Choosing the Right Test Automation Tool: a Grey Literature Review of Practitioner Sources

Päivi Raulamo-Jurvanen
M3S (M-Group)
University of Oulu
Finland
paivi.raulamo-jurvanen@oulu.fi

Mika Mäntylä
M3S (M-Group)
University of Oulu
Finland
mika.mantyla@oulu.fi

Vahid Garousi
SnT Center, University of Luxembourg
Department of Computer Engineering
Hacettepe University, Ankara, Turkey
garousi@svv.lu

ABSTRACT

Background: Choosing the right software test automation tool is not trivial, and recent industrial surveys indicate lack of right tools as the main obstacle to test automation. **Aim:** In this paper, we study how practitioners tackle the problem of choosing the right test automation tool. **Method:** We synthesize the "voice" of the practitioners with a grey literature review originating from 53 different companies. The industry experts behind the sources had roles such as "Software Test Automation Architect", and "Principal Software Engineer". **Results:** Common consensus about the important criteria exists but those are not applied systematically. We summarize the scattered steps from individual sources by presenting a comprehensive process for tool evaluation with 12 steps and a total of 14 different criteria for choosing the right tool. **Conclusions:** The practitioners tend to have general interest in and be influenced by related grey literature as about 78% of our sources had at least 20 backlinks (a reference comparable to a citation) while the variation was between 3 and 759 backlinks. There is a plethora of different software testing tools available, yet the practitioners seem to prefer and adopt the widely known and used tools. The study helps to identify the potential pitfalls of existing processes and opportunities for comprehensive tool evaluation.

CCS CONCEPTS

Software and its engineering→Software maintenance tools • Software and its engineering→Formal software verification • Software and its engineering→Empirical software validation.

KEYWORDS

Grey literature review; software test automation; test automation tool; tool selection.[1]

1 INTRODUCTION

The growing size and complexity of modern software systems increases the need for test automation [4]. One of the important factors resulting in delays in software projects has been reported to be tool-related issues [33]. Findings from a past study [29] reported high initial investments in automation setup, and tool selection and training as notable limitations and challenges of test automation. According to the recent industrial surveys [7, 19, 39], software test automation is considered an area of increased interest amongst practitioners. Software test automation is tool-oriented domain, claimed to be the main area of improvement opportunities in testing activities, and requiring investments in time, cost and effort [7], even when utilizing open source or proprietary tools. It is suggested that research should address how testing can be improved by domain knowledge, by discovering specialized approaches, processes and tools [4].

We find reflection on past experience and knowledge beneficial for the problem of choosing the right test tool. Dybå et al. [12] discuss "*reflective practice*" in Software Engineering (SE), how to be successful, what other people may think and how to cope if the assumptions are wrong. An exploratory study [1] concluded that software engineers spend a considerable portion of their time daily in some form of information gathering. According to the results, the internet is often used as the primary source of information. Aspects such as costs, ease of access and trustworthiness of information have been considered important in information seeking practices, over time [1, 13, 18]. The SE community should attempt to collect evidence, not only from formal studies such as experiments and industrial case studies [11], but also from experience reports about industrial projects and contexts, shared online by practitioners (referred to as the grey literature). In SE, the state-of-the-practice can be captured and documented by practitioners themselves, by observing courses of action and compiling experience, knowledge and expert opinions, in the forms of technical reports, blogs, web-pages and white papers.

As per our experience in conducting industry-academia collaborations in software testing, we have observed that choosing the right software test automation tool is not trivial for many practitioners. To address that need in this paper, we study how practitioners tackle the practical problem of choosing the right test automation tool in practice, in the software industry, by synthesizing the "voice" of the practitioners with a grey literature review of 60 sources. To address that need, we raise five research questions (RQs) in Section 3, and study the different tool selection criteria (RQ1), research methods used (RQ2), tool selection

processes (RQ3), quality of and interest in those sources (RQ4), and test tools and SUTs (RQ5).

The remainder of this paper is structured as follows. A review of the related work is presented in Section 2. We describe the goal of the study and the research methodology in Section 3 and Section 4, respectively. The results are presented in Section 5. Section 6 summarizes the findings and provides discussion on the lessons learned. Finally, in Section 7, we draw the conclusions and suggest areas for further research.

2 BACKGROUND AND RELATED WORK

Tool selection has mostly been claimed to be based on visible attributes or intuitive understanding of expected impacts rather than on established, formal criteria or evaluation of the tools, or analysis of impacts on a specific project [6, 9]. Furthermore, it has been claimed that for increasing productivity, project size and development processes are important factors for selecting a tool [6]. There are scientific case studies and experimental comparisons of technology investigations on the level of a particular need, such as test data generation, for example. However, they do not address general tool selection criteria for candidate tools, do not establish a baseline relative to decision-making for choosing a tool and do not address the voice of the practitioners at large.

Finding the right tool for a given context and to a given purpose is a difficult practical problem, as we experienced in [30]. There is a vast number of software testing and test automation tools available, both commercial and open source. The process of choosing the right tool requires, at least in theory, finding of a set of suitable candidate tools, comparison of those candidate tools and finally, selection of the most appropriate, efficient and effective one for the testing needs and tasks in the context in question.

Despite the extensive supply of software test automation tools available, the findings from research and recent industrial surveys indicate lack of right tools as the main obstacle to test automation [8, 19, 20]. Similar findings have been observed in [29] as tool selection was found to be a limitation to test automation and available tools in the markets (at the time) were not always suitable for the needs of practitioners. It is claimed that when focused on a goal and having a limited number of candidate tools, people may apply those tools in an inappropriate way, or a confirmation bias may lead to impractical solutions [38]. "*I suppose it is tempting, if the only tool you have is a hammer, to treat everything as if it were a nail*" [23]. Unsurprisingly, consultation services related to test tools and automation in general have been ranked among the most required services from external consultants in software testing [19].

An experimental study with a web application concluded HP QTP to be the best tool when compared with TestComplete and Selenium on the basis of SUT, budget and required efficiency [21]. A comparison of two visual GUI testing tools (Sikuli, an open source tool and a commercial tool remaining nameless) in an industrial context was presented in [5]. The study concluded visual GUI testing as applicable technology for automated system testing. However, there were no statistically significant differences between the two tools, as those were reported to work equally well in either test development or test execution [5]. Findings in [34] revealed the unit testing tools (JCrasher, TestGen4j and JUB) generated tests very poorly for the task of detecting defects. Lack of effective, quality test data generation tools (especially free tools at the time) was considered a problem [34]. In a literature review of acceptance test driven development (ATDD) [35] FitNesse was concluded to be a tool easy to learn and use, and identified as the most prominent tool in the research papers while specification and maintenance of test cases were claimed to be time consuming, in general. Although tools like JAccept, Cucumber, Robot, RSpec Selenium, EasyAccept and Sikuli were referred to by name in that paper, only Fit and FitNesse were included in the search terms of the study [35].

