test IoT-Based Services Before Deploying Them Into Real World," in *European Wireless 2013; 19th European Wireless Conference*. VDE, 2013, pp. 1–6.

[P29] W.-K. Chen, C.-H. Liu, W. W.-Y. Liang, and M.-Y. Tsai, "ICAT: An IoT Device Compatibility Testing Tool," in *2018 25th Asia-Pacific Software Engineering Conference (APSEC)*. IEEE, 2018, pp. 668–672.

[P30] M. Krichen, "Improving Formal Verification And Testing Techniques For Internet Of Things And Smart Cities," *Mobile networks and applications*, pp. 1–12, 2019.

[P31] S. T. Demirel, M. Demirel, I. Dogru, and R. Das, "InterOpT: A new testing platform based on oneM2M standards for IoT Systems," in *2019 International Symposium on Networks, Computers and Communications (ISNCC)*. IEEE, 2019, pp. 1–6.

[P32] S. Popereshnyak, O. Suprun, O. Suprun, and T. Wieckowski, "IoT Application Testing Features Based On The Modelling Network," in *2018 XIV-th International Conference on Perspective Technologies and Methods in MEMS Design (MEMSTECH)*. IEEE, 2018, pp. 127–131.

[P33] A. Makhshari and A. Mesbah, "IoT Bugs And Development Challenges," in *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*. IEEE, 2021, pp. 460–472.

[P34] E. J. Marinissen, Y. Zorian, M. Konijnenburg, C.-T. Huang, P.-H. Hsieh, P. Cockburn, J. Delvaux, V. Rožić, B. Yang, D. Singelée, *et al.*, "IoT: Source Of Test Challenges," in *2016 21th IEEE European test symposium (ETS)*. IEEE, 2016, pp. 1–10.

[P35] M. M. Hossain, S. Al Noor, Y. Karim, and R. Hasan, "IoTBed: A Generic Architecture for Testbed as a Service for Internet of Things-Based Systems," in *ICIoT*, 2017, pp. 42–49.

[P36] F. Nikolaidis, M. Marazakis, and A. Bilas, "IoTier: A Virtual Testbed To Evaluate Systems For IoT Environments," in *2021 IEEE/ACM 21st International Symposium on Cluster, Cloud and Internet Computing (CCGrid)*. IEEE, 2021, pp. 676–683.

[P37] M. Norris, B. Celik, P. Venkatesh, S. Zhao, P. McDaniel, A. Sivasubramaniam, and G. Tan, "IoTRepair: Systematically Addressing Device Faults In Commodity IoT," in *2020 IEEE/ACM Fifth International Conference on Internet-of-Things Design and Implementation (IoTDI)*. IEEE, 2020, pp. 142–148.

[P38] H. Kim, A. Ahmad, J. Hwang, H. Baqa, F. Le Gall, M. A. R. Ortega, and J. Song, "IoT-TaaS: Towards A Prospective IoT Testing Framework," *IEEE Access*, vol. 6, pp. 15 480–15 493, 2018.

[P39] A. Velez-Estevez, L. Gutiérrez-Madroñal, and I. Medina-Bulo, "IoT-TEG 4.0: A New Approach 4.0 for Test Event Generation," *IEEE Transactions on Reliability*, vol. 71, no. 3, pp. 1368–1380, 2021.

[P40] L. Gutiérrez-Madroñal, I. Medina-Bulo, and J. J. Domínguez-Jiménez, "IoT-TEG: Test Event Generator System," *Journal of Systems and Software*, vol. 137, pp. 784–803, 2018.

[P41] I. Schieferdecker, S. Kretzschmann, A. Rennoch, and M. Wagner, "IoT-Testware: An Eclipse Project," in *2017 IEEE international conference on software quality, reliability and security (QRS)*. IEEE, 2017, pp. 1–8.

[P42] P. M. Pontes, B. Lima, and J. P. Faria, "Izinto: A Pattern-Based IoT Testing Framework," in *Companion Proceedings for the ISSTA/ECOOP 2018 Workshops*, 2018, pp. 125–131.