A past Systematic Literature Review (SLR) [29] reported differences in the sources of evidence for benefits and limitations of test automation. Interestingly, tool selection and training was one of the limitations discussed. It was concluded that benefits often originated from stronger sources of evidence (e.g. experiments and case studies) while limitations were mainly based on experience reports [29]. The expected reason for benefits being reported by stronger sources of evidence [29] was the publication bias of positive results [22]. Publication bias has been claimed to be a problem, particularly for formal experiments. According to [22], the issue can be addressed with other sources of evidence like scanning grey literature or even asking experts and researchers for unpublished results. According to Poston and Sexton [28], in early years of test automation (in 1990's), only a few practitioners wrote follow-up reports on savings or losses related to test automation tools. Nowadays, the amount of grey literature related to selection and usage of such tools is vast. The internet enables people to collaborate and share information, experiences and knowledge about related benefits, challenges and limitations.

Our recent study [16] about when and what to automate in software testing found that test-tool-related criteria are part of the decision-making on whether to automate software testing at all. The test-tool-related criteria included decision of the tool to use, meeting tool costs, availability of a suitable tool fitting the purpose and positive results from experimenting with the potential tool [16]. In this paper, we build upon the findings of [16] and focus solely on the tool selection.

The body of related work, as discussed above, supports our views of the fact that choosing the right tool(s) for software test automation is an important context-specific, practitioner-oriented problem and there is a need for Grey Literature Review (GLR) to synthesize the "voice" of practitioners as they have shared in their writings online. It is claimed that "*inclusion of grey literature might create an opportunity to take into account the important contextual information, without losing the level of rigor required for a systematic review*" [3]. A recent study on Multivocal Literature Reviews (MLRs) emphasized the importance of grey literature for topics where the "voice of practice" is broad (and more active than academic literature) [14]. A checklist for

assisting the decision-making whether or not to include grey literature from another domain [3] is shown in **Table** 1, where the rationale for our case is indicated in bold. One or more "Yes" responses in that checklist suggest inclusion of grey literature [3]. Thus, we find grey literature of relevance in the matter of tool selection process.

3 GOAL AND RESEARCH QUESTIONS

The goal of the study was to analyze how software practitioners address the practical problem of choosing the right test automation tool by conducting a GLR. The focus was on finding out the important criteria for choosing test automation tools as well as type of methodology used and type of contribution provided by practitioners. The most comprehensive investigation in this subject would have been to conduct a MLR, a.k.a., state-of-the-evidence review by synthesizing the knowledge from both the formal (peer-reviewed) literature and the grey literature. However, as the first phase towards that objective, we conducted and report a GLR in this paper and postpone the full MLR to the future work. Based on the above goal, we formulated five research questions (RQs):

1. What are the (independent) criteria recommended by practitioners for choosing test right automation tools? How can we classify those criteria?
2. What types of research methods have been utilized in the sources and what types of contribution types have been proposed for choosing the right tools? How are the arguments / criteria for choosing the right test automation tools validated?
3. What types of processes, if any, have been proposed for choosing the right tools? Do those processes vary and if yes, how?
4. What type of evidence there is, if any, for the sources to support the credibility of claims in the sources?
5. What were the most referenced (or compared) tools by the sources? What types of test levels or systems under test (SUTs) were discussed in those sources?

4 METHODOLOGY

4.1 Search Keywords and Source Selection

A GLR was conducted in the fall of 2016. First, we conducted exploratory searches (using the regular Google search engine), using search strings such as "select automation test tool" and "choosing the right test tool". We also explored the related search strings, proposed by Google ("searches related to xxx", where xxx was the previously used string). Our search approach is based on our previous experience in conducting SLR and MLRs, e.g. [14-16]. Additionally, Google provides related searches with some tool names, but naturally those tool names were excluded from the searches. We expanded the final search string according to the relevant results, see **Table** 2. The words within a column were considered as synonyms (combined with OR), while the words in the columns (A-D) constructed the whole search string for the purpose of the research (combined with AND). Thus, the formal search string used was "(select OR choose OR Comparison OR Best OR Right) (Test OR Testing) (automation OR Automated OR Automatic) (Tool OR Framework)".

Search is claimed to be the attempt to make sense of all information just possible to find [2]. Google claims "*a journey of a query starts before you ever type a search*" and a search "*happens billions of times a day in the blink of an eye*" [17]. To return the most useful results for a search query, Google collects and organizes information with crawling and indexing. They claim to use only constantly changing algorithms to determine the sites to crawl as well as interval for and number of pages to fetch from each site [17]. Thus, it is notable that search results may vary. The following factors may affect search results in general: previous search history, previously clicked Google links, geographic location, use of Google account (while searching), type of device used, type of search in general and possible Google ads on the page [24].

Our search using the search string listed above resulted in total of about 194,000,000 results (Google hits), in English. The first 100 results were selected as the initial pool of sources and stored locally to keep the contents.

Table 1: Rationale to include grey literature in state-of the-evidence reviews [3] for choosing the right tool

Complex intervention	**Yes**/No
Complex outcome	**Yes**/No
Lack of consensus about measurement of outcome	Yes/**No**
Low volume of evidence	**Yes**/No
Low quality of evidence	**Yes**/No
Context important to implementing intervention	**Yes**/No

Table 2: Search strings used for the GLR

A	B	C	D
Select	Test	Automation	Tool
Choose	Testing	Automated	Framework
Comparison		Automatic	
Best			
Right			

Table 3: Classification scheme of the sources (attributes & criteria for choosing the right tool)

Source attributes				Classification of Criteria for choosing the right tool		
Demographics	**Website statistics (Number of)**	**Contribution type**	**Type of Research method**	**Test-tool technical**	**Test-tool external**	**Team or Environment related criteria**
Year; Author title; Organization; Number of References;	Readers; Shares; Google hits for the title; Comments; Backlinks	Heuristics / Guidelines; Comparison Framework; Method / Technique; Tool; Model; Metric; Process; Empirical (case) study; Listing tools (Other)	Example Validation Research; Evaluation Research; Philosophical Research; Experience / Opinion; Other	Tool stability; Usability; Reporting capabilities; Test data related; Maintainability; Versatility/Customizability; Scripting language; Capture & Replay; Other	Tool cost / fees; Support for test tool; Vendor evaluation; Other	Matching test requirements; Fit to Operating environment, Tool chain, IDE; Team having necessary skills; Other

Table 4: Company affiliations of the authors of sources

360Logica; AFourTech; Aspire Systems; Avantica; Bitbar Tech.; Blue Ocean Solutions; Brooks Bell; Cigniti; Copyright Clearance Center; Conversion Uplift; Cisco; Falafel Software; G4S India; Gallop Solutions; Gerrard Consulting; GlowTouch Tech.; Happiest Minds Tech.; HCMC Software Testing Club; Infosys; Karl Groves; KMS Tech.; Liberty Mutual; MentorMate; Micro Focus; Microsoft; Ness Tech.; Object Computing; pCloudy; Perfecto Mobile; PractiTest; Principle Logic; QASource; QASymphony; QATestingTools.com; Ranorex; Sauce Labs; SEQIS Software Testing; SmartBear Software; Softcrylic; SoftServe; Software Testing Space; Suyati Tech.; Symbio; TechWell; TestLab4apps; Test Talk; Testuff; The Church of Jesus Christ of Latter-Day-Saint; ThoughtWorks; Traq Software; Trust IV; Xoriant Solutions; Zephyr

We utilized the systematic mapping process as applied in [16]. The process of selecting the sources for the literature review had three phases and was conducted, as follows:

1. **Screening**. The initial pool of sources was reviewed by the 1st author to propose relevant sources for the study.
2. **Application of inclusion/exclusion criteria.** The 2nd and 3rd author of this paper reviewed half of the sources each. Only sources related to the context of choosing the right tool for software test automation (guidelines, processes, comparisons etc.), were to be included. As our focus was on grey literature, we excluded any academic papers. Moreover, to have a clear focus on tool selection, we excluded those sources that only listed different tools (with no comparison or proposed criteria for selection) and listed tools but provided only random criteria (different criteria for tools) for comparison. At this phase the sources had been reviewed by two of the authors.
3. **Voting.** Any collision of opinions between two reviewers was resolved by the third reviewer (2nd or 3rd author). Thus, to be selected for the final pool of sources, a source had to be voted in by at least two of the authors of this paper.

It is notable that every coding in the research data (inclusion, exclusion, identification of a criteria or classification) was justified with a relevant finding from the given source. Thus, all authors were able to come to a decision of their own with the same piece of evidence for the propositions. Screening of grey literature sources can be a time-consuming process since usually there is no applicable abstract or summary available.

4.2 Pool of Sources & the Online Repository

The finalized pool of sources included 60 sources. The research data is available in https://goo.gl/w1eh71. In this paper, the pool of sources is referenced as [Source N], where the word "Source" is used to differentiate the sources of research data from the sources in the bibliography, and "N" identifies the index of the source in the research data. Criteria can be found in an online source [Source N] when there is a marking in the corresponding column.

4.3 Data Extraction and Qualitative Synthesis

The criteria (related to choosing the right tool for test automation) and the classification presented in this paper were formulated by following a systematic qualitative data analysis approach [25]. The analysis was done collaboratively between three researchers. In qualitative data analysis, coding is not only technical, preparatory work for analyzing the data but also "*deep analysis and interpretation of the data's meanings*" [25]. Thus, coding is a heuristic data condensation task in which the most meaningful material is collected and analyzed by reflection. There were a few pre-defined criteria, based on our past knowledge of the area, namely "Matching the test requirements", "Necessary skills", "Tool cost and fees", "Tool stability" and "Operating environment". The final, systematic map was synthesized by generalizing and refining the criteria iteratively, as the review progressed. The list of criteria was formed from distinct concepts and categories in the data, by conducting open and axial coding (i.e. analyzing issues from the sources and establishing relations between those identified concepts). The resulting synthesized

classification of the criteria and the source attributes are shown in Table 3. The refinement process had three phases, as follows:

1. **Identification of Criteria.** The sources were analyzed by the 1st author of this paper. The initial criteria was complemented by proposals for new criteria acquired from the sources.
2. **Classification.** The two other authors of this paper (2nd and 3rd author) both again reviewed half of the sources each, verifying and validating the initially classified data and new proposed criteria. Source attributes were added to assess the quality of the sources [27, 32, 36].
3. **Systematic mapping.** The data was synthetized, systematically mapped, and reviewed by all authors. Finally, any classification related issues were resolved by discussing the matters amongst three authors.

5 RESULTS

5.1 Demographics of the Sources

Some of the sources were published without a specific author (by a company). Mostly the sources were published by identifiable, experienced individuals employed by a consultation company or a tool vendor. Practitioners seem to write mostly about their personal perceptions and experiences rather than about collective reflections. The authors (having identified themselves) possessed titles like "Test Architect", "Software Test Automation Architect", "Principal Test Architect", "Principal Software Engineer" and "QA Engineer", to name a few. Many of the sources were published by a consultation company. The names of all the companies are shown in Table 4. Of the companies, AspireSystems, Bitbar Technologies, SmartBear and QASource were associated with two sources each, while the rest of the companies with just one. Furthermore, two of the sources were personal blogs and one was published by an independent practitioner. Overall, our sources covered 53 different companies. Most sources (about 68%) were published during the past three years (2014-2016), see Fig. 1.

5.2 RQ1 - Criteria for Choosing the Right Tool

To answer the research question regarding the most important criteria, we analyzed and classified criteria from the sources. We compiled a total of 14 different criteria for choosing the right tool. Those were classified as test-tool technical, test-tool external and team/environment related issues, see Fig. 4. On average, the sources referenced 8.9 of the criteria. One of the sources, [Source 27] referenced all 14 specific criteria (not referencing any of the "other" criteria).

The team/environment related criteria (excluding references to the "other" criteria) had the highest average number of references, 50.7. Of all the different criteria, the most referenced one was matching the test requirement ("fit to purpose", n=59), followed by being fit to the operating environment (n=51, both of those team/environment criteria), test tool costs/fees (n=49, test-tool external criteria) and usability (n=46, test-tool technical criteria).