[P43] Y. Teranishi, Y. Saito, S. Murono, and N. Nishinaga, "JOSE: An Open Testbed For Field Trials Of Large-Scale IoT Services," *Journal of the National Institute of Information and Communications Technology*, vol. 62, no. 2, pp. 151–159, 2016.

[P44] Y. Gao, J. Zhang, G. Guan, and W. Dong, "LinkLab: A Scalable And Heterogeneous Testbed For Remotely Developing And Experimenting IoT Applications," in *2020 IEEE/ACM Fifth International Conference on Internet-of-Things Design and Implementation (IoTDI)*. IEEE, 2020, pp. 176–188.

[P45] A. van Den Bossche, R. Dalcé, and T. Val, "Locura4IoT: A Testbed Dedicated To Accurate Localization Of Wireless Nodes In The IoT," *IEEE Sensors Journal*, vol. 22, no. 6, pp. 5437–5446, 2021.

[P46] P. A. Regis, A. N. Patra, and S. Sengupta, "Low-Cost Wireless Testbed For Internet Of Things Ad Hoc Networks Prototyping And Evaluation," in *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*. IEEE, 2020, pp. 1–6.

[P47] D. Olianas, M. Leotta, and F. Ricca, "MATTER: A Tool For Generating End-To-End IoT Test Scripts," *Software Quality Journal*, vol. 30, no. 2, pp. 389–423, 2022.

[P48] A. Ahmad, F. Bouquet, E. Fourneret, F. Le Gall, and B. Legeard, "Model-Based Testing As A Service For IoT Platforms," in *Leveraging Applications of Formal Methods, Verification and Validation: Discussion, Dissemination, Applications: 7th International Symposium, ISoLA 2016, Imperial, Corfu, Greece, October 10-14, 2016, Proceedings, Part II 7*. Springer, 2016, pp. 727–742.

[P49] M. Tappler, B. K. Aichernig, and R. Bloem, "Model-Based Testing IoT Communication Via Active Automata Learning," in *2017 IEEE International conference on software testing, verification and validation (ICST)*. IEEE, 2017, pp. 276–287.

[P50] M. Bures, B. S. Ahmed, V. Rechtberger, M. Klima, M. Trnka, M. Jaros, X. Bellekens, D. Almog, and P. Herout, "PatrIoT: IoT Automated Interoperability And Integration Testing Framework," in *2021 14th IEEE Conference on Software Testing,Verification and Validation (ICST)*. IEEE, 2021, pp. 454–459.

[P51] J. Esquiagola, L. C. de Paula Costa, P. Calcina, G. Fedrecheski, and M. Zuffo, "Performance Testing Of An Internet Of Things Platform," in *IoTBDS*, 2017, pp. 309–314.

[P52] M. Singh, G. Baranwal, and A. K. Tripathi, "QoS-Aware Selection of IoT-Based Service," *Arabian Journal for Science and Engineering*, vol. 45, no. 12, pp. 10 033–10 050, 2020.

[P53] L. Chhiba, A. Marzak, and M. Sidqui, "Quality Attributes for Evaluating IoT Healthcare Systems," in *Innovations in Smart Cities Applications Volume 5: The Proceedings of the 6th International Conference on Smart City Applications*. Springer, 2022, pp. 495–505.

[P54] M. Singh and G. Baranwal, "Quality of Service (QoS) in Internet of Things," in *2018 3rd International Conference On Internet of Things: Smart Innovation and Usages (IoT-SIU)*. IEEE, 2018, pp. 1–6.

[P55] C. Păduraru, R. Cristea, and E. Stăniloiu, "RiverIoT: A Framework Proposal For Fuzzing IoT Applications," in *2021 IEEE/ACM 3rd International Workshop on Software Engineering Research and Practices for the IoT (SERP4IoT)*. IEEE, 2021, pp. 52–58.

[P56] K. Inçki, İ. Arı, and H. Sözer, "Runtime Verification Of IoT Systems Using Complex Event Processing," in *2017 IEEE 14th International Conference on Networking, Sensing and Control (ICNSC)*. IEEE, 2017, pp. 625–630.