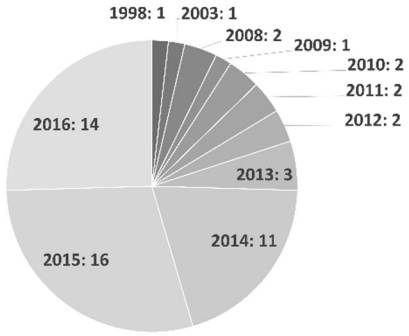

Figure 1: Number of sources published per year

On average, the sources had 4.4 references to eight test-tool technical criteria. Four of the sources had references to all eight criteria ([Source 25], [Source 27], [Source 44] & [Source 55]) and only [Source 47] did not reference any of these criteria. The most referenced test-tool related criteria was usability (nearly 77%), followed by reporting capabilities (about 63%) and tool stability (about 57% of the sources referencing it).

Test-tool external criteria are essential for those considering acquisition of a tool. On average, the sources had 1.9 references to three test-tool external criteria. Two of the sources, [Source 17] & [Source 32], did not have any references to these criteria. The tool related costs were the most referenced criteria (almost 82%), followed by test tool support (about 68%) and vendor evaluation (almost 42% of the sources). We identified one reference to the "other" criteria of this category, namely a political issue (order from parent company or principled restriction to use only some locally supported tool).

The team and environment related criteria were separated from test-tool external criteria as being more related to the actual test environment and people involved in it, thus related to the actual context of the test automation. All sources had at least one reference to three criteria, the average being 2.5. All but one source, [Source 12], had a reference to the criteria matching the test requirements. Being fit to the operating environment was the second most referenced criteria (85%) and team having necessary skills was referenced by 70% of the sources.

5.3 RQ2 – Research Methods and Type of Contribution

5.3.1 Research Methods. The empirical evidence of grey literature is not expected to be scientifically sound. We analyzed the sources based on the evidence of the types of research methods, as in [27, 36], see Table 3. We could mainly identify examples (e.g. examples of weighing pros and cons of tools or implementation solutions, or detailed process descriptions), experiences/opinions (e.g. just listing or processes or explaining different criteria) and other categories from the sources (see Table 5) which is understandable due to the nature of the sources. A

category "Other" was added to the facet of research methods, to include a pure tool comparison study ([Source 40], a comparison of Ranorex and Selenium tools by the tool vendor).

5.3.2 Type of Contribution. The pool of sources was checked based on the type of contribution. The last two types included in our classification scheme for contribution were "comparison framework" and "other". Some sources provided a systematic comparison of two or more tools by some preferred, specific criteria. Such systematic comparison of tools was defined as a "comparison framework". For the sources listing one or more tools (or characteristics of those) without proper comparison we used "others". Most sources were considered to be of heuristic nature or to provide guidelines, see Table 6. Only the [Source 19] was identified as a source providing model and metrics for tool selection. The sources providing a comparison framework were published mainly during the years 2016-2014 (4, 2 and 3 sources, respectively), for pdf-documents the publication year was not available. The [Source 8], having the most backlinks included a comparison framework.

The [Source 17] from year 2003 focused on different test automation frameworks (namely test script modularity, test library architecture, keyword-driven/table-driven/data-driven testing and hybrid test automation) with IBM Rational toolset, instead of different tools. It seems the tool comparisons have gained interest and popularity amongst the practitioners during the past years. In our sources, Selenium was compared mainly with QTP/UFT and TestComplete. The comparison frameworks were roughly similar to the criteria covered and to the level of comparison. However, most of the comparison frameworks emphasized the importance of scripting language, ability to integrate and technical support available. The summary of the criteria having references from at least half of the sources, is shown in Table 7. The emphasis of these top criteria from comparison frameworks slightly differs from the important criteria collected from all the sources, see Fig. 4.

Table 5: Types of research method for the sources

Example [n=7]	[S5, S19 , S34, S41, S49, S50, S51]
Experience/Opinion [n=59]	[S1–S4, S6–S60]
Other [n=1]	[S40]

Table 6: Contribution types of the sources

Heuristics/Guidelines [n=59]	[S1–S18, S20–S60]
Comparison Frameworks [n=11]	[S8, S9, S17, S19, S22, S25, S40, S42, S44, S48, S58]
Method / Technique [n=1]	[S51]
Model [n=1]	[S19]
Metric [n=1]	[S19]
Process [n=7]	[S5, S34, S36, S41, S43, S49, S50]
Listing tools (only) [n=20]	[S8, S9, S13, S19, S21, S22, S25, S26, S28, S30, S37–S40, S42–S44, S47, S48, S58]

Table 7: Top criteria in Comparison Frameworks

Scripting Language [n=9]	[S8, S9, S19, S22, S25, S42, S44, S48, S58]
Ability to Integrate [n=8]	[S8, S17, S19, S22, S25, S42, S44, S48]
Technical Support [n=8]	[S8, S9, S19, S22, S40, S42, S48, S58]
Record &Playback [n=7]	[S8, S9, S22, S25, S40, S44, S58]
Element Identification [n=6]	[S8, S9, S19, S25, S40, S48]
Usability [n=6]	[S9, S17, S19, S22, S25, S40]
Costs [n=6]	[S8, S9, S19, S22, S40, S48]
Mobile Support [n=6]	[S8, S22, S40, S42, S48, S58]

The level of detail in or the accuracy of the comparison frameworks varied. For example, for the criteria "scripting language" some of the sources were very specific by listing available languages for some tools and some did not even cover all compared tools in such comparison, or were very vague in the wording. For example, for Selenium the [Source 9] listed "Many (Java, C#, Perl, Python, etc.)" while the rest of the sources provided an extensive list of names (e.g. "Java, C#, Ruby, Python, Perl, PHP, JavaScript"). For "usability", one of the sources provided numbers [Source 9], while one used verbal comparison "Less" or "Much more" [Source 19] and some, e.g. [Source 22], had clear descriptions, like for JMeter ("Friendly GUI / easy to install") and for NeoLoad ("Ease of use / No scripting required /Single GUI for all actions"). Some of the criteria are ratable and some exposed to interpretation without a proper baseline. None of those sources providing a comparison framework directly indicated any tool better than another, but rather emphasized the unique challenges of every project.

5.4 RQ3 – Processes

Our aim was to check whether the sources proposed processes for choosing the right tools. We intended to define an overview of the phases considered appropriate by the practitioners. Seven of the sources provided some type of a process, either with defined steps or with more general outline. The different process phases, described in the sources, are listed in Table 8. It is notable that the phases are not in definite order and some of the phases seem overlapping, at least considering the first four phases, due to missing or inadequate process descriptions. Only [Source 5] did not include explicit definition of requirement or identification of the problem, and two sources, namely [Source 41] & [Source 43] included those both.

The processes focused mostly on shortlisting some suitable candidate tools. Interestingly, all seven sources proposed to conduct an evaluation of the tool(s) with live trial, Proof of Concept (PoC) or demo. However, those three evaluations can actually be of different nature regarding the SUT, test scenarios

selected, data used and people involved. The [Source 41] stresses that the pilot phase may also have unanticipated effects, i.e. by changing routines or testing procedures in unexpected ways.

Table 8: Tool selection process phases

Select Project Lead [n=1]	[S5]
Assess Desire for Change [n=1]	[S34]
Identify Problem [n=3]	[S34, S41, S43]
Define Requirements [n=5]	[S36, S41, S43, S49, S50]
Evaluation Checklist / Scorecard [n=2]	[S5, S36]
Research Tool Vendors [n=1]	[S5]
Shortlist Tools [n=6]	[S5, S34, S36, S41, S43, S50]
Allocate Resources [n=1]	[S43]
Analyze / Select Top Tools [n=4]	[S34, S36, S49, S50]
Live Trial / Proof-of-Concept or Demo [n=7]	[S5, S34, S36, S41, S43, S49, S50]
Present Results [n=1]	[S36]
Decision [n=3]	[S41, S43, S50]

Only [Source 36] explicitly included a phase where the results are presented to the team. Furthermore, three of the sources specified a phase for making the final decision. It seems the processes were rather general guidelines and as such could be used for selection purposes in contexts other than of this study. In fact, [Source 48] compares the process of acquisition of open source tool and a commercial tool to buying a branded car and assembling a car by oneself.

5.5 RQ4 – Contrasting the Sources

We aimed to examine any evidence available for the sources, to add to credibility of the claims or the sources as such. Statistical data for the sources, specifically, number of readers, number of shares, number of comments and number of Google hits for the title was collected on December 19th, 2016. The figures for the first three of those are shown in Table 9 and for Google Hits in Fig. 2. The data is shown only for the sources having the data available (i.e. providing value of zero or more). Number of readers and shares was used to generate a figure of normalized "popularity" metrics. The number of backlinks was checked for the sources to provide a sign of popularity (Fig. 3).

5.5.1 Number of Readers and Number of Shares. Five of the sources showed the number of readers on the website, see Table 9. The [Source 25] had been read 163360 times since its publication in 2014. However, the number implies the number of visits on the website, whether the article has been read in full is not known. Interestingly, only two sources provide both the number of readers and shares ([Source 36] & [Source 43]).

5.5.2 Number of Comments. Some of the sources allow readers to submit comments. The number of sources having comments was 35, of which 17 had just zero comments, see Table 9. The [Source 13], titled "Automated Testing - How to choose the Best Automation Testing Tool" had the most comments, 81 and it had been shared 32 times. Of those sources having the number of shares available (zero or more) only three sources had more comments than shares ([Source 13], [Source 36] and [Source 37]). Thus, it seems that sharing was more common than commenting on the sources, in general.

Table 9: Website statistics for the sources (when available)

Format: Value [Online Source Reference]	
Number of Readers [n=5]	**163360 [S25]**, 52453 [S26], 26550 [S28], 13503 [S36], 1489 [S43].
Number of Shares [n=17]	19 [S1], 5 [S2], 24 [S6], 106 [S9], 32 [S13], 66 [S22], 45 [S24], 31 [S27], 3 [S31], 2 [S33], 1 [S35], 0 [S36], 0 [S37], 0 [S38], 6 [S43], 16 [S45], **973 [S58]**.
Number of Comments [n=35]	0 [S2], 0 [S4], 0 [S6], 0 [S7], 4 [S9], 29 [S10], **81 [S13]**, 1 [S16], 1 [S17], 0 [S20], 0 [S21], 1 [S22], 0 [S24], 9 [S25], 4 [S26], 11 [S27], 5 [S28], 1 [S31], 0 [S32], 0 [S35], 10 [S36], 11 [S37], 1 [S42], 0 [S43], 2 [S45], 0 [S46], 0 [S50], 10 [S52], 1 [S54], 0 [S55], 0 [S56], 0 [S57], 2 [S58], 0 [S59], 0 [S60].

5.5.3 Number of Google Hits for the Title. We also checked the number of Google hits for the source titles. For titles having very general wording (33 sources), we also used the company name in the search string, e.g. for [Source 20] we used the search string ("Choosing the Right Mobile Test Automation Tool" AND "Blue Ocean Solutions") and for [Source 27] we used ("Guide Test Automation" AND ThoughtWorks). The most hits, about 31100, were found for [Source 44] titled as "Selecting The Right Automated Testing Tool (SmartBear)".

5.5.5 Backlinks. An interesting figure of statistics for the sources was backlinks, a reference comparable to a citation [37]. There are free tools available for analyzing backlinks, some even for academic publications only. However, some tools have limitations, e.g. related to number of returned links or possibility to save the results. After checking a few possibilities, we selected the "Online Backlink Checker" http://theseotools.net/backlink-checker by TheSEOTools.Net. It is free and allows to save the results to a file (the limit of 1000 backlinks was not an issue for the sources). The backlinks were fetched on December 16th, 2016. The smallest number of backlinks was three [Source 38] and the highest was 759 [Source 8]. The sources had on average 136 backlinks, median being 39. As much as 30% of our sources had over 100 backlinks and about 78% had at least 20 backlinks. The number of backlinks for the sources is presented in Fig. 3.