[P57] M. Abououf, R. Mizouni, S. Singh, H. Otrok, and E. Damiani, "Self-Supervised Online and Lightweight Anomaly and Event Detection for IoT Devices," *IEEE Internet of Things Journal*, vol. 9, no. 24, pp. 25 285–25 299, 2022.

[P58] F.-K. Tsai, C.-C. Chen, T.-F. Chen, and T.-J. Lin, "Sensor Abnormal Detection And Recovery Using Machine Learning For IoT Sensing Systems," in *2019 IEEE 6th International Conference on Industrial Engineering and Applications (ICIEA)*. IEEE, 2019, pp. 501–505.

[P59] H. Sándor, B. Genge, and Z. Szántó, "Sensor Data Validation And Abnormal Behavior Detection In The Internet of Things," in *2017*

16th RoEduNet Conference: Networking in Education and Research (RoEduNet). IEEE, 2017, pp. 1–5.

[P60] N. A. M. Alduais, J. Abdullah, A. Jamil, L. Audah, and R. Alias, "Sensor Node Data Validation Techniques For Real-time IoT/WSN Application," in *2017 14th International Multi-Conference on Systems, Signals & Devices (SSD)*. IEEE, 2017, pp. 760–765.

[P61] L. Sanchez, L. Muñoz, J. A. Galache, P. Sotres, J. R. Santana, V. Gutierrez, R. Ramdhany, A. Gluhak, S. Krco, E. Theodoridis, *et al.*, "SmartSantander: IoT Experimentation Over a Smart City Testbed," *Computer Networks*, vol. 61, pp. 217–238, 2014.

[P62] J. Kiruthika and S. Khaddaj, "Software Quality Issues And Challenges Of Internet Of Things," in *2015 14th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)*. IEEE, 2015, pp. 176–179.

[P63] Q. Mateen, M. Sirshar, *et al.*, "Software Quality Assurance in Internet of Things," *Int. J. Comput. Appl*, vol. 109, no. 9, pp. 16–24, 2015.

[P64] T.-B. Tan and W.-K. Cheng, "Software Testing Levels in Internet of Things (IoT) Architecture," in *New Trends in Computer Technologies and Applications: 23rd International Computer Symposium, ICS 2018, Yunlin, Taiwan, December 20–22, 2018, Revised Selected Papers 23*. Springer, 2019, pp. 385–390.

[P65] G. Murad, A. Badarneh, A. Qusef, and F. Almasalha, "Software Testing Techniques in IoT," in *2018 8th International conference on computer science and information technology (CSIT)*. IEEE, 2018, pp. 17–21.

[P66] J. Sendorek, T. Szydlo, and R. Brzoza-Woch, "Software-Defined Virtual Testbed For IoT Systems," *Wireless Communications and Mobile Computing*, vol. 2018, pp. 1–11, 2018.

[P67] A. Kaiser and S. Hackel, "Standards-Based IoT Testing With Open-Source Test Equipment," in *2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C)*. IEEE, 2019, pp. 435–441.

[P68] G. Guerrero-Ulloa, M. J. Hornos, and C. Rodríguez-Domínguez, "TDDM4IoTS:A Test-Driven Development Methodology For Internet Of Things (IoT)-Based Systems," in *Applied Technologies: First International Conference, ICAT 2019, Quito, Ecuador, December 3–5, 2019, Proceedings, Part I*. Springer, 2020, pp. 41–55.

[P69] R. Leone, F. Sismondi, T. Watteyne, and C. Viho, "Technical Overview of F-Interop," in *Interoperability, Safety and Security in IoT: Second International Conference, InterIoT 2016 and Third International Conference, SaSeIoT 2016, Paris, France, October 26-27, 2016, Revised Selected Papers 2*. Springer, 2017, pp. 11–17.

[P70] J. P. Dias, H. S. Ferreira, and T. B. Sousa, "Testing and Deployment Patterns for the Internet-of-Things," in *Proceedings of the 24th European Conference on Pattern Languages of Programs*, 2019, pp. 1–8.