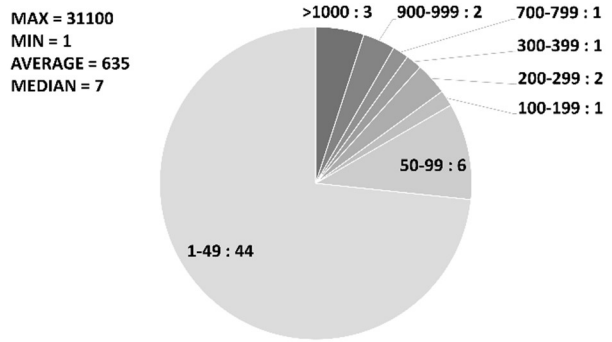

Figure 2: Number of Google Hits for the sources

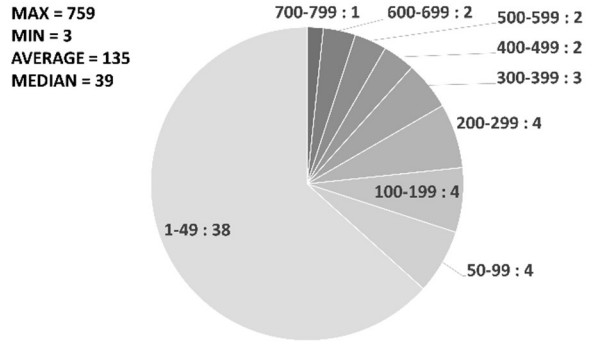

Figure 3: Number of backlinks for the sources

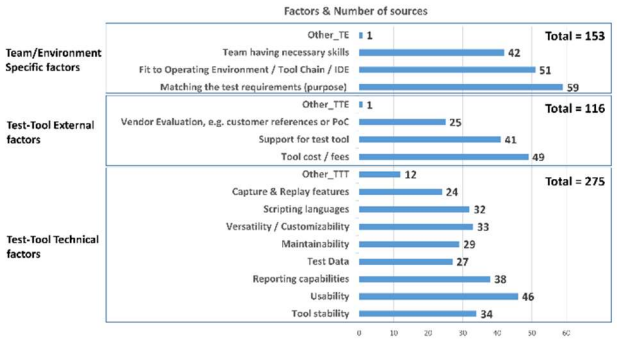

Figure 4: Criteria to consider when choosing the right tool

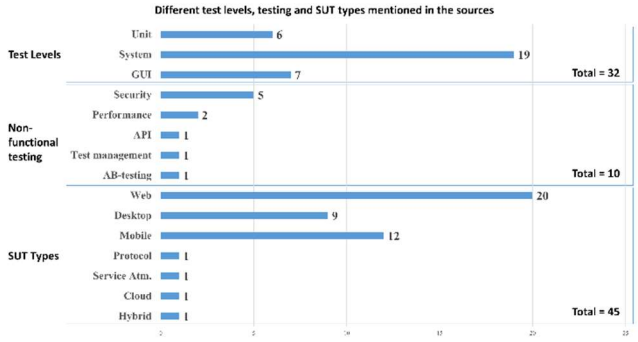

Figure 5: Test levels, test and SUT types mentioned

As a backlink is comparable to a citation, we compared the above findings with our previous citation analysis of papers published in the ESEM symposium [31]. 295 (of the total of 513) papers included in the study had been cited at least once. Those papers had 1,897 citations in total, yielding an average of 6.43 and median of two citations per paper (for the cited papers). It was expected that the longer an academic paper is available, the better the chances for it to be cited [31]. Interestingly, the source having the most backlinks in this study was published in 2016. First, grey literature is freely and easily available for the public. Second, usage of right keywords enhances visibility of a website. Third, tool selection is a topic of interest amongst practitioners worldwide [7, 19]. Thus, a website may become popular in a short period of time.

5.6 RQ5 – Tools and Test Levels

Finally, we focused on analyzing the most popular tools referred to by name in the sources. There was a total of 203 references to different tools, of which 137 were unique. Selenium was the tool of most interest with 15 references (Selenium IDE and WebDriver were both mentioned just once and counted separately). Some sources focused on or discussed about one or more specific test level, test type or SUT type, see Fig. 5. Some sources ([Source 6], [Source 23] & [Source 52]) mentioned a specific domain: ERP, Network and Apps & Games.

6 DISCUSSION

Regarding RQ1, in the test-tool technical issues, the most important criteria seemed to be usability (nearly 77% of the sources referencing it). In the test-tool external issues, the most important criteria was tool costs/fees (nearly 82%). Unsurprisingly, the criteria with most references (about 98%) in the team/environment related category was matching the test requirements. In the findings of [30] the category most referenced in that study was usability, followed by functional suitability, maintainability and costs, as the first four categories. Both findings indicate that although costs may be important, usability and functional suitability may need to be the primary drivers for success in the process of choosing the right tool.

Analysis of research methods used in and contribution types provided by our sources as RQ2 revealed that our sources were mainly considered to be of heuristic nature or to provide guidelines, see Table 5 and Table 6. Only one of the sources provided both a model and metrics for tool selection and eleven sources provided a comparison framework (a listing of differences between functionality or of important characteristics of a few tools). Regarding the contribution types, we could mainly identify examples and experiences/opinions. We identified and analyzed proposed processes for RQ3. All seven sources providing a process as a contribution included a live trial, PoC or demo in their listing, see Table 8. While all those proposed processes emphasized the need and purpose of tool evaluation, less than half of the sources referenced vendor evaluation. In general, the processes were mostly personal reflections, based on experience on work in the field.

Interestingly, in early 1990's a study claimed that trial use would often lead to wrong decisions [28]. The findings of that study (at the time) defined lack of time to be the biggest problem for such trial, followed by different levels of user expertise required for evaluation. Thus, tool evaluation would be recommended only if there were people available who could "*devote enough time and appropriate expertise to complete a thorough trial use*". Otherwise it was recommended to rely on the "*well-researched evidence*" [28]. Only a few of our sources providing a process explicitly required expertise or experience from the practitioners participating in the evaluation process, but rather referred to "*time and resources*" [Source 34], "*suitable resources*" [Source 43] or "*time and effort*" [Source 50]. The [Source 49] stressed that "*Tool evaluation… requires a lot of research irrespective who does the evaluation*". In fact, problems related to the topic seem to be acknowledged by practitioners as test tools and automation related services are ranked among the most required services from external consultants [19].

Grey literature seem to have its place in SE, not only in serving the practitioners but also in providing an interesting aspect into academic studies. We analyzed the evidence available to add credibility for the claims and sources as RQ4. Few of the sources provided specific evidence about their popularity, see Table 9. However, the number of backlinks for the sources shows somewhat significant importance to the public, e.g. [Source 25] had 163360 readers, was shared nine times and had 16 backlinks. Similarly, [Source 26] had 52453 readers, four comments and as many as 247 backlinks. As a reference, a peer-reviewed paper of comparative study on Selenium, QTP and TestComplete, published in 2013 [21] had 14 citations (according to results from Google Scholar, http://scholar.google.com/, Dec. 29th, 2016). Comparative figures such as number or comments or shares seem to be either unimportant or inessential (or perhaps unwanted) for sources of grey literature, in general.

The RQ5 focused on the most referenced tools, test levels and SUTs. The tools referred to by name, either in the comparison frameworks or just by listing tools, seemed to provide rather similar results as found in [30] where Selenium and UFT & QTP were the 2nd and 4th most popular software test automation tools. Eight tools of the list of the most referenced tools were also included in the top 15 tools in [30], namely Selenium, UFT/QTP, Appium, Sikuli, Fitnesse, Junit and SoapUI. The finding is interesting, in particular, since our study focused on the process of choosing the right tool, in general, thus not on any specific domain, testing type or testing method.

6.1 RQ5 – Threats to Validity

This section provides a discussion about the limitations of this research and validity of the results presented. We focused only on grey literature (in English) collected from the Internet using Google search. The non-scientific nature and possible issues related to data collection of the sources were acknowledged by the researchers. *Internal validity* was addressed by performing initial searches to refine the research strings. The analysis of the selected data was subject to interpretation and debate. To reduce bias in the selection and analysis of the sources and to have mutual agreement on the research data, we reviewed the sources in pairs. The coding of the research data (inclusion, exclusion, identification of a criteria or classification) was justified with a relevant finding from the given source. Thus, the same pieces of evidence were used for evaluating the decisions. However, we realize that the search algorithms (and related factors, discussed in Chapter 4.1.) and the choice our search terms affected the results. Also, utilization of freely available sources may have led to bias in favor of open source tools. To consider *construct validity* we utilized the process proposed in [27] and reviewed and refined the data iteratively to have a consensus on the adequate criteria and metrics for the study. For *conclusions validity*, we acknowledge our analysis was biased by our interpretation and even possibly by our own experience although the sources were published by professionals in the diverse field (of domains) of SE. Regarding *external validity*, the findings of the research are not fully generalizable as such. The contextual criteria have a significant impact on the process. We plan to conduct a comprehensive MLR to include scientifically sound perspective to the topic.

7 CONCLUSION & FUTURE WORK

In this study, we conducted a Grey Literature Review (GLR) to listen to the "voice" of practitioners in the process of choosing the right tool for software test automation. We analyzed also the credibility and popularity of the pool of our sources, i.e. origin of the sources and any evidence how those sources have been adopted by the practitioners. The sources were mostly published by experienced, identifiable practitioners (representing 53 companies) and mostly based on experiences or opinions while some provided e.g. examples of or weighing pros and cons of tools or implementation solutions. As contribution, most of the sources just provided heuristics or general guidelines, listed some tools, provided a comparison framework (more detailed comparison of two or more tools by some preferred, specific criteria) or proposed a process.

Despite the limitations of the study, the findings allow us to predict that although there is rather common consensus about the important criteria, those are not used systematically. Although the

team and environment related, context specific criteria seemed to be the most referenced criteria, only about half of the test-tool technical criteria were referenced by the sources, on average. The criteria seem to be context specific and not applicable as such in all cases. Some criteria may be e.g. important in two different contexts but in totally different phases of the selection process. Moreover, the context may define the significant weight for criteria, too. For example, for a project having a limited budget costs could be the major driver for the selection process as a whole (only focusing on open source tools or compromising on features) while for some other project costs could be negotiable, a matter of preference when comparing the possible candidates and making the final choice.

Although there seems to be a plethora of different software testing tools available, the practitioners seem to be interested in, prefer and adopt the widely known and used tools. There is no rigor process for selecting the candidate tools for tool evaluation although the sources providing a process as a contribution included a live trial, proof-of-concept or demo phase in their listing. The most referenced, or compared tools were Selenium, QTP/UFT and TestComplete. The familiarity or popularity of Selenium was unsurprising as it was also the most popular true testing tool in our previous study [25]. Surprisingly, the same tools seem to be popular despite the research method, i.e. surveys, web-scraping and GLR. (Web-scraping is a technique to access web-pages, to extract a structured view of the desired data from the internet [26].) Cialdini [10] argues that the increasing tendency for cognitive overload is likely to increase the prevalence of shortcut decision making proportionately. He explains "social proof" as tendency to see a situation as favorable or appropriate when others are doing it normally. "Social proof" as a weapon of influence is claimed to be most influential under two conditions: uncertainty and similarity [10], which seem to apply to our topic.

In future research we plan to conduct a more comprehensive Multivocal Literature Review (MLR) on the topic of choosing the right tool for software test automation with intent to demystify tool evaluation at large.

ACKNOWLEDGMENTS

Vahid Garousi was supported by the National Research Fund, Luxembourg FNR/P10/03.

REFERENCES

[1] S. Allard, K. J. Levine and C. Tenopir. 2009. Design engineers and technical professionals at work: Observing information usage in the workplace. *J. Am. Soc. Inf. Sci. Technol.* 60, 3, 443-454. DOI: 10.1002/asi.21004

[2] J. Battelle. 2006. *The Search: How Google and Its Rivals Rewrote the Rules of Business and Transformed Our Culture.* Nicholas Brealey Publishing, London, UK.

[3] Karen M. Benzies, Shahirose Premji, K. A. Hayden and Karen Serrett. 2006. State□of□the□evidence reviews: advantages and challenges of including grey literature. *Worldviews on Evidence□Based Nursing.* Wiley Online Library 3, 2, 55-61. DOI: http://onlinelibrary.wiley.com/doi/10.1111/j.1741-6787.2006.00051.x/full

[4] Antonia Bertolino. 2007. Software testing research: Achievements, challenges, dreams. *In 2007 Future of Software Engineering* IEEE Computer Society, 85-103.

[5] Emil Borjesson and Robert Feldt. 2012. Automated system testing using visual gui testing tools: A comparative study in industry. *In Software Testing, Verification and Validation (ICST), 2012 IEEE Fifth International Conference on* IEEE, 350-359. DOI:10.1109/ICST.2012.115

[6] Tilmann Bruckhaus, NH Madhavii, Ingrid Janssen and John Henshaw. 1996. The impact of tools on software productivity. *IEEE Software.* IEEE 13, 5, 29-38.

[7] Capgemini Consulting. 2015. World Quality Report 2015-2016. https://www.capgemini.com/thought-leadership/world-quality-report-2015-16.

[8] Capgemini Consulting, J. Vaitilo and N. L. Madsen. 2016. World Quality report 2015-2016, Nordic Region. https://www.sogeti.com/globalassets/global/downloads/testing/wqr-2015-16/wqr-2015_country-pullouts_nordic-region_v1.pdf.

[9] Elliot J. Chikofsky, David E. Martin and Hugh Chang. 1992. Assessing the state of tools assessment. *IEEE Software.* IEEE Computer Society 9, 3, 18.

[10] Robert B. Cialdini. 2001. *Influence: Science and practice.* Allyn & Bacon, Boston.