[P71] A. Malini, A. Yugakiruthika, S. P. Gunasekar, and R. Preethi, "Testing As A Service Focused on Semantic Interoperability: An Approach," in *2021 4th Biennial International Conference on Nascent Technologies in Engineering (ICNTE)*. IEEE, 2021, pp. 1–5.

[P72] M. A. Walker, D. C. Schmidt, and A. Dubey, "Testing At Scale Of IoT Blockchain Applications," in *Advances in Computers*. Elsevier, 2019, vol. 115, pp. 155–179.

[P73] S. Bosmans, S. Mercelis, J. Denil, and P. Hellinckx, "Testing IoT Systems Using A Hybrid Simulation-Based Testing Approach," *Computing*, vol. 101, pp. 857–872, 2019.

[P74] C. Buratti, A. Stajkic, G. Gardasevic, S. Milardo, M. D. Abrignani, S. Mijovic, G. Morabito, and R. Verdone, "Testing Protocols for the Internet of Things on the EuWIn Platform," *IEEE Internet of Things Journal*, vol. 3, no. 1, pp. 124–133, 2015.

[P75] N. Medhat, S. Moussa, N. Badr, and M. F. Tolba, "Testing Techniques in IoT Based Systems," in *2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS)*, 2019, pp. 394–401.

[P76] M. Krichen and M. Lahami, "Towards A Runtime Testing Framework For Dynamically Adaptable Internet Of Things Networks In Smart Cities," *Smart Infrastructure and Applications: Foundations for Smarter Cities and Societies*, pp. 589–607, 2020.

[P77] M. Leotta, D. Ancona, L. Franceschini, D. Olianas, M. Ribaudo, and F. Ricca, "Towards A Runtime Verification Approach For Internet Of Things Systems," in *Current Trends in Web Engineering: ICWE 2018 International Workshops, MATWEP, EnWot, KD-WEB, WEOD, TourismKG, Cáceres, Spain, June 5, 2018, Revised Selected Papers 18*. Springer International Publishing, 2018, pp. 83–96.

[P78] D. Amalfitano, N. Amatucci, V. De Simone, V. Riccio, and F. A. Rita, "Towards a Thing-In-the-Loop Approach For The Verification And Validation Of IoT Systems," in *Proceedings of the 1st ACM Workshop on the Internet of Safe Things*, 2017, pp. 57–63.

[P79] M. Leotta, F. Ricca, D. Clerissi, D. Ancona, G. Delzanno, M. Ribaudo, and L. Franceschini, "Towards An Acceptance Testing Approach For Internet Of Things Systems," in *Current Trends in Web Engineering: ICWE 2017 International Workshops, Liquid Multi-Device Software and EnWoT, practi-O-web, NLPIT, SoWeMine, Rome, Italy, June 5-8, 2017, Revised Selected Papers 17*. Springer, 2018, pp. 125–138.

[P80] E. E. Kim and S. Ziegler, "Towards An Open Framework Of Online Interoperability And Performance Tests For The Internet Of Things," in *2017 Global Internet of Things Summit (GIoTS)*. IEEE, 2017, pp. 1–6.

[P81] P. Singh, F. Flammini, M. Caporuscio, M. Saman Azari, and J. Thornadtsson, "Towards Self-Healing in the Internet of Things by Log Analytics and Process Mining," in *30th European Safety and Reliability Conference and the 15th Probabilistic Safety Assessment and Management Conference,(ESREL2020 PSAM15), 01–05 November 2020, Venice, Italy*, 2020, pp. 4644–4651.

[P82] N. Ly-Trong, C. Dang-Le-Bao, D. Huynh-Van, and Q. Le-Trung, "UiTIOT v3: A Hybrid Testbed For Evaluation Of Large-Scale IoT Networks," in *Proceedings of the 9th International Symposium on Information and Communication Technology*, 2018, pp. 155–162.

[P83] D. Kuemper, T. Iggena, R. Toenjes, and E. Pulvermueller, "Valid. IoT: A Framework For Sensor Data Quality Analysis And Interpolation," in *Proceedings of the 9th ACM Multimedia Systems Conference*, 2018, pp. 294–303.