[11] Tore Dybä, Barbara Kitchenham and Magne Jorgensen. 2005. Evidence-based software engineering for practitioners. *Software, IEEE.* IEEE 22, 1, 58-65.

[12] Tore Dybå, Neil Maiden and Robert Glass. 2014. The reflective software engineer: reflective practice. *IEEE Software.* IEEE 31, 4, 32-36. DOI: https://doi.org/10.1109/MS.2014.97

[13] Raya Fidel and Maurice Green. 2004. The many faces of accessibility: engineers' perception of information sources. *Information Processing & Management.* Elsevier 40, 3, 563-581. DOI: http://dx.doi.org/10.1016/S0306-4573(03)00003-7

[14] Vahid Garousi, Michael Felderer and Mika V. Mäntylä. 2016. The need for multivocal literature reviews in software engineering: complementing systematic literature reviews with grey literature. *In Proceedings of the 20th International Conference on Evaluation and Assessment in Software Engineering* ACM, 26. DOI:http://dx.doi.org/10.1145/2915970.2916008

[15] Vahid Garousi and Mika V. Mäntylä. 2016. A systematic literature review of literature reviews in software testing. *Information and Software Technology.* Elsevier 80, 195-216. DOI: http://dx.doi.org/10.1016/j.infsof.2016.09.002

[16] Vahid Garousi and Mika V. Mäntylä. 2016. When and what to automate in software testing? A multi-vocal literature review. *Information and Software Technology.* Elsevier DOI: http://dx.doi.org/10.1016/j.infsof.2016.04.015

[17] Inc Google. n.d. How Search Works - Inside Search - Google. https://www.google.com/insidesearch/howsearchworks/.

[18] Morten Hertzum and Annelise M. Pejtersen. 2000. The information-seeking practices of engineers: searching for documents as well as for people. *Information Processing & Management.* 36, 5, 761-778. DOI: http://dx.doi.org/10.1016/S0306-4573(00)00011-X

[19] ISTQB (International Software Testing Qualifications Board). 2016. ISTQB® Worldwide Software Testing Practices Report 2015 - 2016. http://www.istqb.org/references/surveys/istqb-worldwide-software-testing-practices-report.html.

[20] Jussi Kasurinen, Ossi Taipale and Kari Smolander. 2010. Software test automation in practice: empirical observations. *Advances in Software Engineering.* Hindawi Publishing Corporation 2010, DOI: http://dx.doi.org/10.1155/2010/620836

[21] Harpreet Kaur and Gagan Gupta. 2013. Comparative Study of Automated Testing Tools: Selenium, Quick Test Professional and Testcomplete. *Int.Journal of Engineering Research and Applications ISSN.* Citeseer 1739-1743.

[22] Barbara Kitchenham. 2004. *Procedures for performing systematic reviews.*

[23] A. H. Maslow. 1966. *The Psychology of Science: A Reconnaissance.* Maurice Bassett, 2004,

[24] M. McEvoy. 2015. 7 Reasons Google Search Results Vary Dramatically. http://www.webpresencesolutions.net/7-reasons-google-search-results-vary-dramatically/

[25] Matthew B. Miles, A. M. Huberman and Johnny Saldana. 2013. *Qualitative data analysis: A methods sourcebook.* SAGE Publications, Incorporated,

[26] Alberto Pan, Juan Raposo, Manuel Álvarez, Justo Hidalgo and Ángel Viña. 2002. Semi-Automatic Wrapper Generation for Commercial Web Sources. *Engineering Information Systems in the Internet Context.* 231, 265-283.

[27] Kai Petersen, Robert Feldt, Shahid Mujtaba and Michael Mattsson. 2008. Systematic mapping studies in software engineering. *In 12th international conference on evaluation and assessment in software engineering*

[28] Robert M. Poston and Michael P. Sexton. 1992. Evaluating and selecting testing tools. *Software, IEEE.* IEEE 9, 3, 33-42.

[29] Dudekula M. Rafi, Katam R. K. Moses, Kai Petersen and Mika V. Mäntylä. 2012. Benefits and limitations of automated software testing: Systematic literature review and practitioner survey. *In Proceedings of the 7th International Workshop on Automation of Software Test* IEEE Press, 36-42.

[30] P. Raulamo-Jurvanen, K. Kakkonen and M. Mäntylä. 2016. Using Surveys and Web-scarping to Select Tools for Software Testing Consultancy. *In Proceedings of the 17th International Conference on Product-Focused*

*Software Process Improvement. P. Abrahamsson et al. (Eds.), PROFES2016.*1-16. DOI:10.1007/978-3-319-49094-6_18

[31] Paivi Raulamo-Jurvanen, Mika V. Mantyla and Vahid Garousi. 2015. Citation and Topic Analysis of the ESEM papers. *In 2015 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)* IEEE, 1-4. DOI:https://doi.org/10.1109/ESEM.2015.7321193

[32] J. Tyndall. 2010. *The AACODS checklist.*

[33] Michiel Van Genuchten. 1991. Why is software late? An empirical study of reasons for delay in software development. *IEEE Trans. Software Eng.* IEEE 17, 6, 582-590. DOI: https://doi.org/10.1109/32.87283

[34] Shuang Wang and Jeff Offutt. 2009. Comparison of unit-level automated test generation tools. *In Software Testing, Verification and Validation Workshops, 2009. ICSTW'09. International Conference on* IEEE, 210-219. DOI:10.1109/ICSTW.2009.36

[35] Johannes Weiss, Alexander Schill, Ingo Richter and Peter Mandl. 2016. Literature Review of Empirical Research Studies within the Domain of Acceptance Testing. *In Software Engineering and Advanced Applications (SEAA), 2016 42th Euromicro Conference on* IEEE, 181-188.

[36] Roel Wieringa, Neil Maiden, Nancy Mead and Colette Rolland. 2006. Requirements engineering paper classification and evaluation criteria: a proposal and a discussion. *Requirements Engineering.* Springer 11, 1, 102-107. DOI: 10.1007/s00766-005-0021-6

[37] The F. E. Wikipedia. 2016. Backlink. https://en.wikipedia.org/wiki/Backlink.

[38] Wiktionary contributors. 2016. If all you have is a hammer, everything looks like a nail. https://en.wiktionary.org/wiki/if_all_you_have_is_a_hammer,_everything_looks_like_a_nail.

[39] S. Yehezkel. 2016. Test Automation Survey 2016. http://blog.testproject.io/2016/03/16/test-automation-survey-2016/